UNIVERSITY OF SOUTHAMPTON

Faculty of Engineering and Physical Sciences
School of Engineering

An Investigation into Data Driven Modelling of Rail Degradation due to Rolling Contact Fatigue

by

Christina Marie Riley

ORCiD: 0000-0002-3415-3297

A thesis for the degree of
Doctor of Philosophy

June 2023

University of Southampton

Abstract

Faculty of Engineering and Physical Sciences
School of Engineering

Doctor of Philosophy

An Investigation into Data Driven Modelling of Rail Degradation due to Rolling Contact Fatigue

by Christina Marie Riley

One of the major problems affecting the UK rail network is a family of defects known as Rolling Contact Fatigue (RCF). RCF is a phenomena which arises from repeated contact stresses at the wheel-rail interface resulting in cracks forming at the rail surface, which if left unmanaged can lead to rail fracture. Management of RCF is largely performed using re-profiling methods such as rail grinding and milling. The objectives of such techniques are to restore rail profiles, remove minor cracks, and stall cracks in their early stages of growth, and therefore these activities have typically been performed cyclically at time (or traffic) based intervals. In recent years, the advances in monitoring technologies has dramatically increased the data available to the network operator, in particular Eddy Current technology, which is capable of identifying the depths of RCF cracks in their early stages. This data set is previously unexplored, and presents the opportunity for investigating modern data mining methods to discover insights that may better inform RCF maintenance strategies. Real, operational data however are often noisy, and if the noise is not accounted for can have significant implications on the accuracy of subsequent analysis and modelling.

This thesis thus investigates the use of numerous data pre-processing techniques which enable Eddy Current data to be reliably used for information extraction and data-driven modelling. In particular, we address the difficulties in spatially aligning low frequency, sparse data by incorporating data partitioning, cross correlation and optimisation methods. Additionally, the successful preparation of the data enables two main approaches to be explored. Firstly, simple analytical techniques are applied to derive degradation patterns which can augment the current preventive and corrective maintenance decision making processes. Secondly, we demonstrate a methodology for developing a RCF prediction model using several machine learning algorithms for regression analysis. Whilst the resulting models show excellent function fitting capabilities, particularly in the case of ensemble, tree-based methods, we also highlight the potential problems that may arise when using these methods. Despite this, future developments of these models could present excellent opportunities for modelling these complex relationships. At the same time, the data processing and analytical techniques could be presently incorporated into existing RCF management strategies.

Contents

List of Figures

List of Tables

Declaration of Authorship

I declare that this thesis and the work presented in it is my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Parts of this work have been published as:

Signed:..... Date:.....

Acknowledgements

There are many people that I would like to thank, without whom I'm not sure I would have made it through to the end.

First of all to my supervisors, Andy and Ondrej for your guidance and support especially during those early months of 2020. Andy, thank you for your insight, patience and understanding, especially the incredibly last minute and unsociably timed corrections over your Christmas break.

To the many people at Network Rail for your contributions. Patric Mak for passing on your wealth of knowledge when I started, Ron Sutherland for always being there to help and also your support with SQL and NR Software issues. Julian Williams, thank you for answering my unrelenting stream of questions, your company in office and expertise were invaluable. To Kevin Anderson, Brian Whitney, Mark Burstow for your technical input, and Matt Keillor for helping me with Track-Ex. Andy Kirwan, Piers Treacher and the rest of the WLC team; thank you for welcoming me into your team from the beginning. To Susan, Richard and the rest of my wonderful adopted team for keeping me sane in the office, I will really miss all of you.

To my family, especially my Mum and Dad for your encouragement, support and always inspiring me to do more, I wouldn't have even considered this path without you. Thanks also Mum for reviewing my work, and as always of course to you both for helping with the kids. To Jo and Kev, for your unwavering support especially with the children, you never fail to put everyone else first. To Ria, Jimbo, Luke, Em, Theresa, Frank, Oli, Amelie and Louis, I am so incredibly lucky to have a family like you.

To my children, for always giving me a sense of perspective; Sofia for your unique approach to life and ability to entertain and be present, to Luka for being the best conversationalist and the most caring brother to your sister Sofia, and to Baby Number 3 for spurring me on in the final months. And above all to my life partner Matt, for being the greatest father to our children, for keeping my spirits high, and for supporting me even on the tough days, I absolutely could not have done this without you.

For my family Matt, Sofia, Luka and Baby T

List of Acronyms

ACTRAFF	Actual TRAFFic
ANN	Artificial Neural Network
BNSF	Burlington Northern Santa Fe
CBM	Condition-Based Maintenance
CNN	Convolutional Neural Network
CPR	Canadian Pacific Railway
DD	Data-driven
DDM	Data-driven Modelling
DL	Deep Learning
DVT	Driving Van Trailer
EC	Eddy Current
ECT	Eddy Current Testing
ELR	Engineers Line Reference
EMGTPA	Equivalent Mega Gross Tonnes per Annum
EPSRC	Engineering and Physical Sciences Research Council
FOC	Freight Operating Company
FE	Finite Element
FEM	Finite Element Methods
FN	False Negative
FNN	Feed-forward Neural Network
FP	False Positive
GCC	Gauge Corner Cracking
GDI	Grand Damage Index
GEML	Great Eastern Mainline
IM	Infrastructure Manager
INM	Integrated Network Model
kNN	K-Nearest Neighbours
KNN	K-Nearest Neighbours
KPI	Key Performance Indicator
LH	Left Hand
LHR	Left Hand Rail
LOF	Local Outlier Factor
LR	Linear Regression
MAC	Minimum Action Code
MAE	Mean Absolute Error
MAUP	Modifiable Area Unit Problem
MBD	Multi Body Dynamics
MBS	Multi Body Simulations
MGT	Mega Tonnes
ML	Machine Learning
MLP	Multi Layer Perceptron
MMH	Maximal Margin Hyperplane
MWR	Magic Wear Rate
MP	Multi-Pass
MSE	Mean Squared Error

NaN	Not a Number
NDE	Non Destructive Evaluation
NDT	Non Destructive Testing
NETRAFF	Net Traffic
NN	Neural Network
NR	Network Rail
NRC	National Research Council
NRIL	Network Rail Infrastructure Limited
RAM	Route Asset Manager
RCF	Rolling Contact Fatigue
RDMS	Rail Defect Management System
RF	Random Forest
RH	Right Hand
RHR	Right Hand Rail
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
RRV	Road-Rail Vehicle
RSU	Roller Search Unit
SC	Switches and Crossings
ST	Spatio-Temporal
STDM	Spatio-Temporal Data Mining
SMC	Soft Margin Classifier
SML	Supervised Machine Learning
SP	Single Pass
SVC	Support Vector Classifier
SVM	Support Vector Machine
SVR	Support Vector Regression
SWML	South Western Main Line
TID	Track ID
TL1	Track Length 1
TL2	Track Length 2
TOC	Train Operating Company
TRGM	Track Geometry
TRV	Track Recording Vehicle
UT	Ultrasonic Testing
UTU	Ultrasonic Test Unit
VBA	Visual Basic
VDM	Vehicle Damage Matrix
VTAC	Vehicle Track Access Calculator
WLRM	Whole Life Rail Model
WRI	Wheel-Rail Interface

Chapter 1

Introduction

1.1 Motivation

Railway operators have endured unprecedented turbulence in recent years. The height of the COVID 19 pandemic resulted in cessation of passenger traffic, followed by the emergence of new working patterns and changed attitudes toward rail travel. Over the 2019/20 - 2020/21 period there was a 78% reduction in passenger journeys (Department for Transport, 2021), and a reduction of 41.8% for the 2021/2022 period compared with 2019/20 (Network Rail, 2022), indicating a sustained reduction in passenger uptake since restrictions were lifted. Yet there remains sustained pressure to improve performance, with faster, higher capacity and heavier trains, whilst simultaneously ensuring system reliability and safety at a reasonable cost to network users. Maintenance budgets are tightening, and strategies must therefore balance these conflicting aims to ensure optimum efficiency and long term viability of railway operations.

Maintenance of Britain's railway infrastructure is the responsibility of network owner and operator Network Rail Infrastructure Limited (NRIL), who oversee the 20,000 miles of track, 30,000 bridges, tunnels and viaducts and the thousands of signals, level crossings and stations (Network Rail, 2022). In 2021, the total amount spent on maintenance and renewals accounted for 49% of overall expenditure (Network Rail, 2021b), £1,892m was spent on maintenance alone, £3,910m on renewals, with an additional £288m spent on Schedule 4 performance regime payments (Network Rail, 2021b). [1]

The rails are a critical part of the track subsystem, subject to high and repetitive vertical and longitudinal contact forces at the wheel-rail interface. These pressures

[1] Schedule 4 payments are aimed to compensate the train operator for loss of service where the Infrastructure Manager (IM) take back possession of areas of the network, normally as a result of planned maintenance works.

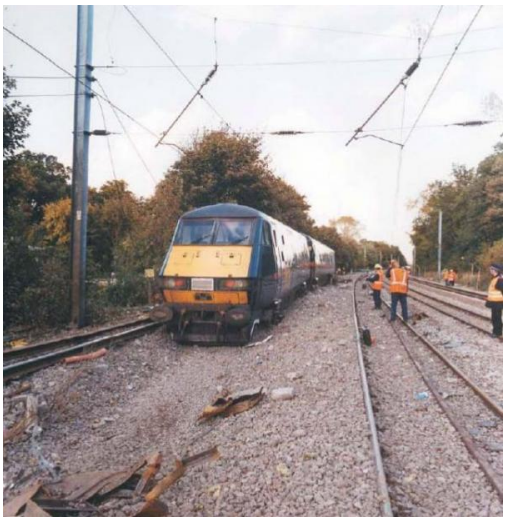
eventually result in the formation of defects, which can lead to rail breaks and potential derailments if left unmanaged. There are few incidences more devastating in recent years than the Hatfield crash of October 2000, where four people were killed and 34 injured when a GNER train travelling from King's Cross to Leeds was derailed not far from Hatfield (ORR, 2006).

Images from the crash site (see Figures 1.1, 1.2a, 1.2b) exposed the catastrophic failure of the high rail around Welham Curve, which shattered into some 200 pieces (ORR, 2006). The formal enquiry stated that the immediate cause was a result of the 'fracture and subsequent fragmentation of the high rail over a 35-metre length due to substantial transverse fatigue defects in the rail head. These had their origins in gauge corner cracks, a form of Rolling Contact Fatigue (RCF), which had developed in the rail surface.' (ORR, 2006).

FIGURE 1.1: The Hatfield rail track site following the accident (Grassie, 2005)

Rolling Contact Fatigue (RCF) is a family of fatigue-driven damage phenomena, characterised by small cracks forming in the running band of the rail. The cracks are caused by repeated cyclic loading, in combination with frictional forces due to the rolling/ sliding contact, which eventually exhaust the ductility of the rail steel (Magel et al., 2004). Once the cracks reach a critical size, they have the potential to propagate to downward turning transverse cracks that can cause the rail to break.

A key approach to managing RCF defects is re-profiling of the rail by grinding or milling. These approaches generally have two aims; firstly to restore the rail profile and optimal wheel/rail contact to reduce contact pressures, and secondly to remove minor surface cracks and stall moderate cracks from propagating. In the years following Hatfield, an enormous programme of measures (the 'National Recovery Programme') to deal with widespread RCF included huge amounts of rail renewals



(A) Looking North in the direction of travel showing the Driving Van Trailer (DVT) and coach M

(B) Looking North from the area of derailment

FIGURE 1.2: Images from the Hatfield Crash Site (ORR, 2006)

and corrective grinding, much of which was likely to be over-compensatory due to a lack of understanding of the phenomena (Merkert, 2005; Grassie, 2005). Premature renewals and unnecessary grinding not only reduces the life of the rail though loss of material and premature replacement, but additionally increases maintenance costs and can cause disruptions to track availability.

Historically, the predominant forms of maintenance policy within the railway industry have been *reactive*- e.g. where the rail is replaced when it has exceeded its wear limits or critical defects have occurred, and *time-based* (typically based on tonnage thresholds being reached). The outcome is generally that the replacement comes too late when the system is run to failure, or the intervals are too conservative due to historic failures such as Hatfield and insufficient understanding of how the system degrades. Condition-Based Maintenance (CBM) policies are widely viewed to be the most economical, aiming to maximise network availability whilst reducing the risk of failure (Al-Douri et al., 2016). In CBM, system health is determined by dynamically monitoring the conditions of the asset, and pre-empting faults before they occur, so that interventions can be planned accordingly (Lin, 2005). Asset degradation models provide a distinct advantage, since predicting future states and understanding key drivers can assist in determining when and where the intervention should be employed in a robust and repeatable manner (Fumeo et al., 2015). These models may be based on the physical laws describing system behaviour or derived from operational condition data (data-driven).

In conclusion, to balance the safety of network users, service reliability and shrinking budgets, it is important that a robust methodology is developed based on evidence and understanding of rail degradation to inform maintenance planning.

1.2 Problem Statement

The work in this thesis addresses the balance between managing the safety of rail users, with the pressing need for efficiency due to tighter budget constraints and performance targets. Grinding is a critical strategy for managing RCF, but it is expensive, may cause disruptions to railway operations and must be done effectively rather than needlessly reducing rail life.

Ideally, in order to establish an optimised, CBM strategy for the planning of when and where grinding is performed, two approaches can be explored. Primarily, the operator can gauge a better understanding of the phenomena and its development across the network by utilising monitoring data and analysing degradation rates. Secondly, the development of physical or data-driven asset degradation models for predicting RCF may assist in understanding and predicting possible future issues. Both approaches provide a consistent approach for informing the decision making process and can incorporate real asset data.

Currently there are two predominant strategies employed by Network Rail (NR) for rail re-profiling, 1) Preventive, and 2) Corrective grinding.

Preventive grinding is performed in the first instance to ensure that the rail profile is restored to its optimal shape, thus alleviating the contact stresses on the gauge corner. Yet, the additional benefit of surface re-profiling is the removal of minor cracks, and prevention of moderate sized cracks from propagating. The prevailing idea is that if cracks are growing into the rail at a particular rate, then the loss of material by both natural and deliberate means (wear and grinding, respectively) must at least reach this rate in order to contain cracking (Grassie, 2005), this rate is termed by Kalousek (1997) as the 'Magic Wear Rate (MWR)'. This type of grinding involves cyclically removing only a small amount of rail material with grinding trains that can travel at moderate speeds. The current practice at NR is to grind curves after every 15 MGT of traffic, and tangent track after every 45 MGT (Network Rail, 2018b). It is understood that this strategy is derived from the studies conducted by the National Research Council (NRC) of Canada on heavy haul lines in North America (Stanford et al., 2001). However, the operational and environmental conditions observed at the time of this study will differ considerably from the UK network today, and therefore these strategies require updating based on the UK operational data.

Corrective grinding conversely is targetted at treating existing RCF sites or sections of track before there is substantial risk of the rail breaking. Typically it is reactionary and requires multiple grinding passes, machine milling, or in some cases re-railing. At NR, EC and ultrasonic measurements are used to guide the planning of these activities. The process varies between the network regions and is dependent on the Route Asset Manager (RAM), but is typically based on an assessment of site risk carried out by

experts. Determination of risk and rates of degradation are performed by observation and expertise rather than through repeatable, objective analyses.

Over the past 20 years since the Hatfield crash, several physics-based RCF damage evolution models have been developed, varying considerably in complexity and applicability, as discussed by Krishna et al. (2021). Many of the widely used approaches employ Multi Body Simulations (MBS) of railway vehicles to obtain the dynamic wheel/ rail loads and kinematics for a range of running conditions. However, these approaches, as far as is understood from the literature, and through discussions with maintenance teams at NR, are rarely used to inform grinding practices.

In recent years, the application of data-driven techniques has grown in popularity due to the surge in availability of large data sets and advancements in computing power. In data-driven approaches, the relationships between system input and output variables are determined directly from the system experimental data. Contrary to physical models, Data-driven Modelling (DDM) is not reliant on explicit modelling of the underlying system, and can therefore capture complexities and patterns which may otherwise be missed. Their appealing flexibility however can sometimes be their downfall, as without a good understanding of the system under study and preparation of the data, these methods can fall foul of modelling noise rather than capturing the process of interest (Cios et al., 1998). Historically, RCF cracking is detected using ultrasonic methods, but emerging technologies such as EC based systems are able to detect cracks in their very early stages of development, and thus generate larger data sets with the opportunity for data mining.

In summary, NR presently utilise two key grinding strategies, neither of which capitalise on the availability of operational condition data or existing RCF initiation models for various reasons that will be considered in Chapter 2. This work seeks to advance these strategies through combining these elements with the objective of:

- Improving Safety: better planning of grinding can ensure interventions are performed before defects become critical,
- Improving Efficiency: significant savings could be made by reducing unnecessary intervention and increasing rail life,
- Improving Consistency: a consistent, evidence based approach across the network that can be validated/ tested in future.

The proposal described in this thesis is the use of data mining techniques with RCF condition data to provide insights which inform the decision making process for rail grinding planning. The main body of work is divided into three parts:

1. EC Data Processing: Due to issues with data quality, the first part of this thesis discusses methods to clean, process and formulate the data mining problem with a set of modelling data, which will enable subsequent parts 2 and 3,
2. Data Analysis: Presents two relatively simple, analytical approaches for using the pre-processed EC RCF data (in particular degradation rates and degradation based Key Performance Indicators (KPIs)) to augment existing preventive and corrective interventions strategies,
3. Regression Analysis: illustrates the development and utilisation of Machine Learning (ML) algorithms to mine the RCF condition data and generate a suitable model for RCF prediction. Additionally, this section presents a method for comparing these DDMs, and the observed damage with a simulation generated using the software tool Track-Ex, which implements a simplified version of the Whole Life Rail Model (WLRM).

This thesis presents a body of work that demonstrates the use of previously unexplored rail condition data, a comprehensive and repeatable methodology for processing this data so that it can be readily used to augment existing interventions strategies through data analysis or can be used to build a RCF prediction model using ML algorithms. This work has been conducted in conjunction, and with support from the UK rail operator Network Rail Infrastructure Limited (NRIL) and the Engineering and Physical Sciences Research Council (EPSRC) (Grant Number 18000134). Operational and maintenance data provided by NR from the UK rail network is utilised.

1.3 Research Aims

The objective of this research is the advancement of grinding planning with the focus on analysing condition data and the application of data-driven modelling for RCF prediction. The primary research question is stated as follows:

> *Can data-driven methods be used to build an effective data-driven RCF prediction model to help inform grinding strategies?*

To commence the body of work required to answer this question, a number of sub-questions were formed:

1. What are the critical drivers for RCF initiation and degradation?
2. How are RCF cracks detected and monitored?

3. What are the current grinding practices in use by Network Rail?
4. What existing models are available for RCF evolution modelling?
5. How can the available RCF condition data be formulated for regression analysis?
6. What insights can be gained from the RCF condition data?
7. Can regression analysis be used to build an effective RCF regression model?
8. How does a data-driven approach compare with existing physic-based models?

1.4 Outline of Thesis

This thesis is divided into two major parts, Part 1 (Chapters 2-3) introduces and describes the problem domain, including the background literature on RCF, RCF mitigation practices and RCF modelling. Part 2 (Chapters 4-7) includes exploratory data analyses, pre-processing and regression modelling. The material is organised as follows, with the relevant research sub-question number expressed in parentheses:

- **Chapter 2: RCF** introduces the principles behind the RCF phenomena, the key drivers and typical mitigation practices. There is a particular focus on the techniques of grinding and milling, and eddy current technology for detection (Q1-2),
- **Chapter 3: RCF Modelling** discusses existing RCF modelling techniques, including physics-based and data-driven models in the literature. The chapter concludes with the data-driven methodology and models being proposed (Q3),
- **Chapter 4: Experimental Data** introduces the data sources central to this thesis, alongside contextual exploration of patterns and relationships in the data, and identification of the most appropriate methods for processing and formulating the data for modelling (Q5),
- **Chapter 5: Data Pre-Processing** presents a methodology for processing eddy current data and integrating other relevant data sources to generate a modelling dataset for use in subsequent analysis and regression modelling (Q4),
- **Chapter 6: Practical Applications of Data Analysis** provides a simple approach for segmenting the track for preventive grinding and outputs that can be used in risk assessments for corrective grinding. The methodology is based on analyses of the processed EC data (Q5),

- **Chapter 7: Regression Analysis with Machine Learning Algorithms** A selection of ML algorithms are trained using the fully processed and integrated modelling data set with various parameter settings. The full results are presented and the proposed model is selected. This model is compared with the physics-based approach, Track-Ex (Q6-8).
- **Chapter 8: Summary** The conclusions, contributions and recommendations are put forward in Chapter 8.

Part I

Literature

Chapter 2

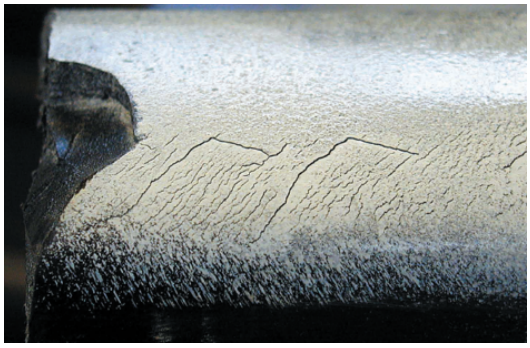
Rolling Contact Fatigue

In common with all transport systems, railways contain elements that must be continuously maintained. In particular, as the primary interface between the vehicles and track system, the rails are subject to a harsh operating environment with complex and variable forces. Wear and rail defects are thus an inevitable consequence of these conditions where the wheel and rail come into contact. A specific example is metal fatigue, which is a condition that arises from the repeated cyclic loading of railway vehicles on the rail. Fatigue is the progressive, localised , permanent structural change that occurs in materials subjected to fluctuating stresses and strains that may result in cracks or fracture after a sufficient number of fluctuations (Boyer, 1986). Fatigue in rails is characterised by minor cracking at the wheel-rail interface which may propagate due to sheer and bending stresses and ultimately lead to rail failure (Cannon et al., 2003).

The defects that lead to fatigue failure are known collectively as Rolling Contact Fatigue (RCF). Sub-surface initiated shells, surface initiated squats and head checking all fall into this category, and are illustrated in Figure 2.1. The various terms for different types of RCF can generally be characterised by their appearance and location on the rail as depicted in Figure 2.2.

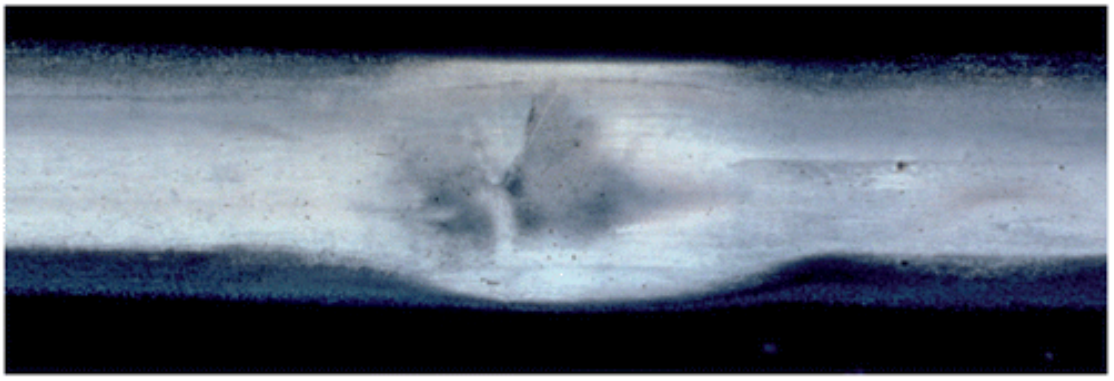
2.1 RCF Principles

Fatigue at the wheel-rail interface is distinguished by the high stress field produced by the wheel-rail contact and cyclic contact stresses which are largely compressive due to high axle loads (Fletcher et al., 2009). These stresses can cause localised areas of damage (or cracking). The propensity and frequency of such cracks in the rail depend on operational and environmental conditions such as track condition, track curvature, traffic loading, vehicle type, maintenance interventions and climatic conditions. This



(A) Headchecks (Network Rail, 2021a)

(B) Spalling



(C) Squats (Grassie, 2005)

FIGURE 2.1: Types of Rolling Contact Fatigue Defects

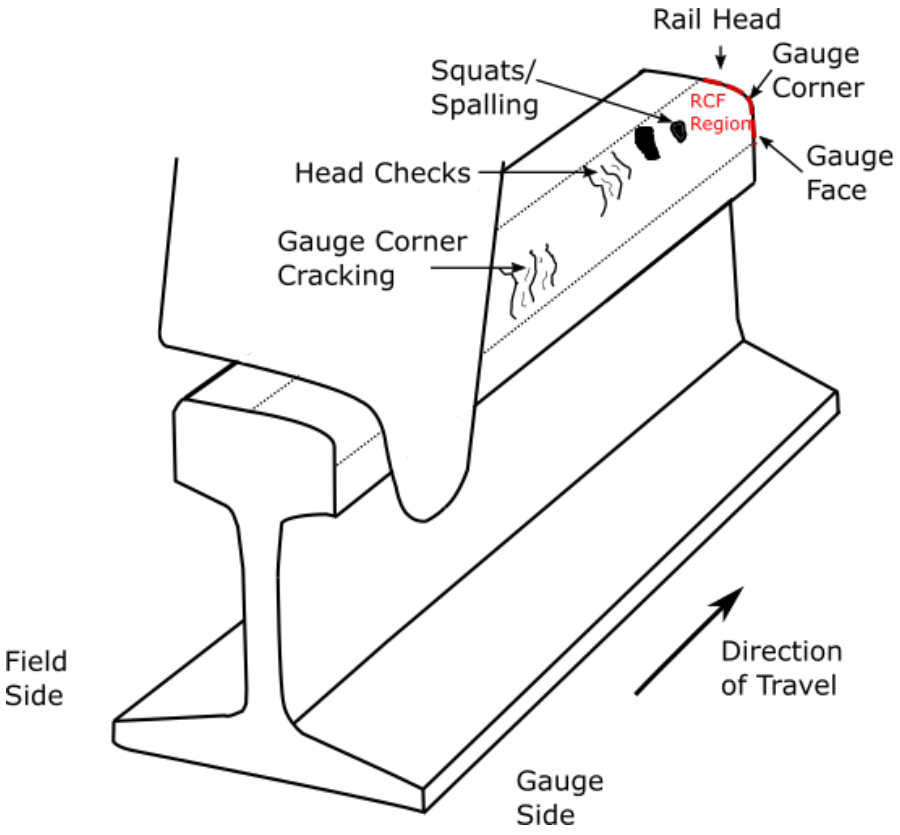


FIGURE 2.2: Rail Section with RCF Defects

study focusses on a particular type of *surface initiated* cracking typically developed as a consequence of frictional rolling/ sliding contact that causes plastic flow of the surface of the material. The initiation of such cracks are not normally associated with any specific metallurgical, mechanical or thermal fault, but as a result of the rail material's inability to sustain the imposed operating conditions indefinitely (Cannon et al., 2003). As the plastic deformation exceeds the fracture strain (or fracture point) of the material, a surface crack is formed.

There are believed to be two main causes for surface initiated RCF; either due to rolling contact accompanied with load and surface roughness; thus resulting in local stress concentrations, or if the rolling contact is accompanied by interfacial shear and slip (due to curving, braking and traction) which can result in plastic deformation of the contacting surfaces and subsequent crack initiation and growth (Ekberg et al., 2014). Further discussion on the generation of high frictional creep forces when steering, braking and accelerating are described in Appendix A.

The process of RCF crack initiation and propagation can be summarised in three distinct stages and are illustrated in Figure 2.3 (Burstow et al., 2009):

1. **Phase 1: Crack Initiation/ Early Crack Growth** Repeated cycles of rolling/ sliding contact between metals (such as the steel used in modern rails) instigate a process known as 'ratchetting' or 'cyclic creep'. Ratchetting (illustrated in Figure 2.4) is the incremental accumulation of plastic strain, whereby the loads that the material is subjected to are repeatedly above the yield of the material. Eventually this process causes the material to fracture generating small surface cracks at the head of the rail (Kapoor et al., 2002; Magel et al., 2004). This process is estimated to take place over a period of 3-6 MGT (Magel et al., 2003).

2. **Phase 2: Shallow Angle Crack Growth**

   There are several mechanisms under which cracks grow into rails: Firstly where shallow angle crack growth is driven by shear stresses which occur at the wheel/ rail interface. In this case cracks typically propagate at an angle of 15 degrees to the head of the rail and in the same direction of travel (traction direction). Crack growth then slows as the distance from the surface stresses increases some 10-15 mm below the surface (Smith, 2002). In other instances, the presence of fluids such as water or grease based lubricants play a crucial role (Shaulk, 2016):

   - **Hydraulic crack growth:** whereby the fluid is trapped in the crack and when the wheel drives over the crack it causes pressure on the fluid, driving crack growth,
   - **The fluid entrapment crack growth mechanism** is the mechanism where the crack closes when the wheel passes over it and the fluid is entrapped

and pressurised. This causes high pressure at the crack faces and tensile stresses at the tip of the crack causes the growth (Dollevoet, 2010),

- **The squeezed film crack growth mechanism:** a fluid is trapped in the crack and growth in the direction of the load motion, the crack mouth opens under tractive effort, drawing the fluid in where it expands under pressure as the wheel passes over and the crack is closed (Bogdański et al., 1999).

3. **Phase 3: Branching Crack Growth** At this stage the growth rate of a crack accelerates rapidly until failure or intervention. The growth is driven by by bulk bending and axial stresses in the rail due to wheel loads and residual stresses in the rails from manufacturing processes. Eventually this causes the majority of cracks to branch upwards, leading to surface shelling, with chunks of material spalling from the rail surface. Alternatively, in some cases the cracks will turn downwards and may develop into a transverse defect and ultimately a broken rail (Magel et al., 2003).

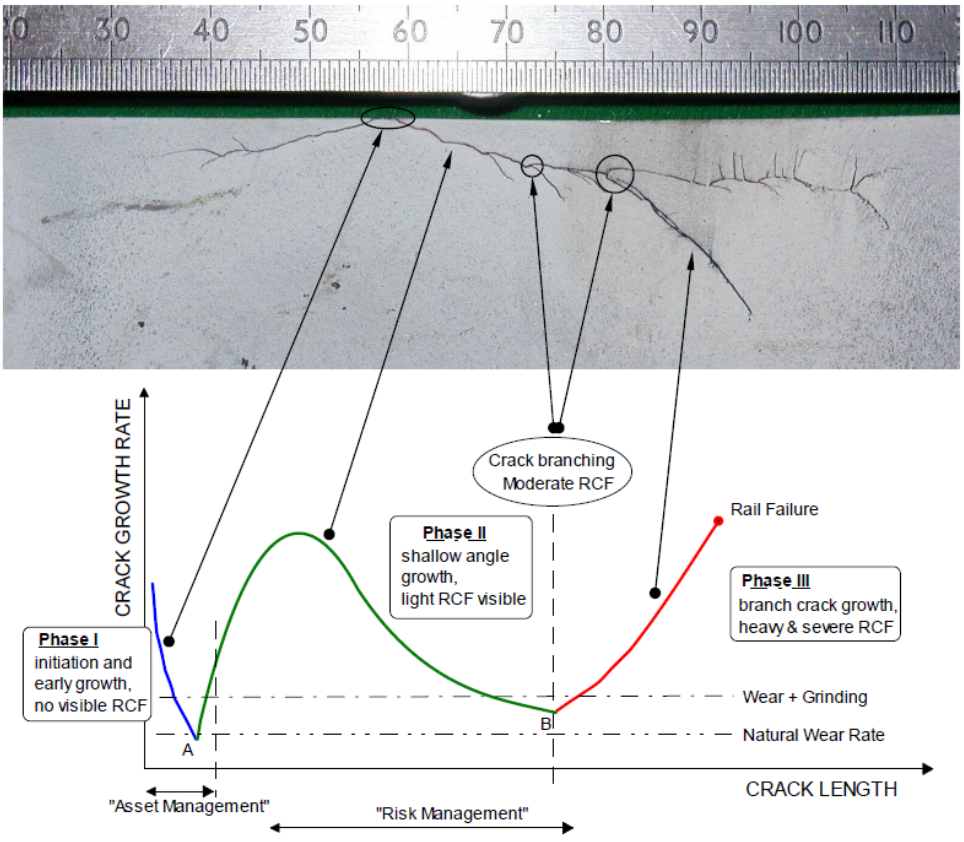


FIGURE 2.3: Phases of RCF crack development and growth (Burstow et al., 2009)

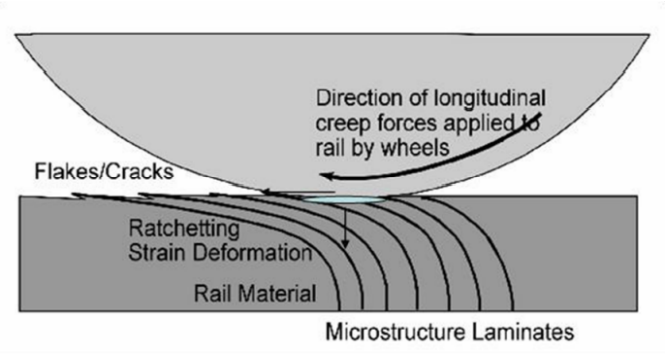


FIGURE 2.4: Illustration of Ratchetting

2.1.1 Wear

The same shear stresses caused by loading forces, traction, slippage and friction at the wheel/rail contact almost always cause wear in addition to Rolling Contact Fatigue (Magel et al., 2014). Wear is the loss or displacement of material from a contacting surface, and has historically determined the life of the rail due to the gradual loss of rail cross section ultimately resulting in replacement. Today, however, due to improved wear resistance of wheel and rail materials, rail life is now dominated by RCF damage (head checks and squats) (Schmid, 2010).

In fact, these processes are often competing, since the process of crack initiation and propagation are impacted by the rate at which material is worn from the surface. In systems with high wear rates, it is possible to wear away the stressed surface layer before it can develop cracks, or additionally to delay crack propagation by ensuring that the cracks do not reach a critical size for branching. However, high wear rates necessarily reduce component life, and therefore an optimum level of wear where surface fatigue is controlled and component life is maximised is sought. This optimum metal removal rate is known as the 'Magic Wear Rate' (MWR), which is "the rate of wear at which any rolling contact fatigue cracks that are in the initial stages of development are removed either by natural or combination of natural and artificial wear" (Magel et al., 2014). In most systems, due to the quality of the rail steels, the natural wear rates are insufficient to manage crack growth. In these cases grinding and milling can be used to artificially modify the wear rate of the material and control the growth of cracking. Although different rail grinding strategies have evolved over the years, it is generally agreed that the ideal approach is to grind the rail preventively at the MWR (Magel et al., 2014). In North America preventive grinding of rails, alongside improved metallurgies, contributed to a two-fold increase in system rail life and a four-fold increase in system rail fatigue life over the last 25 years. (Magel et al., 2005)

Figure 2.5 illustrates the combination of natural wear and minimum artificial wear (grinding) that are needed to achieve the MWR. The frequency of grinding

intervention needed to achieve this target varies depending on the specific conditions contributing to crack initiation and propagation at the wheel-rail contact.

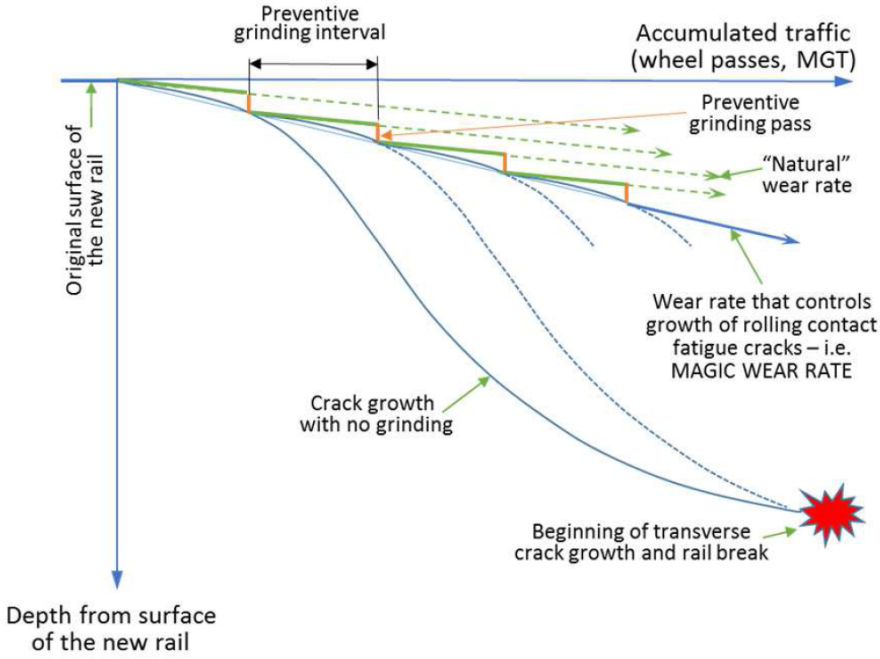


FIGURE 2.5: Illustration of relationship between contact fatigue growth, natural wear and Magic Wear Rates (Magel et al., 2014)

2.2 RCF Management Overview

2.2.1 Managing RCF

In recent years, largely due to the rise in axle loads, traffic density, speed and tractive forces, and improved wear resistance of modern rail materials, infrastructure managers have seen an increase in the occurrence and severity of RCF. There are significant implications with regard to safety and cost to the IM, which has resulted in numerous global efforts to develop effective management strategies.

These strategies vary between locations and Infrastructure Managers (IMs) since they depend on operational conditions, but they can broadly be considered as design or maintenance strategies applied to the vehicle or track side of the Wheel-Rail Interface (WRI) (Magel et al., 2005):

- **WRI Design:**
  - **Wheel/ Rail Metallurgy:**

Rail materials are chosen in order to withstand the high static and dynamic forces present at the wheel-rail interface. Modern rails are typically composed of very high-quality steel alloys featuring a high percentage of carbon (up to 0.7%) as compared with everyday construction steels (Schmid, 2010). The most common rail material in Europe is R260 (previously named UIC 900A) which has 0.62% carbon and 0.7-1.2 % manganese. International specifications divide rail treatments into: 1) naturally hard rail grades and 2) hardened rail grades. The head hardened rail grades were developed to improve wear resistance and strength. However, increasing the wear-resistance of modern day rail steels has been linked to the rise in RCF damage as a result of reduction in natural wear that has assisted in managing RCF. Similarly, wheels are made of carbon steel/ manganese alloy, which is normally obtained by processing and continuous casting of scrap steel.

– **Wheel/ Rail Profile Design:**

The design of wheel and rail profiles are of central importance in railway design since they govern the vehicle dynamic behaviour, the forces between the wheel and rail, and the stresses imposed at either side of the interface. Figure 2.7 illustrates a typical rail and wheel profile in a new(or restored) condition, and also in a worn condition. Commonly used terms to describe different elements of the wheel and rail profiles are provided in Figure 2.7.

Self steering is achieved through the coned shape of the wheel tread for a given rail profile combined with solid axles. The *conicity* of a wheel tread is defined as the tangent of the half angle γ subtended at the apex of the cone (See Figure 2.6), the cone angle is selected to enable self steering on a selected rail profile at given inclination. The rails are inclined inwards to ensure the wheel-contact is centred at the rail crown; the strongest part of the rail, and additionally to assist with self-steering (in the UK inclination is 1 in 20). The exact parameters related to the wheel/ rail profiles, wheel conicity and rail inclination however are not considered in isolation, rather the correct, or optimal combination of these features is of primary importance.

Over time, as the wheels pass over the rails, the profiles may vary as a result of wear, both wheel and rail wear patterns will vary depending on the traffic type and the route characteristics. The illustration shows extreme cases of wear for both the wheel and rail. Wheel flange wear often occurs as a result of vehicle with a stiff primary suspension or where the route has a high proportion of tight curves, whereas tread wear tend to occur on a route composed of tangent track. In either case the result is a modification in the wheel conicity which affects vehicle stability, curving performance

and contact forces. For the rail on the other hand, predominantly straight track results in head-wear, whilst sharp curves tend to result in side wear.

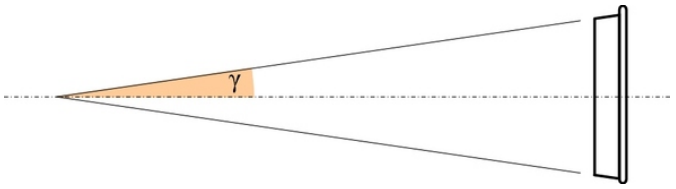


FIGURE 2.6: Wheel Conicity

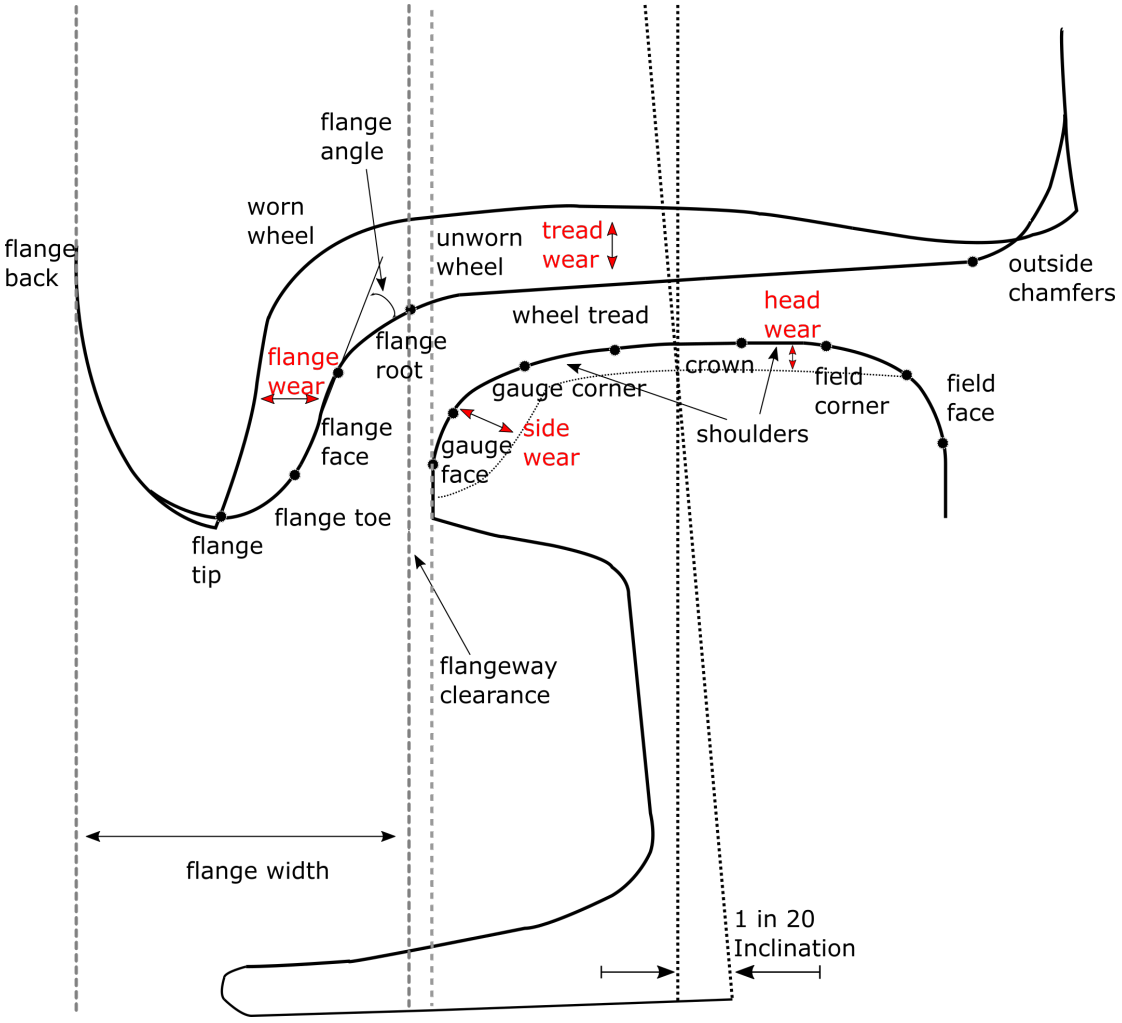


FIGURE 2.7: The Wheel/Rail Interface adapted from (Lewis, 2009)

– **Suspension Bogies:** Flexible bogie suspension improves the steering capability of the running gear. As described in Appendix A, the reduction in the bogie primary yaw stiffness can reduce yaw angles in curves which will reduce lateral creep forces required to keep the vehicle in radial alignment, and in turn reduce RCF initiation on the rails of shallow curves and in wheels.

- **WRI Maintenance:**

- **WRI Monitoring:** Improved and/ or more frequent defect detection can reduce risk of rail failure. Techniques such as Eddy Current testing have been developed in order to detect cracks in their early stages and therefore a greater opportunity to monitor crack growth and intervene where necessary,
- **Friction Management:**

  Wear and damage is also influenced by the frictional characteristics of the interface. Too high a frictional value, results in noise, increased energy consumption, wear and damage to components, whereas too low a value will result in lack of traction, reduced braking and wheel damage. Typical values appear to range between 0.08 and 0.5 (Lewis, 2009). Lubrication of the rail gauge-face/ wheel-flange can reduce wear by 95-100%, however water entrapment has a significant influence on shallow growth through fluid entrapment, and grease based lubrication may also influence crack growth via this mechanism.
- **Grinding/ Milling:**

  Grinding is a powerful tool for both preventing RCF development and removing existing RCF (alongside milling), it further removes surface damage to improve ultrasonic detection (Cannon et al., 2003). It is one of the main strategies adopted by railways to combat RCF and is discussed further in Section 2.4,
- **Track Quality:** Vertical and lateral track irregularities are often associated with rapid rates of RCF development in rail adjacent to the irregularity. (Attoh-Okine, 2016; Zarembski et al., 2016)

2.3 RCF Monitoring

Over the years, various systems have been developed to monitor the health of rails, and substantial efforts are still being made in this field in order to improve detection capabilities. Non Destructive Evaluation (NDE) methods are widely applied for the inspection of rails since they can be carried out in-situ and in many situations the equipment can be installed on trains and testing vehicles to minimise network disruption. We focus on the methods in use at Network Rail for RCF detection, for a more in-depth overview of other Non Destructive Testing (NDT) techniques in use, see (Bombarda et al., 2021; InnoTrack, 2008).

2.3.1 Ultrasonic Testing (UT)

Ultrasonic Testing (UT) technology is one of the most common methods applied in the fields of rail crack detection, and is particularly effective at detecting internal cracks (Xu et al., 2020). The technique uses high frequency beams of sound energy which are transmitted into the rail, and the reflected or scattered energy of the transmitted beam is then detected using a collection of transducers. The amplitude of any reflections together with when they occur in time can provide valuable information about the integrity of the rail. Since there is uncertainty in the direction and location of defects, the energy is transmitted at several different incident angles in order to maximise the Probability of Detection (PoD) of any detrimental features present in the rail (Papaelias et al., 2008).

In many countries, this technique is typically employed on Sperry trains (UTU1 and UTU2 models). The presence of detected defects by the UTU1 and UTU2 Sperry trains is then confirmed through the deployment of portable ultrasonic inspection units known as Sperry Sticks (InnoTrack, 2008). Verification of identified defects is required due to the relatively high occurrence of false alarms. Ultrasonic Detection relies on the fine-tuning of the test equipment by means of parameters such as the signal threshold (amplitude of the signal) and the position of the time window (or acquisition time). If the threshold is set too high then the system will miss cracks whilst if it set too low it will generate many false alarms.

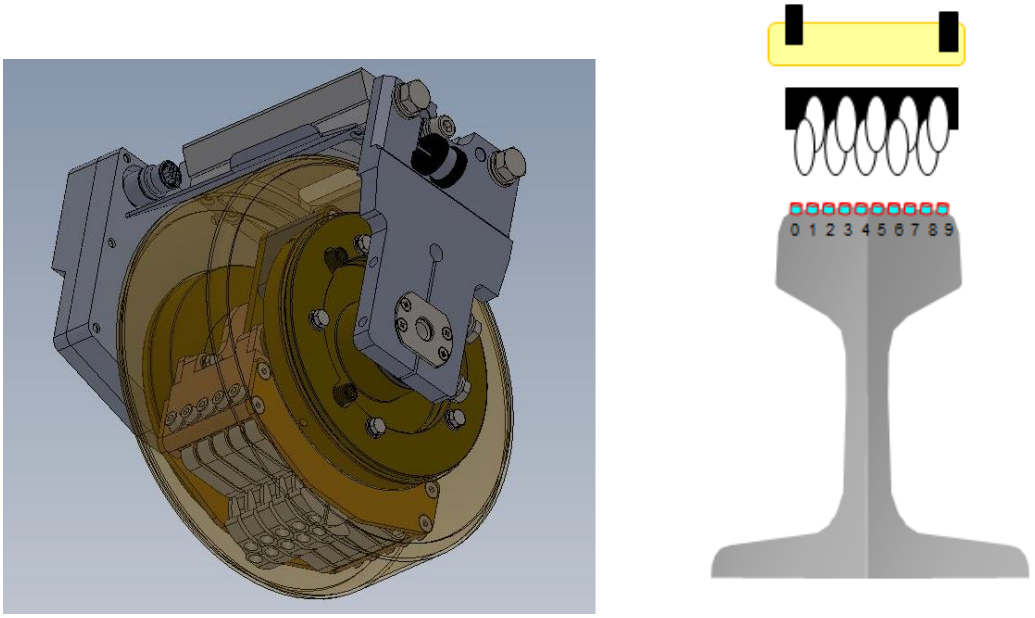
Additionally UT transducers must be in close contact with the track to achieve ultrasonic transmission which limits the possible speeds of inspection. Another more serious drawback of the technique is a phenomena known as 'shadowing' (Peterson, 2000; Cannon et al., 2003). In cases where there is shelling, or detail fractures on the surface of the rail, the equipment may not be able to detect more serious sub-surface cracks. Generally it is known that ultrasonic testing performs well in detecting deep surface-braking and internal defects, however is less effective in detecting small ($<$4mm) surface defects such as head checks and Gauge Corner Cracking (GCC) (Papaelias et al., 2008). Technology such as pulsed Eddy Current sensors have since been developed to detect these types of defects since they have high sensitivity to rail surface cracking (InnoTrack, 2008).

2.3.2 Eddy Current Testing (ECT)

Eddy Current (EC) technology is currently one of the most effective ways of detecting rolling contact fatigue across the railway network due to its high speed detection capabilities and sensitivity to even the smallest near surface defects. At Network Rail, Sperry EC technology are fitted to the rail inspection vehicle (Ultrasonic Test Units) or pedestrian sticks. A pilot scheme was established in 2015 to replace the current visual

inspection for RCF as eddy current technology provides a more reliable, accurate and consistent means of identifying RCF. The Ultrasonic Test Units (UTUs) will provide eddy current data along the route that the train takes, but the train borne inspection process is not approved for use in S&C, and so pedestrian EC testing can be used to record accurate data within these areas.

The Sperry Surface Crack Inspection System houses an eddy current roller search unit (RSU), which contains ten sensors spaced across the width of the running surface of the rail (illustrated in Figure 2.8), and is pressed down to ensure contact with the rail surface. The sensors; known as probes or coils, are positioned at a predetermined distance (*lift-off*) above the rail, increasing or decreasing the lift-off distance impacts the system's sensitivity and penetration capability. High sensitivity of the system can lead to indications of deep damage from an array of very small cracks, which leads to overestimation of damage severity. On the other hand, the depth of penetration is of major importance, since cracks over a given size cannot be reliably sized (in the Sperry System the maximum depth of detection is 5mm). Variation of these parameters considerably effects the eddy current response, and thus must be considered when evaluating the reliability of crack sizing.



(A) Sperry Eddy Current Roller Search Unit (RSU)

(B) Sperry Eddy Current Probe Configuration (Whitney, 2020)

FIGURE 2.8: Illustration of Eddy Current Measurement System

Eddy Current testing is based on the principles of electromagnetic induction, which are illustrated in Figure 2.9 (Bhagi, 2011). A primary magnetic field is generated when alternating current is injected into an induction coil. When the coil is placed over a conductive sample, Eddy currents are induced in the material. Those Eddy Currents generate a secondary magnetic field opposed towards the primary field. Any changes

in the conductivity of the material being examined, such as near-surface defects or differences in thickness, will affect the magnitude of the eddy current. This change, or more specifically the coil impedance is detected using either the primary coil or the secondary detector coil, forming the basis of the eddy current testing inspection technique (Hwang et al., 2015). Crack lengths are calibrated prior to inspection using a test setup that mimics eddy current rail inspections as closely as possible.

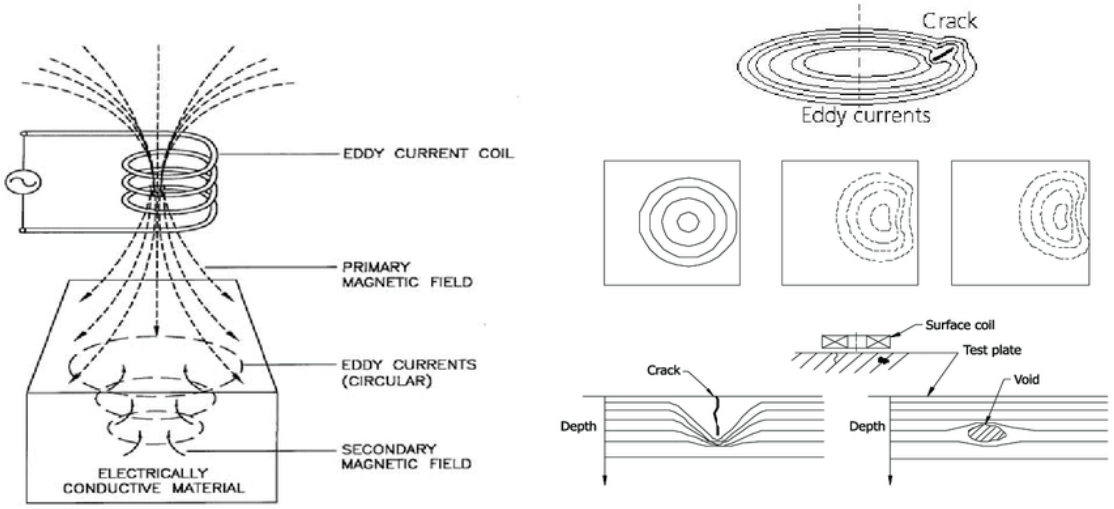


FIGURE 2.9: Principle of Eddy Current Testing(left) and distortion of eddy current due to crack, edge-effect, surface crack, and sub-surface void (right)., (Bhagi, 2011)

2.4 RCF Re-profiling

Rail grinding and milling are used in the rail industry to maintain the safety and quality of the track by removing layers of the rail surface. Grinding is the most common type of material cutting and surface generation process, which typically uses an abrasive method for removing the rail material (Zhu et al., 2019). The metal is cut through the action of a rotating grinding wheel including rubbing, plastic deformation, cutting and chip formation (Zhang et al., 2020) as illustrated in Figure 2.10 [1]. Though it is this mechanism of material removal that has raised some concerns due to the levels of heat generated which may result in rail pre-fatigue or burn of the rail surface (Yuan et al., 2021). In particular, Gu et al. (2015) found that removal rate and surface quality is improved through the elevation of grinding rotational speed, however the grinding temperature and hardness of rail will also increase, causing surface burn to the rail.

In contrast, milling (shown in Figure 2.11) is a relatively new technique used in the railway industry and can be described as a dry rotational cutting process. The method

[1] other types of grinding are oscillating grinding, or high-speed grinding (HSG), see (British Standards Institution, 2018; Popović et al., 2022)

is capable of a much higher level of geometrical accuracy than grinding and can also remove several millimetre defects in only one pass (Kubin et al., 2019).

The value of these techniques are two-fold, firstly grinding (or milling) can remove shallow cracks, or prevent them from propagating. For example, if grinding is performed frequently enough to stall cracks in their early stages of growth (Phase 2), this reduces the chances of cracks turning down into the rail. Secondly, by re-profiling the rail transversely, grinding can restore the rail to its original profile to improve contact conditions and vehicle dynamics. The original wheel/ rail profiles are designed such that contact stresses are minimised where possible and can move wheel contact to areas of the rail less susceptible to RCF. Typically we divide grinding philosophies adopted by IMs as *preventive* and *corrective* (Grassie, 2005), these approaches are described in the following sub-sections, which include the current methods in use at Network Rail.

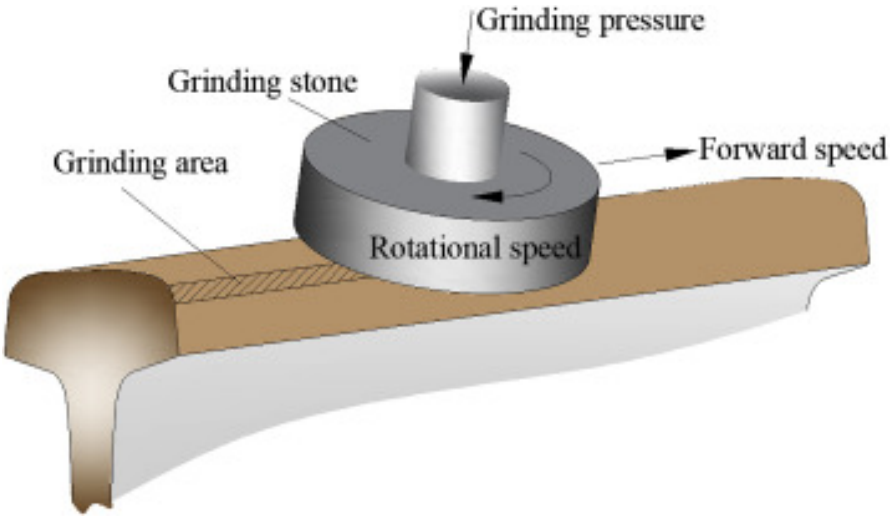


FIGURE 2.10: Rotational Grinding of the rail (Zhou et al., 2019)

Specifically, Network Rail have a range of machinery available to use for rail re-profiling depending on the particular application. For example, the Loram C21 series have a high output and are used for Single Pass (SP) preventive grinding. In order to target specific rail head defects, corrective grinding is performed using the Loram SPML series of Speno RPS32 which enable Multi-Pass (MP) grinding. Currently milling is performed using a Strabag leased Road-Rail Vehicle (RRV) that is not compatible with UK signalling systems and therefore requires mounting on two locomotives, this results in reduced productivity due to logistics and access restrictions. NR are in the process of acquiring a Schwearbau milling train which has significantly greater cost per shift but superior capability. The fleet also includes specific Switches and Crossings (SC) grinding equipment (Harsco), that are used where it is not possible to manoeuvre the plain line machinery. Table 2.1 demonstrates the relative capabilities of each vehicle model, note that the costs calculated are approximate.

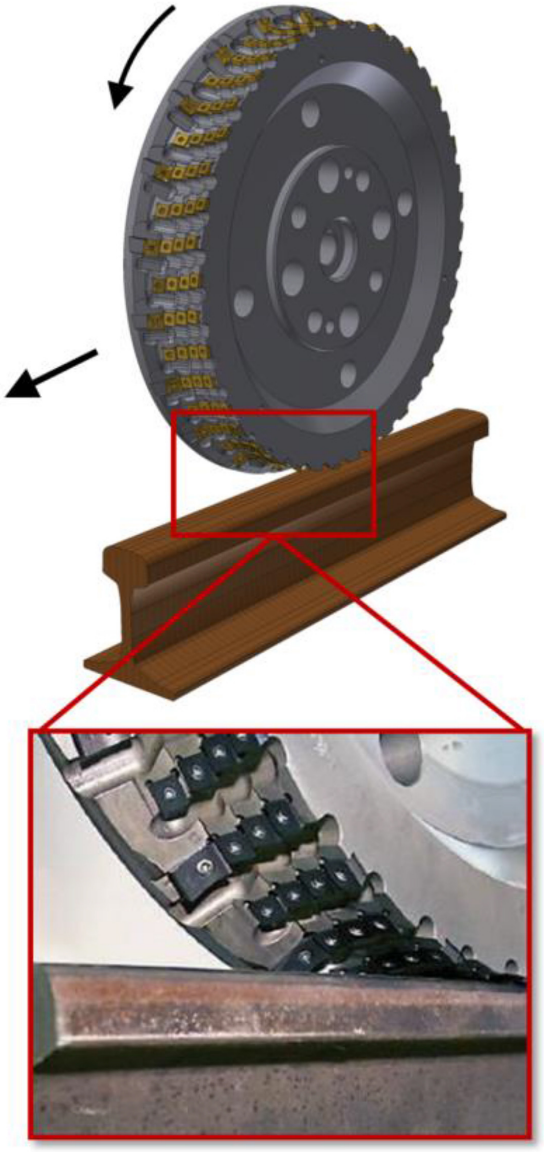


FIGURE 2.11: Rail Milling Wheel (Kubin et al., 2019)

TABLE 2.1: Network Rail Intervention Equipment Capabilities

Model	Removal Depth per pass (mm)	Speed per pass (m/hr)	Cost per shift (£)	Purpose
Strabag RRV Miller	0.75	420	17500	Milling
Schwearbau Miller	5	350	35000	Milling
Loram C21 Series (PL)	0.2	12872	18626	SP Grinding
Speno RPS32 (PL)	0.1	4022.5	12373.51	MP Grinding
Loram SPML Series (PL)	0.1	4022.5	12373.51	MP Grinding
Harsco (SC)	-	-	-	SC Grinding

TABLE 2.2: Current Best Practice Single-Pass Preventive Grinding Intervals (Magel et al., 2014)

Curvature	Rail Grade	Preventive Grinding Interval
Sharp	Premium	14-23 (15-25MGT)
Mild	Premium	27-45 (30-50 MGT)
Tangent	Intermediate	40-54 (45-60MGT)
Tangent	Premium	91 (100MGT)

2.4.1 Preventive Action

Preventive rail grinding is a process of controlled artificial wear which seeks to restore desired rail profiles and achieve the required depth of metal removal with minimal grinding effort and steel wastage (Magel et al., 2003). In particular, and as already noted, we often reference a concept known as the 'Magic Wear Rate', which is the desired optimal metal removal rate achieved through the combination of natural and artificial wear required to remove existing minor cracks and control crack growth. Crucially here the objective is to prevent rail damage rather than correct for it, and when applied effectively can be performed with a single pass of the grinding train, and typically removes less metal than corrective actions. Preventive grinding is generally understood to greatly improve logistics due to fewer passes and higher machinery speeds, and is the most economical grinding strategy.

Numerous practices, particularly regarding how often to grind and how much metal to remove at each cycle have been evolving since the late 1960s. Whilst initially developed for removing corrugation in the rail, eventually philosophies of regular, tonnage-based grinding emerged and established the process as being preventive rather than corrective.

Kalousek et al. (1989) first proposed the use of a 'Preventive Rail Grinding Strategy' where the grinding cycles necessary to remove all initiating surface cracks are defined. The idea was tested by Canadian Pacific Railway (CPR), by grinding up to 6 times per year (i.e. at 10 MGT intervals), and field testing proved that this method would control cracking and was considerably more economical than multi-pass grinding strategies (Magel et al., 2003). Additionally in 1991, the Burlington Northern Santa Fe (BNSF) line applied this strategy whereby grinding at intervals of 18-40 MGT were introduced on curves and an interval of 35-60 MGT on tangent track (Magel et al., 2003). The current single-pass preventive grinding intervals according to Zakharov et al. (2001) are summarised in Table 2.2.

Additionally other strategies have been developed where there is heavy RCF on the rail to start with and a more gradual approach to move towards a preventive strategy is required, example of such programs are the 'Preventive-Gradual Grinding Strategy' (Stanford, 2000) and the 'Predictive-preventive' (Harris et al., 2011).

Outside of North America, Heyder (2014) carried out empirical studies on the propagation of headchecks in the German railway network to better plan grinding intervals. Zoeteman et al. (2014) studied the effect of various rail grinding regimes over a period of 13 years on the Dutchrail network and showed that the interval chosen for the grinding interventions is very crucial since the cracks grow at much higher rates at the later stages of propagation.

2.4.1.1 Preventive Grinding at NR

Alongside other European IMs; and largely based on principles suggested in studies conducted by the NRC on North American railway lines, Network Rail implement the following cycles to re-profile rails with standard steel grad R260 to control HC defects: a) after 15 MGT in curves, b) after 45 MGT in the tangent track (British Standards Institution, 2018) (as illustrated in Figure 2.12). Depending on the track and traffic characteristics, these cycles can be doubled for harder heat-treated rails (R350HT).

According to a senior RAM at NR this approach was introduced at NR circa 2004, and has not been modified since. The implementation of the process is predominantly manual since it relies on a look up being performed for each track section against a curve register (which is currently performed by hand), and the latest information for traffic is obtained. The grinding frequency is thus set according to the track curvature and traffic for that section. Practically speaking, the process is not entirely precise due to the logistics such as the inability to handle annual tonnages that are not constant across each subdivision and staying on cycle on routes where sharp curves are miles apart whereas the majority of sections are covered by mild curves (Harris et al., 2011). It is also believed that the current grinding equipment may not be producing the desired profiles which may prove the practice less effective, and it is not always known whether the RCF cracks have been completely removed when grinding is finished.

Whilst this approach is simple, and relatively straightforward to apply, it is based on empirical data from studies completed outside of the UK network and assuming heavy haul conditions. With the advance of monitoring capabilities and new sources of data for detecting RCF severity, we propose that grinding intervals should be informed using analysis of operational data and in particular the historical Eddy Current Data. Instead of all curves and tangent sections being considered equivalent in terms of their risk, the data could help to better categorise these sections.

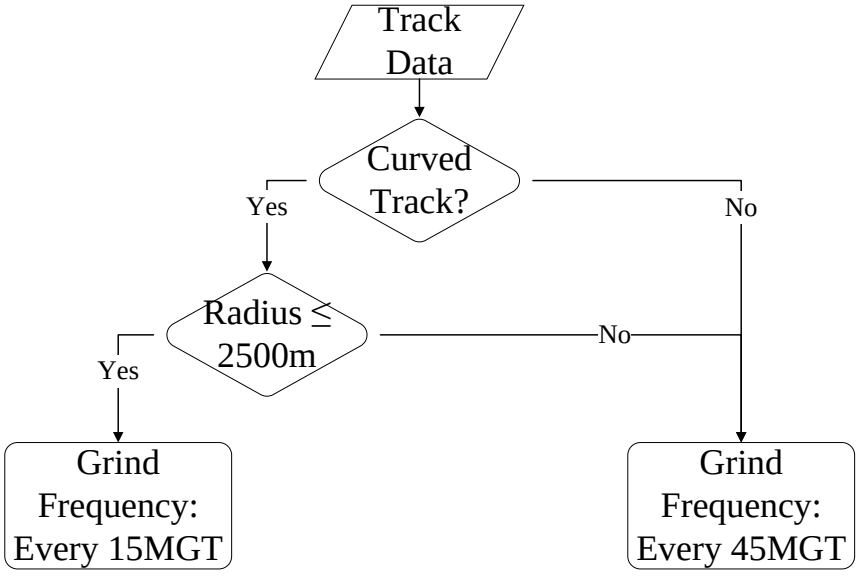


FIGURE 2.12: Current Preventive Grinding Strategy

2.4.2 Corrective Action

In the case of corrective methods, individual RCF cracks or sections of RCF are detected and treated before there is a substantial risk of the rail breaking. Corrective grinding typically involves several passes of the grinding machinery to remove rail damage by RCF. Corrective action includes the detection by ultrasonic means of defects that have developed from RCF cracks, as well as the technique pioneered in UK since late 2000 of classification by visual means (Grassie, 2005).

2.4.2.1 Corrective Grinding at NR

Formally the policy for treating RCF is in adherence with Standard NR/L2/TRK/001/mod07 (Network Rail, 2018a), which assigns Minimum Action Code (MAC) to RCF sites depending on the most recent EC measurements and the track category. Minimum Action Codes define the maximum period in which the intervention work must occur and may further impose speed restrictions where the risk of rail failure is perceived to be high.

However, some routes found the guidance to be overly conservative, generating huge numbers of sites which required remedial actions, speed restrictions and cause considerable network disruption. Anglia have thus proposed an alternative risk-based approach which is illustrated in Figure 2.13 (Wilson, 2018). At the beginning of each year the routes are assigned a number of milling and corrective grinding shifts that can be used and they are responsible for determining which areas are prioritised to

maximise productivity and minimise risk. The process set out in the following sub-sections is intended to determine which sites can be milled and which sites must be replaced and is applied only to the highest criticality sites: Heavy (H), Severe (S) or Very Severe (VS). The procedure outputs the sites to be milled and the sites for re-railing, and these sites are subsequently input to a planning procedure to acquire the appropriate track possessions.

Light (L) and Moderate (M) sites are considered instead for corrective grinding where fewer passes may be applied, in this case a risk assessment is not applied and all sites are put forward for corrective grinding.

1. **Capture Data:** The engineer completing the risk assessment must manually gather all the relevant data sources for each track length across the route. The data required includes:
    - EC Data: Most recent 22 yard site data are extracted from the Rail Defect Management System (RDMS),
    - Wear Data: vertical and side wear data,
    - Defect Data: Latest UTU Run, defines the number defects over the site,
    - Track Data: Curvature, track type, rail age, UTU inspection frequency,
    - Traffic Data: Freight/ Passenger Traffic,
    - Track Geometry Quality: 220 yard standard deviation data,
2. **Data Processing:**
    - Filter Sites: Retain only Heavy (H), Severe (S) and Very Severe (VS) EC Sites,
    - Integrate different data sources: Aggregate the supporting sources over each EC Site,
3. **Site Risk Assessment:**
    - **Calculate Site Inspection Score:** this score is based on the defect data, track geometry data and vertical and side wear,
    - **Calculate Track Risk Score:** this score is based on the line speed, curvature, tonnage, UTU frequency and rail age,
    - **Calculate the Overall Risk Score:** the final score is a weighted average of the above parameters, the weightings are defined by track experts,
    - **Determine Minimum Action Code (MAC) for Site:**, and the site scores are assigned to a Minimum Action Code (MAC) in accordance with Table2.3, these mappings are defined by track experts,

4. **Decision:** Determine where to grind, mill or re-rail: The remaining rail is calculated, which is the (remaining depth - maximum crack depth):
   - If the remaining rail $>$ 1 mm: Mill,
   - If the remaining rail $\leq$ 1 mm,: Re-rail,

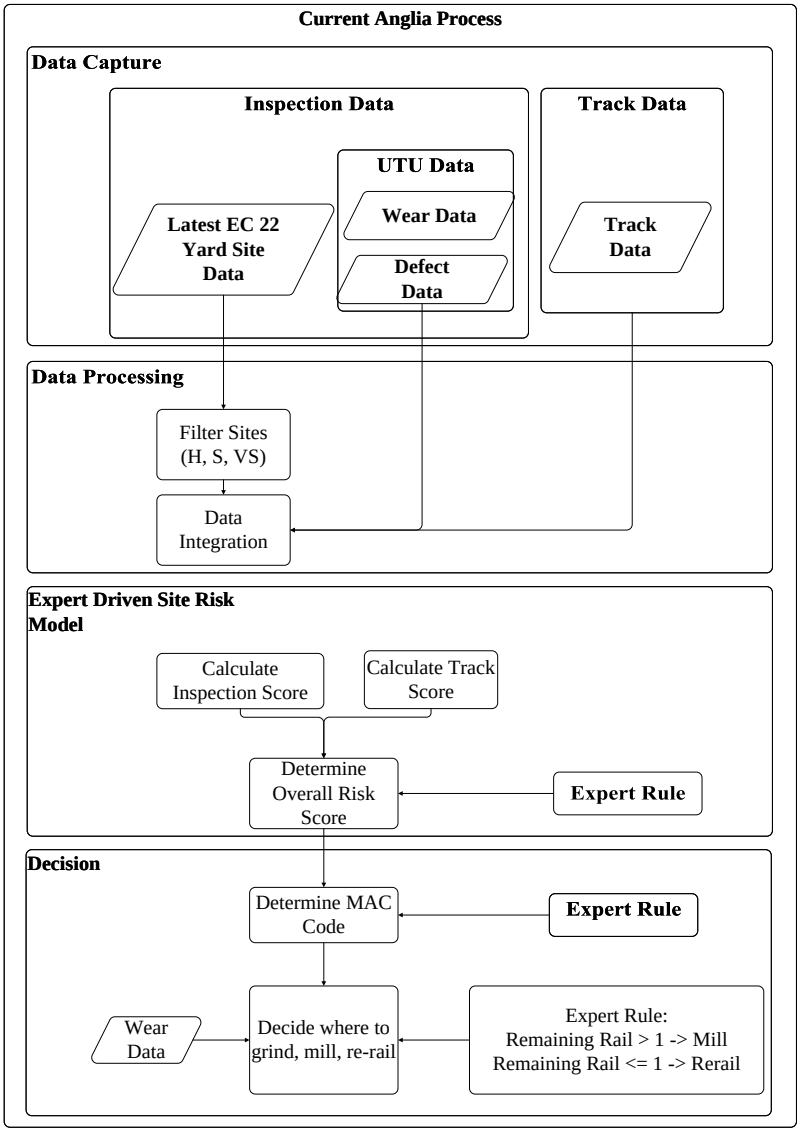


FIGURE 2.13: Flowchart for Current Anglia Corrective Maintenance Process (Derived from personal communication, Kevin Anderson- Network Rail Senior Asset Manager for Anglia, June 6, 2019)

This risk-based approach is largely a manual procedure, which requires gathering of a number of data sources, and additionally the reliance on track -experts to make judgements on RCF risk based on these factors (also determined through expert judgement). This thesis proposes an evidence based methodology for providing information to track experts regarding the historic Eddy Current data and rates of degradation with respect to rail damage and builds on the Anglia approach. This

TABLE 2.3: Minimum Actions to be taken based on Anglia Risk Assessment (Wilson, 2018)

Defect Depth	Defect Class	Minimum Action by Risk Score						
		60+	59-53	52-47	46-40	39-33	32-26	26-0
$0.1 \leq 1.5$mm	Light RCF	R26	R26	R26	R26	R52	R52	R52
$1.6 \leq 3$mm	Moderate RCF	R26	R26	R26	R26	R52	R52	R52
$3.1 \leq 4$mm	Heavy RCF	3K	3L	3L	3S	3S	3T	3T
$4.1 \leq 4.9$mm	Severe RCF	3K	3K	3L	3L	3L	3S	3S
≥ 5mm	Very Severe RCF	3C	3C	3K	3K	3L	3L	3L

TABLE 2.4: Minimum Action Codes from Table 28 (Network Rail, 2018a)

MAC	Minimum Action to be taken	Time Frame
R26	Retest (Visual & Ultrasonic)	every 26 weeks
R52	Retest (Visual & Ultrasonic)	every 52 weeks
3C	Remove defect	Within 13 weeks
3K	Remove defect	Within 26 weeks
3L	Remove defect	Within 52 weeks
3S	Remove defect	Within 2 years
3T	Remove defect	Within 3 years

could include simple processing of the Eddy Current data, or additionally the use of a forecasting model to determine future levels of RCF damage.

2.5 Concluding Remarks

In this chapter we have described the phenomena of RCF in rails and subsequently some techniques for managing it such as frequent monitoring, and rail re-profiling by grinding or milling. The approaches used to treat RCF are generally divided into preventive grinding or corrective grinding (or milling), and the approaches adopted by NR have been discussed.

The current preventive strategy used by Network Rail is based on the track curvature and tonnage passing over the track, and does not use RCF monitoring data to inform this method of segmenting the track. The corrective strategy on the other hand uses the latest Eddy Current data to determine where to intervene, however the basic guidance is overly cautious and often disruptive to asset managers. In response to this, Anglia have developed a risk assessment for prioritising the critical sites, which again uses the latest Eddy Current data in addition to other parameters derived using expert judgement. These approaches however do not employ information on real rail degradation rates, rather they rely on expert judgement.

Additionally it is observed that neither of the current strategies use any existing RCF modelling approaches to inform their decisions, despite the availability of a software tool known as Track-Ex which implements a simplified version of the WLRM. The following chapter describes this method, along with some of the most commonly used approaches for modelling RCF.

The proposals discussed in this research are aimed at augmenting these strategies by using data mining and modelling techniques to determine patterns in the RCF monitoring data (Eddy Current Data). In the case of preventive grinding, we focus on deriving a new method of assigning scores to tangent and curved track sections based on the measured data. With respect to the corrective strategy we propose a simple method for using pre-processed data to determine rates of degradation to highlight hotspots, and further to build a RCF prediction model using machine leaning algorithms.

Chapter 3

Modelling Techniques Applied to RCF Mitigation

Mathematical models can generally be characterised as physical (or mechanistic) or Data-driven (DD). The physical (or mechanistic) models are based on the physical laws describing the behaviour of the asset, whereas data-based modelling is based on the analysis of the data characterising the system under study (Solomatine, 2008).

This chapter focusses on RCF modelling, the predominant physics-based approaches, their uses and limitations and the absence of data-driven modelling in the field of modelling RCF for maintenance planning.

3.1 Physics Based Modelling Techniques

In recent years, numerous models have been developed for the prediction of RCF evolution, and owing to the physical complexity of the phenomena it is still a topic of extensive research. Predominantly, two different groups of RCF damage prediction models exist (Dirks et al., 2015), the models in the first group are able to predict successfully the probability of RCF for many wheel-rail contact conditions (Burstow, 2004; Ekberg et al., 2002), and the second group enable the calculation of actual crack growth through the use of fracture mechanics. Additionally the former category can be subdivided further, owing to more and less complex modelling of wheel/ rail contact, as described by (Krishna et al., 2021) we divide the models into Engineering models and Finite Element (FE) Models:

1. **Engineering Models:**
    - **Global Models:** In these models, the entire contact patch is considered to quantify damage, such models include an energy dissipation approach

known as the Whole Life Rail Model (WLRM) (Burstow, 2004) and methods based on Shakedown Theory (Ekberg et al., 2002). Notably, these models cannot predict the actual crack size, rather they are crack initiation models which consider the risk of surface initiated RCF. However, there are some works that have looked at combining these approached with fatigue-based models relating RCF damage parameters with actual crack length, and validating the results against Eddy Current Measurements (Rodríguez-Arana et al., 2021). Additionally, 'Finite Element Models' are able to predict crack growth through explicitly modelling the individual cracks.

- **Localised Models:** These models discretise the contact patch surface area into numerous elements and quantify the corresponding damage for each element. Models such as the 'Wedge' (Trummer et al., 2016) and 'KTK' models (Dirks et al., 2015) are considered to fall in this category. In comparison with fracture mechanics approaches, the computational demands are significantly reduced, enabling their incorporation with vehicle dynamics simulations.

2. **Finite Element Methods (FEM):** In this group of techniques, crack growth is explicitly modelled using fracture mechanics and detailed finite element analyses. Fracture mechanics approaches use the concept of configurational forces (Simha et al., 2008; Pletz et al., 2014) for crack growth predictions. Crack initiation criteria and crack growth models, embedding sophisticated FEM-based simulation models allow predictions of crack initiation and crack growth based on the complex elastoplastic stress-strain material response. However such approaches are not practical for railway maintenance planning, since they involve long computation times, so that only a few loading cycles at specific locations in the railway network can be calculated in practice (Trummer et al., 2016).

Whilst finite element approaches are the most complete in their treatment of wheel-rail contact and crack growth modelling, they are not practical since they involve such long computation times. To support maintenance planning of entire networks, it is necessary to be able to consider larger areas of track, and a range of operational conditions, rather than a few loading cycles at specific locations. Therefore, we limit our focus here to models in the first category, which are designed to be used in tandem with Multi Body Simulations (MBS) of vehicle dynamics to provide damage quantification for numerous operating conditions. The following sections will first describe the principle of Shakedown theory and the resulting damage models, followed by the energy dissipation approach known as the WLRM.

3.1.1 Shakedown Theory

In operation, rails are subject to continuous loading of wheel passages, under these circumstances the rail material is repeatedly loaded above the elastic limit of the material. The behaviour of the material at the wheel-rail contact points can be summarised using the theories of Elastic-Plastic Shakedown as described in Johnson (1985). These material responses are described below and summarised in Figure 3.1:

1. **Perfectly Elastic Response:** In the case of sufficiently small loads, there is a perfectly elastic response, such that the yield stress σ_y is never exceeded,
2. **Elastic Shakedown:** With increased loading, the yield strength σ_y is exceeded, resulting in a stabilised cyclic plastic flow allowing the material to 'shakedown' to a perfectly elastic response. This is possible below the elastic shakedown limit (σ_{EL}),
3. **Plastic Shakedown:** Once σ_{EL} is exceeded, plastic deformation may take place with every cycle of load such that a closed cycle of plastic strain is obtained (plastic shakedown) without any accumulated plastic deformation,
4. **Ratchetting:** Alternatively, the plastic cycle is an open loop, whereby repeated increments of unidirectional plastic strain are accumulated: a condition known as 'ratchetting'. This is the region where RCF cracks form. The accumulation of deformation continues until the material ductility is exceeded and the material ruptures (rail break).

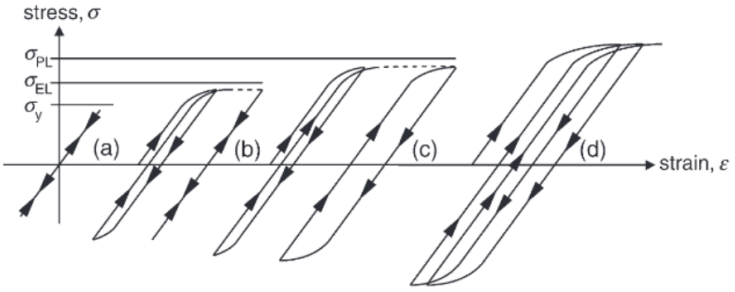


FIGURE 3.1: Material Response: Elastic-Plastic Shakedown

Shakedown theorems are fundamental to the understanding of material response, as they can describe whether plastic loading will result in continued plastic deformation (ratchetting) or whether the material will shake down to an elastic state. In particular, Ponter et al. (1985) discuss the derivation of shakedown limits for a non-conforming rolling-sliding contact, and how these can be summarised in a diagram known as a 'Shakedown Diagram/ Map'. A 'Shakedown Diagram', as illustrated in Figure 3.2 may be derived for any material, illustrating the variation in

material response under different combinations of normal and shear loads within the contact patch (Burstow, 2004). The y-axis is represented by a load factor known as the normalised vertical load p_0/k: which is the ratio of maximum normal contact pressure p_0, and the material ductility- which is represented by the shear yield strength k. The x-axis represents the frictional forces in the contact using the friction coefficient (f) or traction coefficient (μ) (which is calculated as the ratio between the traction force (T) and the normal force (N)), and the boundary curve (BC) separates the region where ratchetting occurs.

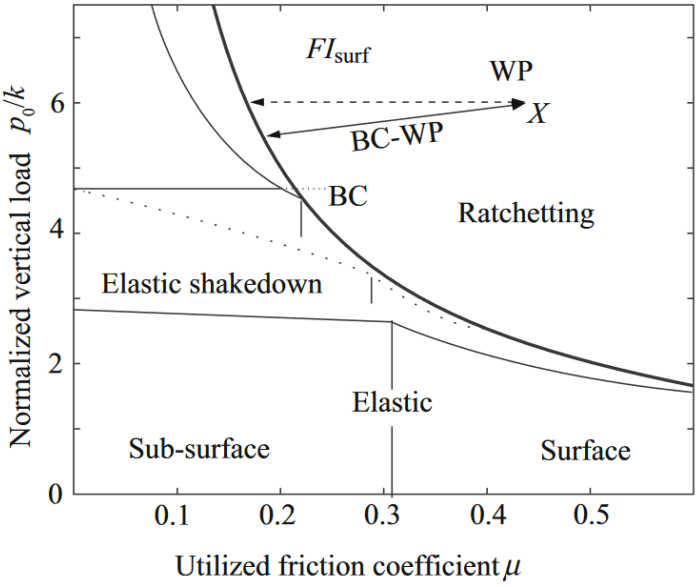


FIGURE 3.2: Shakedown Map (Krishna et al., 2021)

When applied to determine railway RCF, this approach is often used to describe whether rolling contact conditions will result in the failure of the material due to RCF or not. However, the shakedown diagram only presents the material's response in terms of the applied normal load and shear force coefficient (or traction coefficient, the ratio of tangential to normal force). In many cases, conditions may vary so that they require assessment against a number of shakedown limits. For example, the shakedown diagram does not explicitly account for the effect of spin creep or creepage so different shakedown limits would need to be used for varying contact conditions (Burstow, 2004). Further, shakedown maps alone do not provide a means for direct quantification of fatigue damage, rather whether RCF damage is occurring or not. Subsequent approaches have derived different parameters from Shakedown maps to assess the propensity for fatigue damage such as the 'shakedown exceedance' (Burstow, 2003) and 'surface initiated fatigue parameter' representing the effect of shear stress on the rail (Ekberg et al., 2002) which is illustrated in Figure 3.2. In this approach a WP (μ, v) is mapped to the diagram, and the surface fatigue index

FI_{surf} , is determined from the distance of this working point with the boundary curve (BC).

3.1.2 The Whole Life Rail Model

The WLRM is based on the energy dissipated in the wheel-rail contact, which is readily coupled with vehicle dynamics simulations and has been used widely within the rail industry (Bevan et al., 2013; Bevan, 2020; Boyacioglu et al., 2018; Rodríguez-Arana et al., 2021). Here, RCF damage is quantified from the energy generated by the tangential forces in the contact patch arising from wheel/ rail contact, which is assumed to shear the rail material.

The parameter, known as T_γ is commonly used in both wear and RCF damage predictions (Bevan, 2020). It is the energy produced at the wheel-rail contact and calculated from the sum of the products of the creepage and creep forces as defined in equation below:

$$T_\gamma = T_x\gamma_x + T_y\gamma_y + M_z w_z \tag{3.1}$$

where T_x, T_y and γ_x, γ_y are the tangential creep forces and the corresponding creepages in the longitudinal and lateral directions respectively, and M_z and w_z are the spin moment and the corresponding spin creepage respectively (Bevan et al., 2013).

Extensive analysis of RCF sites enabled significant advancements into the understanding of crack initiation, and the subsequent development of a function relating the energy generated in the contact patch ($T\gamma$) with RCF initiation risk, shown in Figure 3.3. The function comprises of four major regions:

1. $T\gamma < 15N$, below this threshold there is insufficient energy to initiate RCF,
2. Above this limit there is a risk that RCF cracks will initiate. Fatigue is initiated at 15N (J/m) and reaches its peak at 65 N.
3. The model additionally estimates the interaction of wear which may remove RCF defects, occurring at high values of $T\gamma > 65N$. The risk of wear is considerably increased in this region, and equivalent to RCF at 175N,
4. At very high values of $T\gamma > 175N$, wear is the dominant surface damage mechanism,

The 'damage index' is determined from the damage function, and is a non-dimensional number representing the proportion of the fatigue life of the material

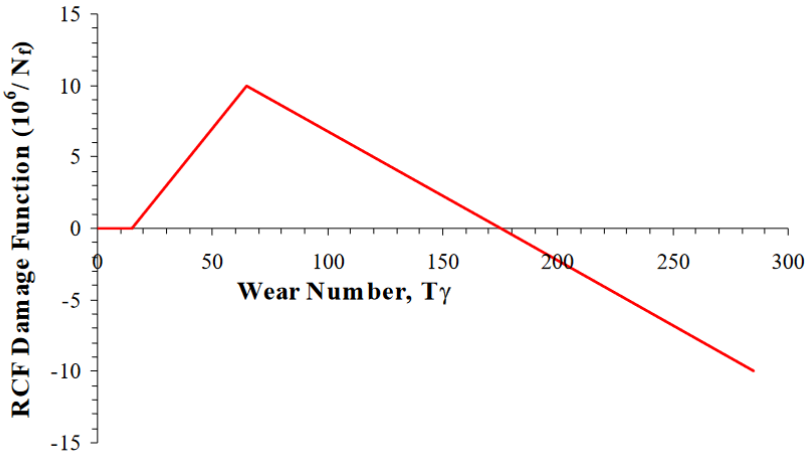


FIGURE 3.3: WLRM Damage Function (Burstow, 2004)

exhausted by the contact condition, i.e. for a single load cycle (or wheel passage). Failure occurs due to damage accumulation, and determination of the number of cycles until failure (i.e. crack initiation) may be described using the Palmgren-Miner Linear Damage Rule (Miner, 1945). The Linear Damage Rule states that failure occurs when the cumulative damage caused by each loading cycle equals one.

More practically, the principles can be extended to determine the damage over a specific site on the track, given knowledge of the accumulated traffic and contact forces generated by each vehicle type. The total accumulated damage for each site location (x, y) can be described by:

$$D(x,y) = \sum_n m_n \sum_i f(F_n v)_{i,n} \tag{3.2}$$

Where $f(F_v v)$ is the function relating damage resulting from wear and RCF to wear number shown in Figure 3.3, i is the number of axles on each vehicle, n is the number of vehicle types passing the site and m is the number of vehicles of each type passing the site in one year.

The WLRM approach has been widely used within the rail industry due to its simplicity, the ability to incorporate the effects of wear and the ease of coupling with MBS software to generate results for numerous operating conditions. There are, however some assumptions and limitations that must be noted when using the model:

- The form of the function shown in Figure 3.3 was tuned to six assessment sites from the UK network, and therefore has been calibrated for rail steel grades at these sites (R260), for specific wheel profiles (P8) and wheel steel grades (R8T) (Burstow, 2004; Bevan, 2020),

- The model assumes that the mechanisms driving RCF cracks operate in the traction direction, such that $T\gamma$ output the model is signed based on the direction of the longitudinal creep forces, and only regions where the force is opposite to the direction of travel are deduced to cause damage. Whilst studies have shown this provides good correlation on the high rail of curves, the assumption results in very low levels of predicted low rail damage (Boyacioglu et al., 2018),
- It is further noted that the model is not intended to predict the *length* of cracks, since it does not include many of the factors which govern crack growth; rather the prediction of cracks as they initiate. So, whilst it may be fairly accurate when cracks are small, as they develop the model may become less accurate. Nevertheless, the model may be used as an indicator as to areas of the rail head which are likely to see the largest cracks, since larger values of the damage index indicate significant wheel-rail contact forces (Burstow, 2004).

Finally, whilst a significantly simpler approach than finite element methods, in order to apply the method in practice, multi-body simulations for each scenario must be conducted- which is not practical on a large scale and for engineers on a day to day basis. Therefore, Network Rail have developed a software tool known as *'Track-Ex'*, which simplifies this approach through the use of look-up tables known as Vehicle Damage Matrices (VDMs), which present the wear number as a function of curve radius and cant deficiency for each vehicle type (Dembowsky, 2013).

3.1.2.1 Track-Ex

Track-Ex is the Excel-based tool developed by Network Rail which incorporates the empirically derived WLRM function to predict RCF on rails (Dembowsky, 2013). The intention of Track-Ex was to provide a tool that was relatively easy to use and fast when compared to full vehicle/ track simulation programs such as Multi Body Dynamics (MBD) software VAMPIRE. Track-Ex enables users to quantify RCF and Wear on a particular route given the expected traffic, route characteristics and track-geometry measurements. Additionally, VAMPIRE is used to generate the forces and geometry of the contact patch for multiple combinations of track and vehicle characteristics. These are culminated in a series of tables known as Vehicle Damage Matrices (VDMs), which act as look up tables for the model. Each VDM contains the wheel/rail forces with respect to curvature and cant deficiency for a particular combination of vehicle type, wheel/ rail profile, lubrication, vehicle type, and wheel/ rail friction condition, a basic illustration of these VDMs is shown in Figure 3.4.

A particular functionality of the software is to perform a 'Route Fleet Analysis' (RFA) which calculates the damage on a route for each vehicle type present in the fleet, scales the damage by the number of such vehicles in the fleet and aggregates (based on total

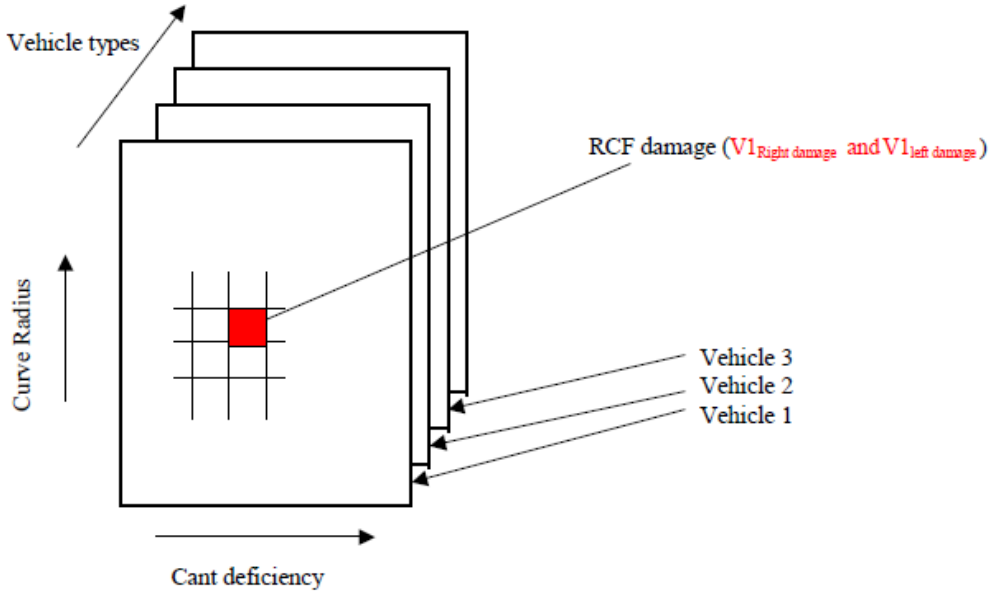


FIGURE 3.4: Illustration of Vehicle Damage Matrices (VDMs) (Burstow, 2013)

traffic) the damage into a Grand Damage Index (GDI). The Damage Index for a particular location (x, y) is calculated using the formula:

$$DI(x,y) = \sum_{min}^{max} m_n \sum_{min}^{max} h_{wh,n}(x,y) d[\mu_{wh,n}(x), T_{\gamma_{wh,n}}(x)] \tag{3.3}$$

where (x, y) are the coordinates of any point of the rail, and the damage is calculated considering m vehicles of n different types with wh wheelsets on each vehicle. The parameter h describes a function with elliptic shape over the contact patch, with its maximum damage ($d[\mu, T_\gamma]$ value) at the centroid. The approach is summarised in Figure 3.5:

1. **Determine Single Axle T_γ:** For each vehicle type, determine the Single Axle T_γ via the appropriate VDM,
2. **Determine Single Axle Damage $d[\mu_{wh,n}(x), T_{\gamma_{wh,n}}(x)]$:** Convert to a Fatigue Damage Index using the Damage Function,
3. **Calculate Scaled Damage:** Sum the damage across the number of axles i for this vehicle over the entire site which determines the approximate dissipated energy, and multiply by the annual tonnage for this vehicle,
4. **Calculate Total Damage:** Sum the scaled damage over all vehicle types in the site.

Whilst there are studies that indicate the approach indicates good correlation with measured values; particularly on the high rail, it appears that for every day rail

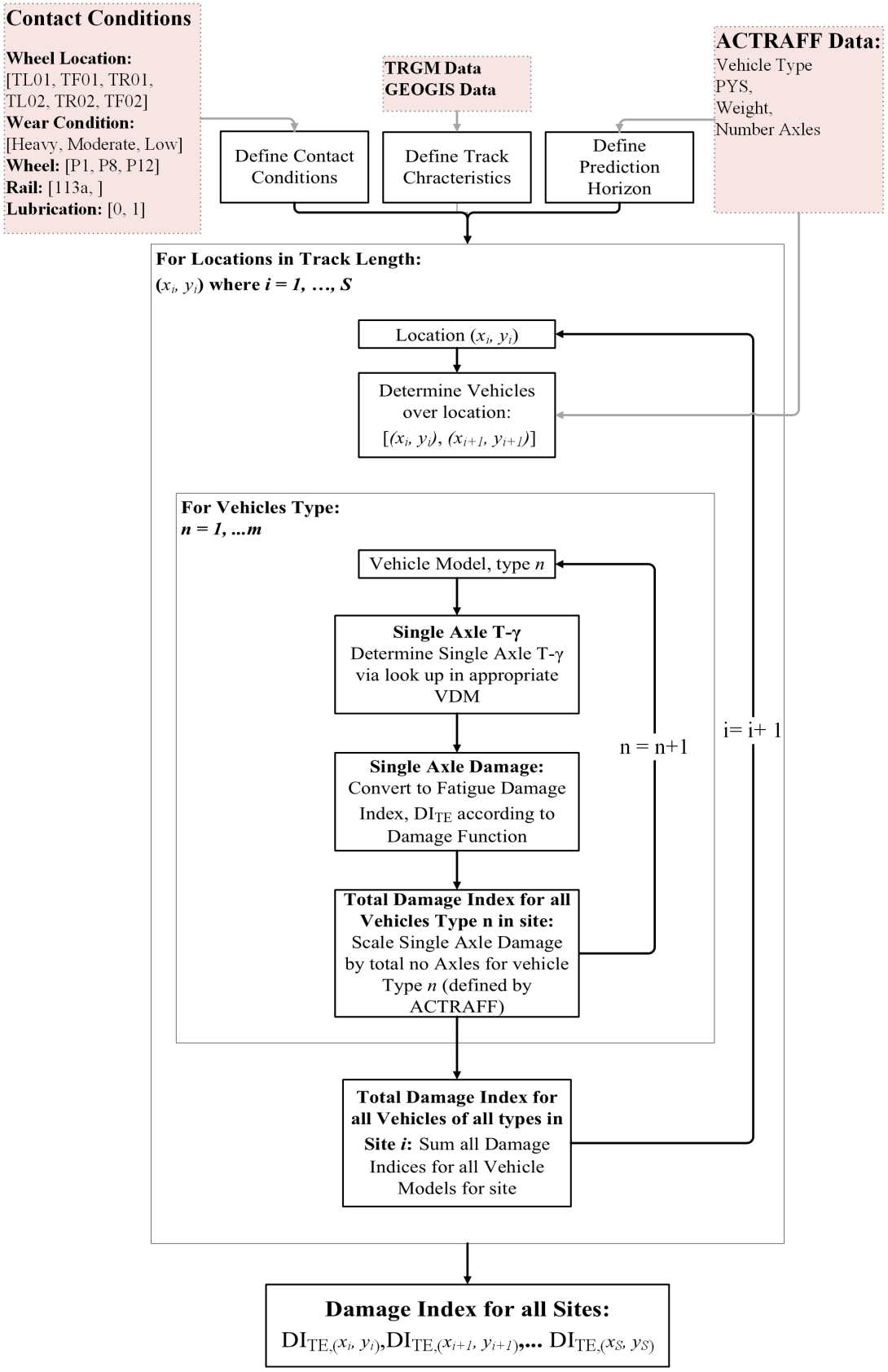


FIGURE 3.5: Track-Ex Damage Calculation Along Route

maintenance decisions and preventive/ corrective maintenance planning the track engineers on the routes do not use Track-Ex. It is believed that this may be for a number of practical and technical reasons:

- **Initial Conditions:** Assumes that the rails are new at the beginning of the analysis, and thus does not reflect the current condition of the rail. The output therefore is not useful at distinguishing how bad the rails are currently vs. for instance how they will look in a year, in order to make decisions,
- **No Indication of Rail Condition:** Following from the previous point, the model does not provide any real time information regarding the rail condition, i.e. the wear and actual EC data/ number of defects to understand the current state of the rail. Rail engineers would then have to extract this information to make any decisions,
- **Assumption of Linear Damage Accumulation:** The model assumes that damage accumulates linearly based only on traffic. Therefore, for a section of track; assuming that the traffic does not vary much from year to year, the resulting damage predictions will be similar from run to run. The model cannot inform the user, given the current rail condition, what the estimated damage will be after a given interval, nor can it include the effects of grinding or rail renewal within a given prediction period,
- **Low/ Tangent Rail Predictions:** The model predicts very low instances of damage on the low rail, and none on tangent rails,
- **Output Parameter:** Note that the output is an indicator of RCF damage initiation based on the estimated contact forces, it does not reflect the length or depth of cracks (whilst there may be correlation), it is therefore difficult to compare with real data. Thus, there is no indication of risk once a crack has initiated, and the degradation is constant regardless of any other operational parameters other than expected vehicle traffic. For example, it has been shown that once a crack has initiated, higher degradation rates are anticipated in some locations compared with others, despite similar traffic. Other parameters such as wear, ultrasonic defect data and rail age contribute to degradation and should also be considered in the maintenance decision,
- **Usability:** The software is difficult to use and apply on multiple track sections, since it requires gathering different data sources and some of the track geometry files are very large. Additionally track geometry files often have sections missing which means analysis would be to be re-run using other files to determine the damage on a particular section of track.

However, it is understood that the major value of the model is understanding of the impact of specific vehicle types and supporting vehicle design. For instance it is used within the Vehicle Track Access Calculator (VTAC), which Train Operating Companies (TOCs) and Freight Operating Companies (FOCs) can use to determine the access charges they will incur for specific vehicles. The objective here is to incentivise the operating companies to opt for more 'track-friendly' vehicles. It is also an effective model for inference, enabling designers to understand the effect of varying individual parameters such as rail type, vehicle type, track curvature, track deficiency and traffic on RCF initiation.

3.2 Data-Driven Modelling Techniques

In contrast to the physics-based approaches, data-driven modelling does not require an explicit understanding of the physical principles that drive the phenomena under study. Rather, data-driven or empirical modelling is based on the analysis of the data characterising the system under study. A model can then be defined on the basis of connections between the system state variables with only a limited number of assumptions about the physical behaviour of the system. Data-driven Modelling (DDM) covers a wide range of domains such as statistics, probability, data mining, machine learning and artificial intelligence. In this thesis, we propose an approach for developing a data-driven model, which utilises machine learning algorithms and RCF Eddy Current data provided by Network Rail to predict RCF. The fundamentals of machine learning, and model development are addressed in the following sections.

Note that for the purposes of predicting RCF evolution, as far as we know there are no known, comparable strictly data-driven models (whilst it is understood that many of the models do utilise some empirically derived relationships, the fundamentals of the models are founded in physics). The DD models which are closely related to this work however are summarised below, and the reader is referred to the references below:

- **Statistical:**
  - Rail Life/ Break Prediction Modelling: Rail failure modelling using real data to estimate distribution parameters (Dick et al., 2003; Palese, 2000; Ben-Gera et al., 2016; Chattopadhyay, 2009; Kumar, 2006),
  - Correlating track parameters to visual RCF damage: Study uses statistical correlation analyses to relate parameters such as curve radius, rail profile and traffic type to damage using visual RCF inspection data (Magel, 2011),
- **Probabilistic:**

  - Determining the probability of crack initiation: using Monte Carlo simulation, (Kulkarni et al., 2006; Jianxi et al., 2011),
  - Maintenance Scheduling: Determine optimal preventive maintenance strategy using Markov chains, with parameters estimated from actual inspection and repair data from a railway line in Norway, (Hokstad et al., 2005),
  - Modelling Multi stage deterioration using Markov chains (Amari et al., 2006)

- **Machine/ Deep Learning:**
  - Economic Impact of prediction of broken rail using logistic regression (Schafer, 2008),
  - Rail Failure Prediction using logistic regression and decision trees (Sourget, 2006),
  - Modelling of Squats:
    * **Squat risk model:** A defect based risk model is presented in (Jamshidi et al., 2016),
    * **Squat defect classification:** Classification is achieved by means of a Convolutional Neural Network (CNN) trained on the rail video data, in combination with crack growth data collected from ultrasonic measurements (Jamshidi et al., 2017),
    * **Squat decision support tool:** (Jamshidi et al., 2018) propose a decision support approach which takes into account both the actual conditions of the rails (using axle box acceleration measurements and rail video images) and prior knowledge of the track. The approach provides an integrated estimation of the rail health conditions using expert based systems in order to support the maintenance decisions for a given time period. This estimation of rail health conditions facilitates grinding planning of those segments that are prone to critical conditions using Mixed Integer Linear Programming (MILP),
    * **Squat defect classification:** Gao et al. (2018) use shallow machine learning techniques to combine data from three inspection systems, ultrasonics, eddy current based, and video data to provide optimal detection and classification of squat type defects,
  - Fumeo et al. (2015) propose a novel algorithm for the prediction of Remaining Useful Life (RUL) of train axle bearings,
  - Defect Prediction from UT data: Random Forests and Artificial Neural Networks are used for defect prediction Guler (2014); Moridpour (2016); Sharma et al. (2018),
  - Predictive Maintenance Scheduling Policies (Lopes Gerum et al., 2019).

3.2.1 Machine Learning

Machine Learning is a multidisciplinary field branching from artificial intelligence, which combines elements of statistics, probability, computer and cognitive science. It is recognised as an algorithmic approach that constructs systems that can learn from data to make inferences and predictions around a particular phenomena. Or, as Arthur Samuel once described it as the *"field of study that gives computers the ability to learn without being explicitly programmed"*.

Fundamentally, machine learning revolves around the presentation of an optimisation problem (Brunton, 2019). Depending on the specific task, broadly we seek to determine the optimal function describing the input data subject to specified objectives and constraints by use of such algorithms. The three critical components of all machine learning algorithms are effectively summarised by Domingos (2012):

- **Representation:** How do we represent the problem such that it is interpretable by a computer? For example: instances, hyperplanes, decision trees, sets of rules, neural networks and graphical models.
- **Evaluation:** Objective/ Scoring/ Cost Function, how does one determine between a good regressor/ classifier and a bad one? This evaluation method provides a performance metric for model predictions.
- **Optimisation:** Search method, how do we find the highest scoring regression function/ classifier? The choice of optimisation technique is key to the efficiency of the learner (how quickly one converges to a solution), and helps determine the outcome in the case that multiple optima exist. Examples of each of these components are illustrated in Figure 3.6.

Table 1: The three components of learning algorithms.

Representation	Evaluation	Optimization
Instances	Accuracy/Error rate	Combinatorial optimization
K-nearest neighbor	Precision and recall	Greedy search
Support vector machines	Squared error	Beam search
Hyperplanes	Likelihood	Branch-and-bound
Naive Bayes	Posterior probability	Continuous optimization
Logistic regression	Information gain	Unconstrained
Decision trees	K-L divergence	Gradient descent
Sets of rules	Cost/Utility	Conjugate gradient
Propositional rules	Margin	Quasi-Newton methods
Logic programs		Constrained
Neural networks		Linear programming
Graphical models		Quadratic programming
Bayesian networks		
Conditional random fields		

FIGURE 3.6: The Three Components of Machine Learning (Domingos, 2012)

3.2.2 Categories of Machine Learning

There are two main classifications of machine learning algorithms: *Supervised Learning Algorithms* and *Unsupervised Learning Algorithms*.

Supervised Machine Learning (SML) relies on the use of domain experts to assign labels to the input data in order to 'teach' the learning scheme of the relationship between the input data and an output variable or class. The ML algorithm generates a mapping between the inputs and outputs which is iteratively updated in accordance with the objectives of the algorithm. This annotated data set is known as the 'training set', and the ultimate objective is to develop a function that is capable of generalising on a much larger 'unseen' set of data (Shetty et al., 2022). More formally, we aim to find a mapping function from a set of input variables (X) and an output variable (Y): $Y = f(X)$. The function is learned through use of an algorithm and a dataset which comprises of labelled examples $(x_i, y_i)_{i=1}^{N}$. Where the output Y is discrete the aim is to classify the data into groups: *Classification*, whereas predicting a continuous output is determined using *Regression*.

Unsupervised problems are *unlabelled*, such that the output is unknown. The learning scheme thus aims to determine patterns or classifications within the data (Lechevalier et al., 2014). The predominant application of unsupervised learning is a technique known as clustering, which identify inherent groupings within the unlabelled data and subsequently assign labels to each data value (Marshland, 2014). Other algorithms include *Association Rules Algorithms* which tend to identify rules that accurately represent relationships between attributes (Alloghani et al., 2020), and *Dimensionality Reduction Algorithms* such as Principal Component Analysis which aims to reduce the number of features in a dataset (Campesato, 2020).

Figure 3.7 shows the most commonly used machine learning algorithms today for supervised and unsupervised tasks.

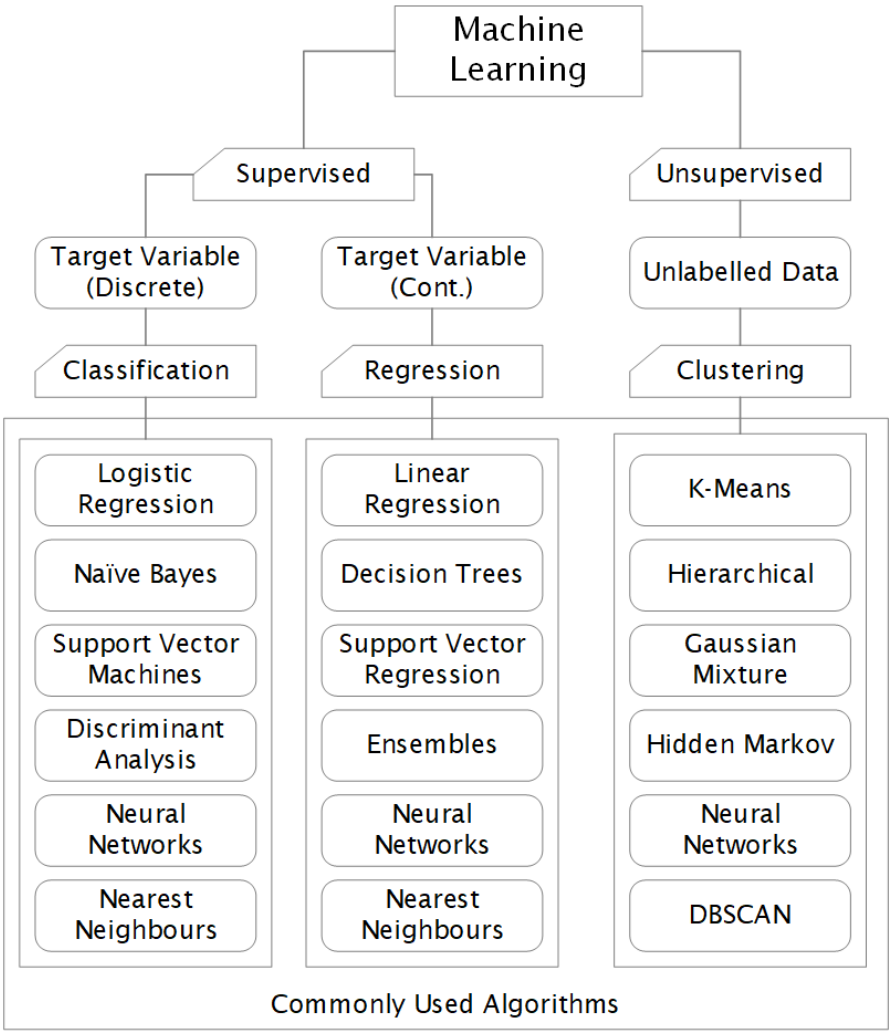


FIGURE 3.7: Broad Classification of Machine Learning Algorithms

3.2.3 Model Validation

A central requirement when building these computational models is to develop an approach which has high prediction capabilities for the input samples, but should also generalise well to previously unseen data. Poor generalisation can be characterised by over-training. If the model over-trains, it just memorizes the training examples and it will not be able to give correct outputs also for patterns that were not in the training dataset. These two crucial demands are conflicting and are also known as the bias-variance dilemma (Mitchell, 1999). This trade-off is illustrated in Figure 3.8 which demonstrates the relationship with model complexity; or the capacity of a model to represent associations between model inputs and outputs. The complexity of different models can be compared by their number of parameters and the way these

parameters interact in the model (eg, linear, non-linear). Models with high complexity often tend to be too sensitive to the dataset used for training, and where different data sets drawn from the same population are used for training, have a high error variance. Models with high complexity and consequently high error variance tend to over-fit, and therefore their generalisation capability is limited. In contrast, low-complexity models may be biased to learning simpler associations between inputs and outputs that might not be sufficient for representing true associations. Developing an optimal model requires a trade-off between bias and variance by controlling model complexity. Techniques such as bagging and boosting can also be used to control the bias and variance of a model (Maleki et al., 2020).

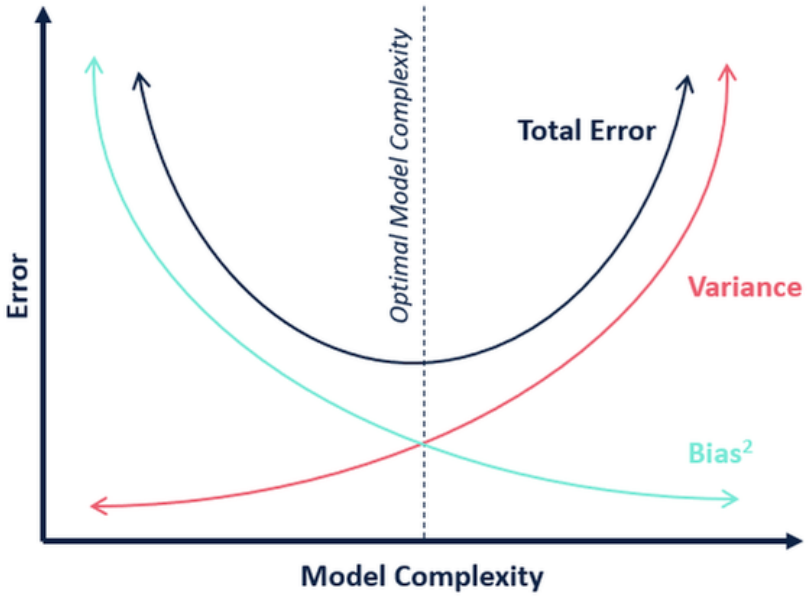


FIGURE 3.8: The Bias- Variance Trade-off

In order to manage the problem of over-fitting to the training data, the data are often partitioned into training, validation and test sets, and a performance measure (such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Square Error (RMSE) or the R^2) is used to reflect the model error when applied to data in these sets. The error made by a model when applied to the data in the training set is referred to as the *training error*, and the error made by a model when applied to data in a test set is referred to as *test error*. The *test error* is used as an estimate for the *generalisation error* (i.e., the error of the model when applied to unseen data).

Additionally the machine learning algorithm, as described above typically transforms a problem into an optimisation problem and uses different methods to solve this problem. The optimisation function is composed of multiple hyper-parameters that are set prior to the learning process and affect how the machine learning algorithm fits the model to data. Notably these differ from the internal model parameters, such as

the neural network's weights, which can be learned from the data during model training (Claesen, 2015). In order to determine a set of hyper-parameter values which achieve the best performance on the data, a process known as hyper-parameter optimisation or tuning is adopted (Wu et al., 2019). Predominantly there are two kinds of hyper-parameter optimisation methods, i.e., manual search and automatic search methods. Manual search requires users to have more professional background knowledge and practical experience, however as the search space grows this becomes more and more difficult to manage, hence the development of automatic search methods such as grid search. Grid search is an exhaustive search technique where the machine learning model is trained with each combination of possible values of hyper-parameters on the training set and evaluates the performance according to a predefined metric on a validation set. Finally, grid search outputs hyper-parameters that achieve the best performance.

Thus the validation data set seeks to tune the model hyper-parameters, whereas model testing is used to estimate the performance of the final model on unseen data. It is generally good practice to ensure that the data in the test set are not used during training or model fine-tuning since this will lead to biased estimates, however in the case of small data sets this is sometimes not possible.

There are many methods for splitting the data into appropriate training, validation and testing subsets, but the most commonly used techniques are:

- **Random:** Selecting a proportion of samples for training and retaining the remaining ones (holding out) as a validation set. Usually, we repeat this process many times and the final estimation of the model performance is the average performance on validation sets of all the iterations; the best-known method used for this type of repartitioning of the data is probably the bootstrap as proposed by Efron (1994).
- **Cross-Validation (CV):** is a data resampling method to assess the generalisation ability of predictive models and to prevent over-fitting, and has two types which are *hold-out cross-validation* and *k-fold cross-validation* (Schaffer, 1993).

Irreducible error, also referred to as Bayes error, is another type of error resulting from the inherent noise in the data. Irreducible error is the lowest possible error achievable for a given task using the available data. This error is independent of the model being used and often cannot be mathematically calculated. It is often estimated by the error made by a group of humans with the domain expertise for the task at hand. The resulting estimate is considered as an upper bound for irreducible error. Understanding these error types is important for developing and evaluating ML models.

3.2.4 Application of ML Algorithms for RCF Prediction

In this study, a central objective is to determine whether a useful RCF prediction model can be generated using ML algorithms and operational Eddy Current data which represents rail damage due to RCF. In particular, we seek to find a function which relates a set of independent variables X; such as track curvature or cumulative tonnage, with a target variable Y (or RCF damage parameter) derived from the Eddy Current data representing observations collected along the track over time. The task we present is a *Supervised Learning Problem*, since we have a set of observations with features (independent variables) and corresponding labels (a target variable). The target variable in its original form is a continuous measurement of crack depth (mm) corresponding to a measurement date and 1 yard section of rail (this is further described in Chapter 4). It is possible, using discretisation, to transform the continuous target variable and generate a set of classes which presents a *classification* task. Whilst the transformation may be helpful with regard to interpretability and simplification of the learning task, in most cases however it introduces some loss of information or discretisation noise (Rajbahadur et al., 2021). We therefore focus this study on the *Supervised Regression Problem*.

Whilst numerous machine learning algorithms exist, this study focusses on a few of those most prominent within the literature for solving the regression problem (Caruana, 2006; Alzubi et al., 2018; Dey, 2016), These algorithms are introduced here and a more in-depth discussion is presented in Appendix B:

1. **Linear Regression (LR):** Linear regression is one of the simplest and most commonly used machine learning algorithms. It is a mathematical approach which investigates the relationship between a dependent (target) and independent variable(s) (predictor). The approach is to fit a line (linear regression) or curve(polynomial regression) to the data points so as to minimise the distances of data points from the curve or line. The most common method of solving this problem is to use the method of 'Least Squares', whereby we fit coefficients which minimise the least squares difference between the regression function and the observations.

2. **Random Forest (RF):**

   The random forest (RF) is a methodology derived from decision trees. A decision tree is a non-parametric supervised learning algorithm which has a hierarchical, flowchart-like tree structure, consisting of a root node, branches, internal nodes and leaf nodes and can be used to solve regression or classification problems (Breiman, 1984). Each of the internal nodes represents a test on an attribute, for example in the case where we attempt to determine animal type based on attributes such as the number of legs, one of the internal

nodes may represent 'Does the animal have more than 2 legs?'. Each branch represents the outcome of the test, and each leaf node cannot be divided further and consequently holds a class label e.g. 'Rabbit', 'Dog' etc. Different criteria can be used to select the predictors and decision rules which define the nodes and leaves, resulting in numerous different tree constructor algorithms which aim to build an optimal tree. Trees can capture highly complex interaction structures in the data and; if grown sufficiently deep, have relatively low bias. On the other hand, they are noisy, high variance estimators which benefit from averaging to prevent over-fitting to the training data. Methods such as random forests use the output of multiple independent decision trees to reduce the overall generalisation error (Hastie et al., 2009).

3. **Support Vector Regression (SVR):** Support Vector Machines (SVMs) can be used for classification as well as regression problems, and the method has increased in popularity due to its good generalisation performance, the ability to determine a global minima, and sparse representation of solution. The principles of the support vector machine are based on the structural risk minimisation(SRM). This induction method seeks to minimise an upper bound of the generalisation error, which consists of the sum of the training errors and a confidence interval, rather than the empirical risk minimisation (ERM) principle which minimises the training error (Vapnik, 2006; Cao, 2003). In practice the algorithm builds an n-dimensional hyperplane which partitions the data subject to an optimisation function and constraints. In addition where the data are non-linearly separable, the data may be transformed to a higher dimensional feature space using a kernel transformation enabling greater flexibility to the algorithm.

4. **K-Nearest Neighbours (KNN):** is a non-parametric method, also known as an instance-based or lazy learning technique which can be used for classification or regression, and is particularly useful where there is little or no prior knowledge about the distribution of the data (Hart, 1968). The KNN rule simply retains the entire training set during learning and assigns to each observation a class or value represented by the majority label or average value of its k-nearest neighbours in the training set. The Nearest Neighbor rule (NN) is the simplest form of KNN when K = 1.

5. **Multi Layer Perceptron (MLP):** The Multi Layer Perceptron (MLP) is a type of neural network designed to approximate any continuous function and solve problems which are not linearly separable. It is an extension of its predecessor, the feed forward network and is generally applied to problems of pattern classification, recognition, prediction and approximation. It consists of three types of layers: the input layer of source nodes, an output layer of neurons (or computation nodes) and one or more hidden layers. The input signal is

processed by the input layer, and the required prediction or classification task is performed by the output layer. In between exist an arbitrary number of hidden layers, which enable the modelling of complex function forms. The training of an MLP is accomplished using the back-propagation algorithm (Rumelhart, 1987). As with a feed-forward network, the data first flows in the forward direction from input to output layer, fixing the network parameters and determining an error signal. This is followed by the backward phase, where the error signal is propagated through the network and adjustments are made to the network to minimise the error (Haykin, 2000).

In general, once the learning problem has been defined; regardless of learning algorithm, the basic architecture of a supervised learning approach (or pipeline) consists of:

1. **Data Collection and Preparation:** The primary task of in the machine learning process is to collect the data believed to be pivotal in predicting the underlying process,
2. **Data Processing:** Preparation of the data to ensure it is in a suitable format for the given learning algorithm, data cleaning, transformation and integration across differing sources,
3. **Feature Selection/ Generation:** Numerous features may be available following the extraction of data, the most relevant must be selected and new features which may act as better predictors may be formed as functions of the raw data,
4. **Choice of Algorithm:** Certain machine learning algorithms are more suited to a particular class problem, selecting the best machine learning algorithm for the problem at hand is imperative in getting the best possible results. The various ML algorithms considered in this study are discussed in Appendix B.
5. **Selection of Model hyper-parameters:** Each machine learning algorithm comprises of model *hyper-parameters* which control the learning process and affect how the machine learning algorithm fits the model to data. They are considered as external parameters, and differ from the internal parameters such as model weights which are learned by the model during training (Claesen, 2015),
6. **Define Training/Test Data:** The data must be partitioned into training and testing sets to enable model validation,
7. **Model Training:** a supervised learning algorithm iteratively calculates a relationship between the input features and output through optimising of a *cost function*,

8. **Model Evaluation:** The trained model makes predictions on the test data and are evaluated using *performance metrics* such as the Mean Absolute Error, Residual Sum of Squares (RSS), and Coefficient of Determination (r^2) amongst many others.

Although shown sequentially here, in practice the approach is iterative, for example preliminary model building may inform additional data processing steps that are required to learn effectively from the data. Development of the model presented in this thesis follows this development process, the first two steps are addressed in Chapters 4 and 5, and the final steps are covered in Chapter 7.

3.3 Concluding Remarks

There are many physics based models in the literature, we have described only the approaches that could be practically applied to making predictions on a network level for maintenance planning. Track-Ex is an empirically based software tool developed by NR which uses MBS simulations to determine contact forces along the rail, and the WLRM damage function to estimate the resulting damage. It is a powerful approach, but it is apparent that it is not used widely by route engineers for maintenance planning due to both technical and practical reasons. Data-driven methodologies, in particular machine learning algorithms offer an alternative solution to RCF modelling where rail condition data is available. In the literature, however, particularly in the case of utilising Eddy Current monitoring data, there are very few studies in this area with work predominantly focussing on track geometry data, video data and prediction of squat type defects. This thesis thus seeks to develop a purely data-driven methodology using machine learning algorithms which combines the key drivers for RCF initiation that may be incorporated into practical strategies for rail maintenance. In order to reach this goal, the phenomena under study must be thoroughly understood, as must the Eddy Current data and supporting data sources including their characteristics and limitations. A data pre-processing methodology must be devised to prepare the data for modelling and further analysis. Finally, the resulting data should be analysed and a regression model developed using the proposed machine learning algorithms and validated against a suitable test set. These elements are all described in subsequent Chapters 4-7.

Part II

Modelling Work

Chapter 4

Experimental Data

4.1 Data Sources

Central to this study is infrastructure data, captured in a multitude of systems and databases ranging in age, quality, and interoperability. For historical reasons, advances in technology and the intended usage of the information, many of these databases are not readily comparable, and some data streams are not contained in relational databases at all. Since data engineering and management is a discipline in itself, it is not always possible to efficiently manage growing sources of data.

Study of the literature and existing physics-based RCF modelling has culminated in a selected range of data sources to be utilised in this study, these are as follows:

- **Eddy Current Data:** the Eddy Current data provides the response data at the core of this study, as described in Section 2 EC technology captures RCF cracks in their very early stages of growth. The Sperry Roller Search Unit, is onboard the UTU, however is an entirely independently operating system to the ultrasonics and other systems on the UTU. The data is collected and processed in real-time, retaining only the yards of track locations where cracks are detected in order to reduce the size of the data set,
- **Track Data:** defines spatial characteristics of the study route, the data is comprised of homogenous track sections containing information such as track curvature, cant, line speed, track category, amongst others. The data has been collected from a table known as the Track Summary table (used as input to the Track Model), which is constructed from information contained in several databases:
  - **Curvature:** is measured in units (1/m) since it is calculated from the curve radius. Curvature is known to be a key driver of RCF, due to the steering

forces required to guide the vehicles around the track, in particular on the high rail of a curve. Note: There are some locations where the track curvature is noted to be absolute 0, it is likely that this indicates missing data as opposed to zero curvature,

- **Track Cant (Super-elevation):** is difference in elevation between the two rails. Cant is designed into the track system, it helps the train to steer around curves, keeping the wheel flanges from contacting the rail surface which reduces friction and wear, canting also allows part of the lateral acceleration to be provided by gravity, thus reducing the lateral forces on the contact patch. It has been observed that by ensuring the rail is in a cant deficiency, where the resultant force exerts on the outside rail more than the inside rail, the RCF damage is reduced on most curves. Cant is measured in (mm), as the amount of super-elevation required to bring the resultant force into balance.
- **Equivalent Mega Gross Tonnes per Annum (EMGTPA):** is used to represent the forces applied to rail by traffic, and represents the scaled volume of traffic,
- **Maximum Line Speed:** higher vehicle speeds impose larger contact forces on the WRI,
- **Locations of Features** the positions of discontinuous features such as S&C, Stations, Tunnels are collected to account for potential variation in the occurrence of RCF due to factors such as traction and braking, and different maintenance practices in these areas,
- **Rail Profile Type:** refers to the rail profile, types include BS113A Flatbottom, BS110 Flatbottom, UIC 60 Flatbottom,
- **Rail Alloy:** the specific rail materials, most commonly across the network the rail is a medium manganese alloy of British origin and also a wear resistant medium manganese variant, however there are some instances of High Performance Rail being installed in recent years along with older compositions.
- **Track Category:** each line of route is assessed on three key factors: the speed required on the line, the annual tonnage, the equivalent tonnage (as described above, it is a measure of the annual tonnage carried over a section of track but accounts for variations in track damage caused by different types of rolling stock). The track is assigned a category from 1A to 6 based on a function related to EMGTPA, Category 1A is the highest (>125mph), and Category 6 is the lowest (<20mph). Figure 4.1 shows the track category matrix.

- **Wear Data:** wear and RCF are inextricably linked, in particular due to some of the positive effects wear can have on the initiation and propagation of RCF

cracks. NR utilise the KLD labs supplied laser rail profile measuring unit, which is mounted upon the UTU. Profile measurements such as vertical wear (head wear), gauge wear, and field wear are captured by the device and data is uploaded to a NR network drive following each recording run,

- **Traffic Data:** is extracted from the Actual TRAFFic (ACTRAFF) database to represent the cumulative forces applied to the rails due to passenger and freight vehicles. The data supplied indicates the passenger and freight tonnage and number of trains travelling between two points on the network for a unique 4 week period. DeltaRail were responsible for collecting this data, and utilised records of traffic movements through signals and freight charging. Reportedly the data only covers 85% of the network, but it is more accurate than other traffic datasets such as Net Traffic (NETRAFF) on the track that it does cover. The ACTRAFF data additionally provides information on the breakdown of vehicle types for each period. Note that the contract for collecting ACTRAFF data was halted in 2019 and is awaiting a replacement methodology, thus there is no traffic data available from 2019. Data for later periods must be estimated.

- **Track Geometry Data:** is captured by the Track Recording Vehicles (TRVs), using an unattended track geometry measuring system with 47 sensors to record data on how far the track is deviating from its ideal geometry,

- **Defect Data:** the presence of other defects and flaws in the rail are believed to affect the likelihood of RCF initiation and risk of propagation. Ultrasonic sensors fitted to the Ultrasonic Test Unit to find flaws in the rail is NR's predominant method for detecting severe defects,

- **Interventions Data:** maintenance actions applied to the rail will have a significant impact on the level of rail damage detected by the EC RSU. For example, rail replacements should entirely remove any traces of RCF, whereas grinding and milling are intended to remove surface cracking and reduce propagation, and so may be observable in subsequent measurements as a reduction in the degradation rate or as an improvement (negative degradation- i.e. the condition of the rail is better than in the previous measurement). The sources of interventions data are from the following:

  - **ELLIPSE DATA:** ELLIPSE is a system for managing and recording asset maintenance activities. It is used by the maintenance function for the key purposes of acting as an asset register, a work bank managements tool, a means of scheduling work and allocating resources, a record of work carried out, and to record asset condition and associated condition monitoring data. The system tracks the majority of in-house maintenance: activities such as tamping and stone-blowing are captured, most machine grinding and milling are captured, however it is likely that there is some

missing data since this work is completed externally. The grinding and milling reports issued by the contractor are presented as individual PDF files,

– **Route Data:** grinding databases captured by the routes independently are also obtained. Entries have a start yardage, end yardage and a measurement date. Note that we only have data from Anglia in this study,

– **Integrated Network Model (INM):** the rail replacement year is taken from the INM. The INM is the geospatial view of the railway network, and is the complete record of the track and its components, and is the most reliable source for rail replacement data. Once rails are replaced the routes must trigger a procedure to update the network model (INM).

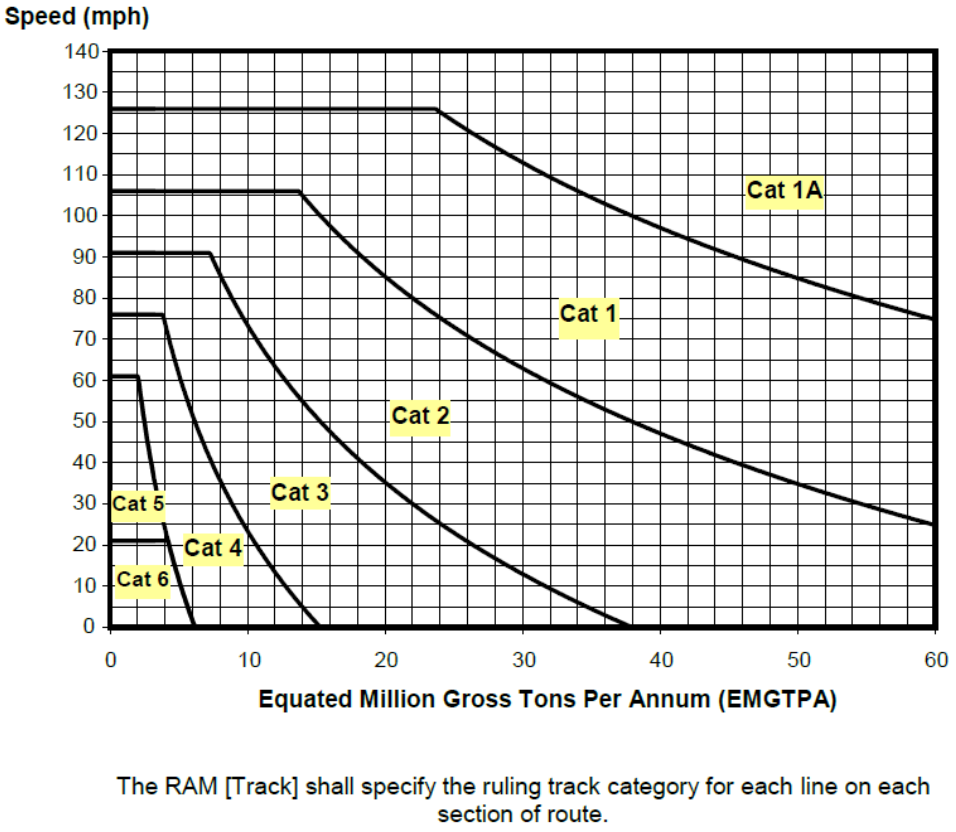


FIGURE 4.1: The Track Category Matrix (Network Rail, 2018a)

4.2 Study Routes

The UK rail network is divided into five regions, Eastern, North West and Central, Scotland, Southern and Wales & Western. These fives regions encompass the 14 major routes, each of which are responsible for operations, maintenance and minor renewals. This study focusses on track lengths from the Anglia and Wessex Routes from Eastern

and Southern regions respectively, chosen due to their priority, amounts of RCF and the size of data set. Note that any main-line railway location on Network Rail owned, or maintained, infrastructure can be uniquely identified by a combination of Engineers Line Reference (ELR), Track ID (Track ID (TID)) and mileage. Two lengths of track were selected for this study, the first is *Track Length 1*: Anglia LTN1 2100 and *Track Length 2*: Wessex BML1 2100, the details of these track lengths are as follows:

- **Track Length 1 (TL1):** *Region: Eastern, Route: Anglia, ELR: LTN1, TID: 2100,*
  One of the motivations for selecting a track length from the Anglian route is due to the maintenance team contributions with regard to understanding of current rail maintenance practices. Additionally, the track length is part of the Great Eastern Mainline (GEML), a 114.5-mile (184.3 km) major railway line on the British railway system which connects Liverpool Street station in central London with destinations in east London and the East of England. It is a major route for freight services from Felixstowe, Tilbury and other locations, and therefore will experience unique operating conditions with regard to loading and vehicle types.

  The line of interest is defined by ELR LTN, the London Liverpool to Norwich Line. The line consists of LTN1 and LTN2, but only the former is included in this study. LTN1 runs from London Liverpool Street to Trowse Junction (just outside of Norwich), and we study the down-fast (2100) line only (i.e. moving from London to Trowse). LTN1 2100 is 113 miles long, with areas of track which have high 'Track Category' which makes it an ideal candidate for study due to the quantity of RCF condition data that is available since tracks with a higher category are measured more frequently.

  The line is defined in Table 4.1 and illustrated in Figure 4.2, note that freight traffic is diverted to Felixstowe at Ipswich.

- **Track Length 2 (TL2):** *Region: Southern, Route: Wessex, ELR: BML1, TID: 2100*
  The second line under study is part of the South Western Main Line (SWML), a 143 mile major railway line between Waterloo station and Weymouth on the south coast of England. The line predominantly acts as a passenger line, serving many commuter areas such as south western suburbs of London and the conurbations based on Southampton and Bournemouth.

  Operating speeds on much of the line are relatively high, with large stretches cleared for up to 100 mph (160 km/h) running. The London end of the line has as many as eight tracks plus the two Windsor Lines built separately, but this narrows to four by Wimbledon and continues this way until Worthing Junction west of Basingstoke, from which point most of the line is double track. A couple of miles from the Waterloo terminus, the line runs briefly alongside the Brighton Main Line west branch out of London Victoria, including through Clapham

Junction – the busiest station in Europe by railway traffic. Tourist special services to a lesser frequency use the line, such as the Cathedrals Express.

The second line in this study is the Bournemouth Main Line (BML), consisting of BML1, BML2 and BML3. This study covers BML1 between London Waterloo and Northam Short Mile (Mount Pleasant). Once again BML1 2100 was selected since it is a long, continuous fast line spanning 75 miles, it thus has a reasonable amount of Eddy Current data for analysis. It has also been mentioned following conversations with track experts as experiencing high levels of RCF, likely due to the shear quantity of traffic passing through some of the major UK stations such as Clapham Junction.

Figure 4.3, includes this line (termination just short of Southampton Central, which then merges with BML2), and is described in Table 4.1.

FIGURE 4.2: The Great Eastern Mainline

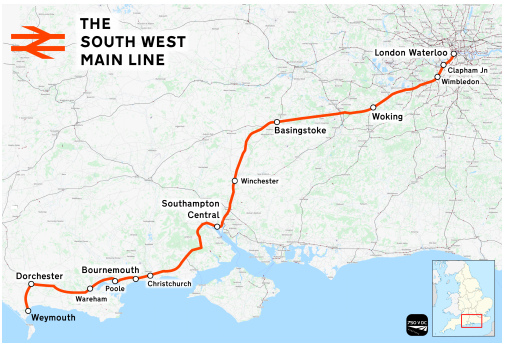


FIGURE 4.3: The South Western Mainline

Additionally, according to the track summary data, TL1 consists of 86 curves and 52.6% of the segments are tangent, whereas for TL2 there are 77 curves with 65.4% of the track segments being tangents. Figure 4.4 shows a comparison of the distribution of curve radii for both track lengths, TL1 has considerably more curved sections with curve radii between 1000 and 2000 m.

TABLE 4.1: Study Route Line Definitions, reference data :

Route	Track Length 1	Track Length 2
Region	Eastern	Southern
Route	Anglia	Wessex
ELR	London Liverpool to Norwich Line 1 (LTN1)	Bournemouth Main Line 1 (BML1)
TID	2100: Down-Fast	2100: Down-Fast
Start	London Liverpool Street: 00 mi	Waterloo: 00.00 mi
End	Trowse Junction: 113.68 mi	Northam Short Mile: 77.68 mi
Major Stations	Bethnal Green: 1.10 mi, Stratford: 4.03 mi, Ilford: 7.28 mi, Romford: 12.30 mi, Chelmsford: 29.6 mi Colchester: 51.52 mi, Ipswich: 68.59 mi (Major Freight Junction - Felixstowe), Diss: 94.79	Clapham Junction: 3.74 mi, Wimbledon: 7.19 mi, Woking: 24.27 mi, Basingstoke: 47.61 mi, Winchester 66.39 mi, Southampton Airport Parkway: 74.66 mi

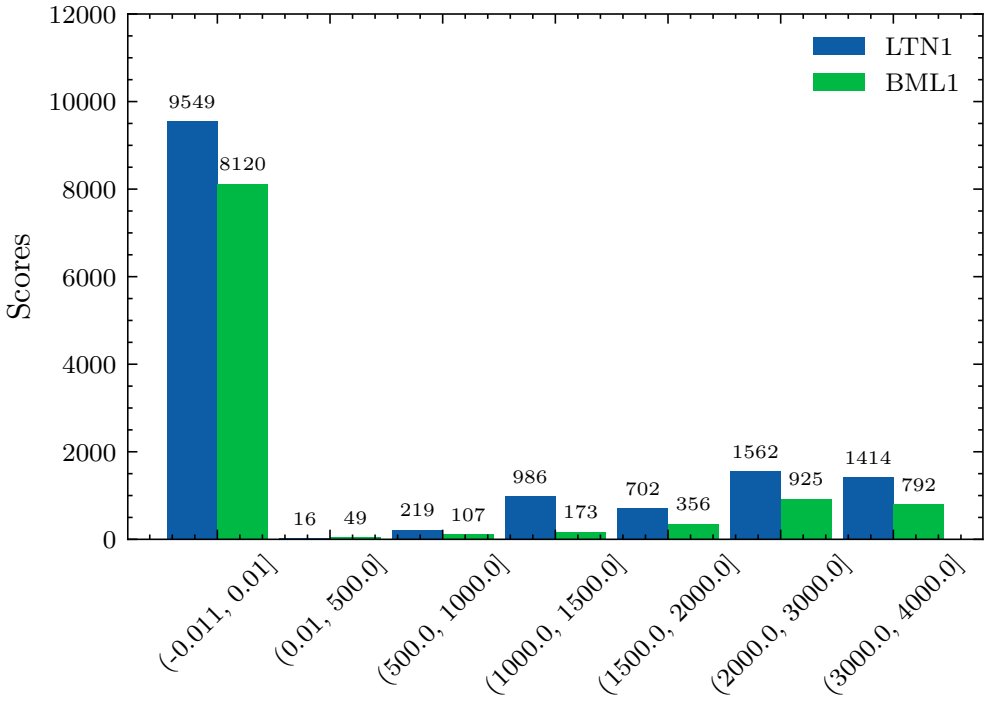


FIGURE 4.4: Comparison of Track Curve Radii Distribution for TL1 and TL2

4.3 Eddy Current Data

The Sperry Eddy Current system automatically detects rail that contains RCF and will initially divide the RCF into 1 yard sections. Rail without RCF is not reported. Each 1 yard section containing RCF is analysed automatically and the following information is reported (Network Rail, 2018c):

- ELR, Track ID, GPS coordinates, single mileage (for individual yards) or start and finish mileages (for grouped sites),
- The maximum crack depth of the deepest crack,
- Category of RCF in accordance with Table 4.2,
- The rail affected (left or right rail),
- The maximum depth measured by each of the 10 eddy current probes. The probes are numbered 0-9 from the gauge across to the field side of the rail, as pictured in Figure 4.5,
- Measurement Date.

The data are initially processed and filtered on board the Roller Search Unit, these are then uploaded by Sperry into the RDMS following an inspection. The planned and maximum intervals between EC inspections are set according to Track Category (shown in Table 4.3 which corresponds to Table 11 from (Network Rail, 2018c)). In the case of existing RCF sites, a follow up assessment is prescribed in accordance with Table 10 in (Network Rail, 2018c) (not shown here), which indicates for example that some Severe and Very Severe sites must be reassessed as frequently as 6 weekly (42 days).

TABLE 4.2: Network Rail RCF Severity Categories, Table 9 from (Network Rail, 2018c)

Crack Depth (mm)	**Severity Category**
$0.1 \leq x \leq 1.5$	Light
$1.6 \leq x \leq 3$	Moderate
$3.1 \leq x \leq 4$	Heavy
$4.1 \leq x \leq 4.9$	Severe
Indicated 5mm deep (>5mm but beyond measurement capability)	Very Severe

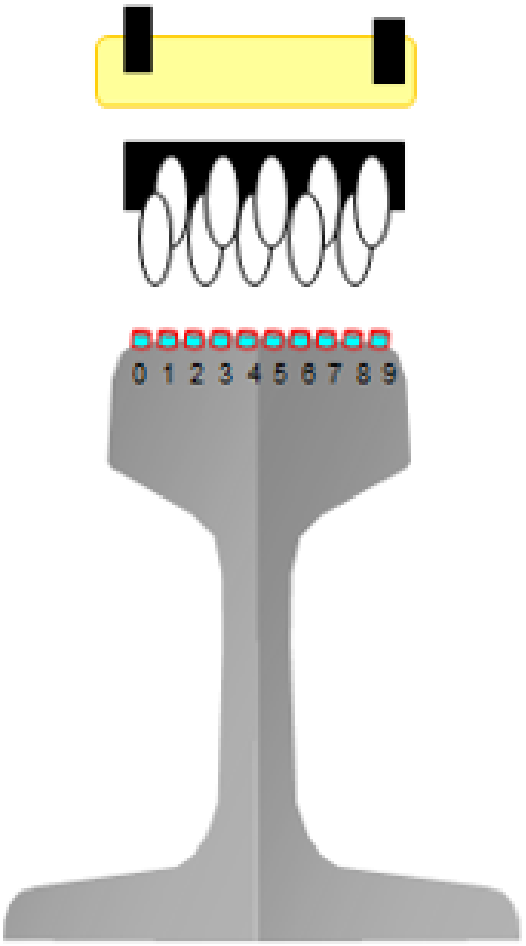


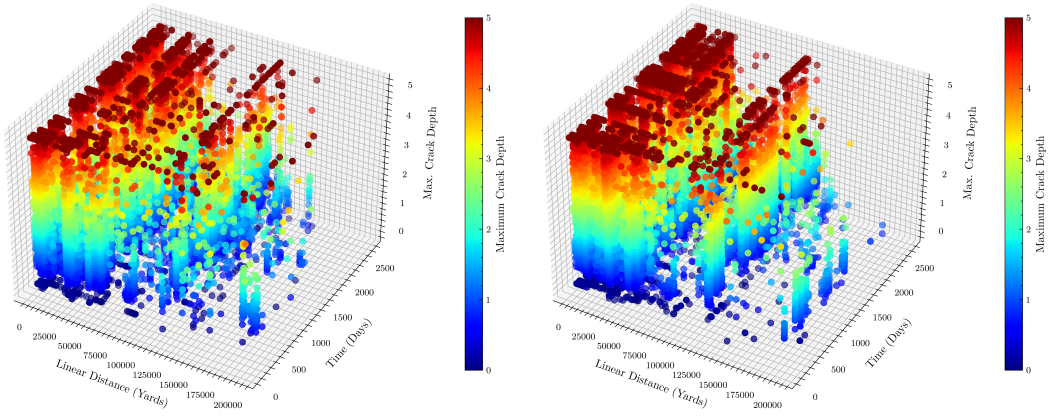
FIGURE 4.5: Sperry RSU Probe Layout (Whitney, 2020)

TABLE 4.3: Network Rail Planned and Maximum Intervals Between Inspections using Eddy Current Testing, Table 11 from (Network Rail, 2018c)

Track Category	Planned Interval	Maximum Interval
1A	8 weekly	18 weeks
1	8 weekly	18 weeks
2	16 weekly	36 weeks
3	26 weekly	60 weeks
4	26 weekly	60 weeks
5	26 weekly	60 weeks
6	52 weekly	130 weeks

4.3.1 Response Variable

In order to answer the central questions driving this study, a suitable *response variable* is required. Since the Eddy Current Data does not track the progression of individual cracks, it is not possible to directly study rates of propagation of specific cracks. Instead, the Roller Search Unit (RSU) captures the *deepest* crack detected over 1 yard for each of the 10 probes. So, instead of considering the crack data as individual events, the output data is an indicator that describes the degree of damage in each yard of track. Figure 4.6 shows a three-dimensional illustration of a set of raw Eddy Current data for the left and right rails. The x axis represents the spatial domain (linear distance (yards)), the y dimension time (days) since the first inspection in the dataset, and the z axis is the maximum probe measured crack depth (mm).



(A) Left Rail (B) Right Rail

FIGURE 4.6: Spatio-Temporal Illustration of 1 Yard Maximum Crack RCF EC Data

For this study the network is divided into 11 yard segments and the 1 yard cracking data are aggregated over these track segments. Cracks are summed according to their severity category (defined in Table 4.2) to generate $L_{(s,t)}, M_{(s,t)}, H_{(s,t)}, S_{(s,t)}, VS_{(s,t)}, Tot_{(s,t)}$ which represent the total number of Light, Moderate, Heavy, Severe, Very Severe and Total cracks in each segment s at time t (spatio-temporal grid cell (s,t)). Additionally, a custom damage index is generated for each segment which is a function of crack density and severity, $DI2$ = weighted sum of cracks where each weight is dependent on the severity of the crack, the weights are doubled with severity such that the resulting damage index $DI2$ can distinguish between cracks of differing severities. $DI2$ is thus calculated as follows:

$$DI2 = \sum_{s}^{s+1} L + 2(\sum_{s}^{s+1} M) + 4(\sum_{s}^{s+1} H) + 8(\sum_{s}^{s+1} S) + 16(\sum_{s}^{s+1} VS) \tag{4.1}$$

This method of spatial discretisation is important in the realm of data-driven modelling and can lead to improved predictive accuracy of models trained on this

data (Cao Feng et al., 2014; Liu et al., 2002; Yang, 2007). Discretisation is applied for the following purposes:

1. Reducing Noise due to positional uncertainties,
2. Enables the integration of multiple data sources. To formulate data-driven tasks, leveraging of other significant data sources is important, especially where data is captured in numerous systems operating at different ranges and frequencies. Spatial discretisation assists with this problem and any discrepancies in position across sources,
3. Reducing the problem sample space: spatial discretisation bounds the sample space and hence reduces problem complexity, instead of predicting a response variable for 1 yard, we increase the interval to 11 yards.

4.4 EC Preliminary Data Exploration

This section presents a summary of preliminary analyses performed on the *raw* Eddy Current data for TL1 and TL2. It aims to facilitate a general understanding of the data under study. The raw data is extracted directly from the Rail Defect Management System (RDMS).

4.4.1 EC Data Overview

A summary of the range of both study routes is provided in Table 4.4, the 'Distance Range' begins from the Route Origin (London Liverpool Street and London Waterloo for TL1 and TL2 respectively), the measurement time range relates to the period of EC inspections[1]. As previously described the frequency of inspection depends on the Track Category and the previously measured condition of the rail; if cracks require monitoring then they are measured at a higher frequency.

It appears that although Track Length 1 (TL1) covers a longer distance than Track Length 2 (TL2), there are more cracking events detected in TL2. This may be a result of the EC measurement frequency (more inspections will result in more cracks being found), or that TL2 has higher propensity for RCF cracking.

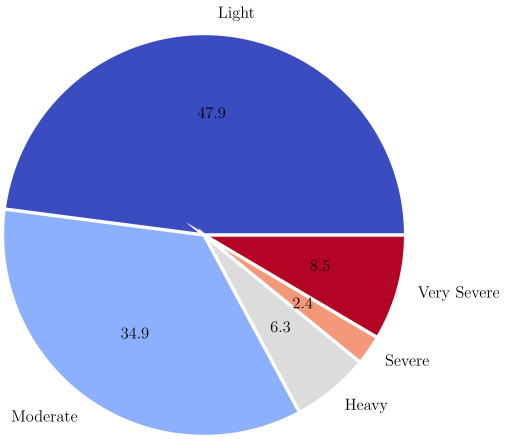
The distribution of damage by severity is fairly consistent between the two routes as illustrated by Figure 4.7a and Figure 4.7b, with more than three quarters of the cracks observed being of Light and Moderate Severity. In both cases, the RCF severity

[1]EC inspections were introduced to the UK network predominantly in 2015, although the system was piloted in some regions in earlier years.

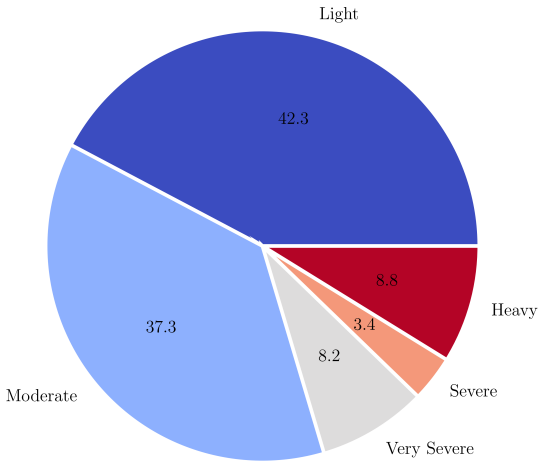
TABLE 4.4: Raw EC Data Overview

	TL1: LTN1, 2100	TL2:BML1, 2100
Total Observations	118188	134933
Distance Range	576 - 200337 yards	501 - 136938 yards
Measurement Time Range	2015-10-07 - 2022-08-31	2015-09-09 - 2021-07-08

category with the third highest presence in the data is 'Very Severe', this is likely because 5mm is the maximum readable depth for the equipment, so all cracks deeper than this are also regarded as 'Very Severe'. Many cracks will be reported as 5mm depth but will in fact cover a wider range of depths greater than 5mm. Further, for both track lengths, the category with the lowest number of measured cracks is 'Severe'.



(A) TL1



(B) TL2

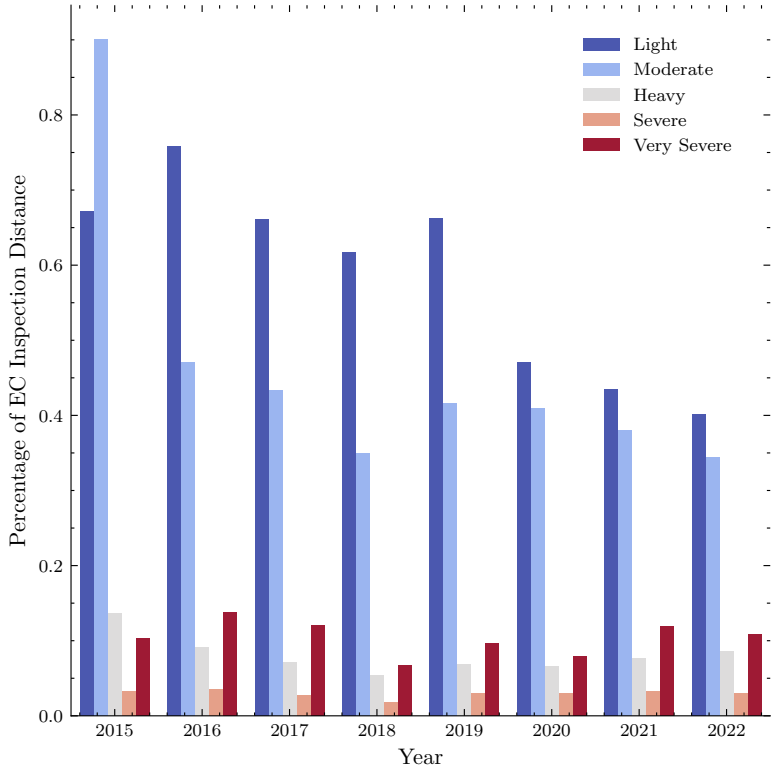
FIGURE 4.7: Distribution of Cracks by Severity by Percentage

4.4.2 EC Damage Over Time

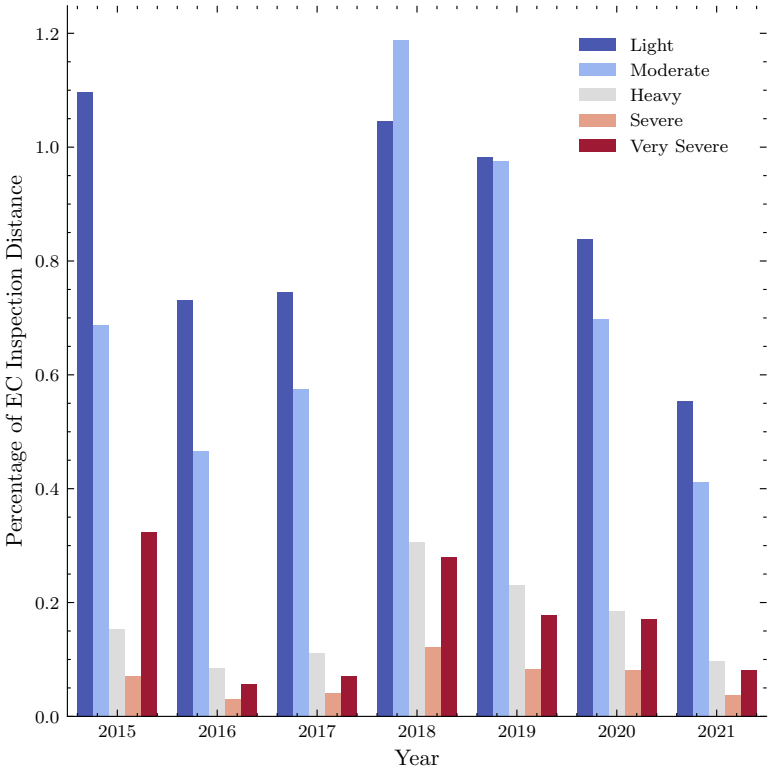
A summary of how damage is progressing with time is presented in this section, Table 4.5 and Table 4.6 summarise the number of L, M, H, S, VS and Total cracks found per year. The table also includes the total number of inspections carried out, and the total yards covered by the eddy current monitoring system each year for the Left Hand Rail (LHR) and Right Hand Rail (RHR); these quantities assist with explaining the considerable variation in the number of cracks each year. An increase in total cracks per year could be a result of increased levels of damage to the rail due to increased traffic, reduced interventions, etc.., or it could simply be a result of more inspections being carried out; generally speaking, more inspections result in more cracks being detected. Noticeably, TL2 has a greater number of inspections carried out per year than TL1. To account for this feature of the data, Figure 4.8a and Figure 4.8b illustrate the rail cracks detected per year in each severity category as a percentage of the total distance covered by the EC monitoring system that year. For both lengths of track an overall trend of decreasing RCF cracking over time is observed. This downward trend may be a result of improvements in intervention practices and RCF management, or due to reduced inspection frequencies or changes in the detection method (such as threshold modifications).

TABLE 4.5: Number of cracking events by year: Track Length 1 (TL1)

Year	L	M	H	S	VS	Total	No. Inspections	Dist. LHR	Dist. RHR
2015	3350	4495	683	164	513	9205	4	248954	249902
2016	13018	8087	1559	598	2365	25627	12	827189	889463
2017	5088	3334	547	208	924	10101	8	389137	381301
2018	7785	4407	673	220	848	13933	10	617215	645290
2019	11959	7508	1231	538	1744	22980	12	888431	918732
2020	5212	4529	724	327	875	11667	8	552337	553642
2021	6193	5429	1100	461	1695	14878	12	702039	723898
2022	4058	3475	872	301	1091	9797	10	522107	489210



(A) TL1



(B) TL2

FIGURE 4.8: Illustration of Damage Evolution over Time

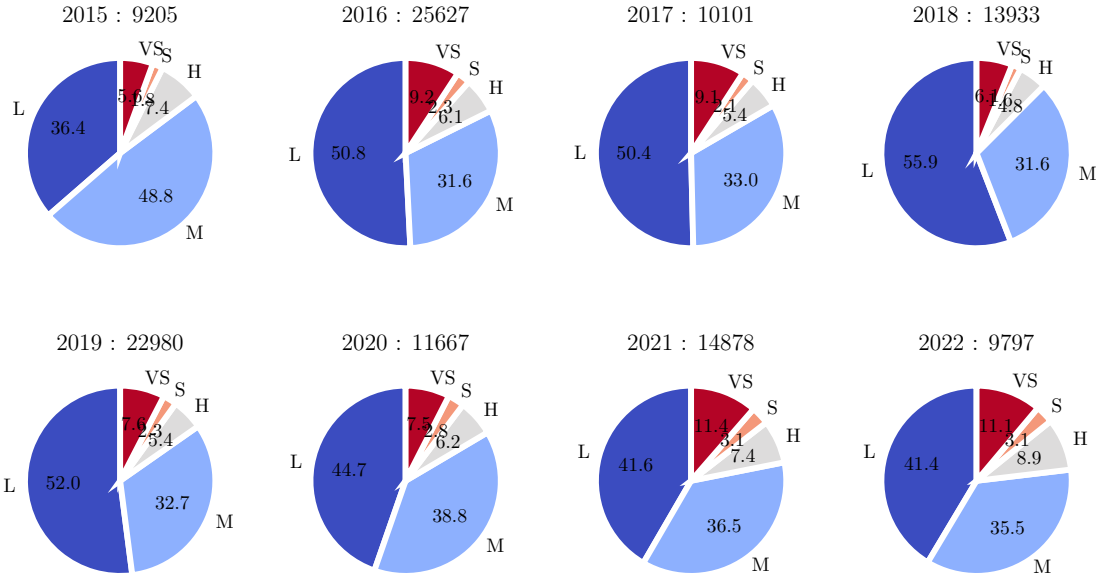


FIGURE 4.9: Severity of Cracking Events over Time: LTN1

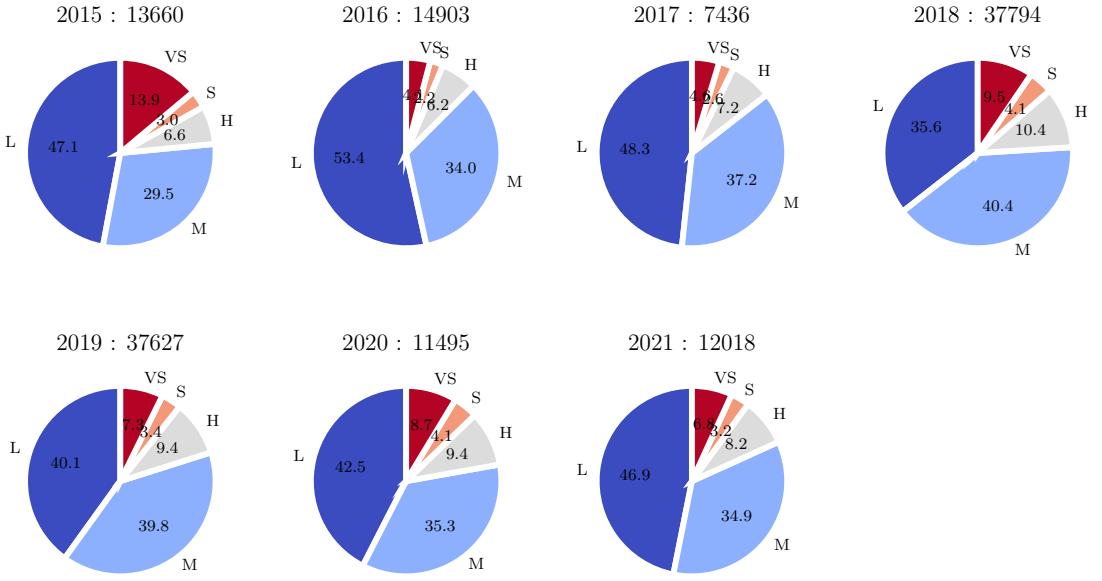


FIGURE 4.10: Severity of Cracking Events over Time: BML1

TABLE 4.6: Number of cracking events by year: Track Length 2 (TL2)

Year	L	M	H	S	VS	Total	No. Inspections	Dist. LHR	Dist. RHR
2015	6428	4026	896	411	1899	13660	15	296951	289296
2016	7965	5074	923	332	609	14903	30	496217	592013
2017	3592	2769	537	193	345	7436	12	245949	235789
2018	13438	15279	3927	1559	3591	37794	31	663490	621644
2019	15077	14980	3544	1282	2744	37627	34	752637	782539
2020	4883	4060	1080	477	995	11495	19	301505	280635
2021	5633	4194	986	382	823	12018	25	508908	508526

4.5 EC Data Properties

Following the preliminary analyses, a number of notable properties were uncovered from the EC data, these are described in the following section.

4.5.1 Presence of Space and Time

Eddy Current RCF cracking data have both temporal and spatial dimensions, broadly data with these characteristics are known as Spatio-Temporal (ST) data. ST data are found largely in fields such as climate and environmental science, crime, medical research and transportation, and their scope are increasing exponentially. The data differ from traditional numeric or categorical data due the complexity of ST data types, the presence of dependencies between instances (*autocorrelation*) and the variation of statistical properties with space and time (*heterogeneity*). Many widely used statistical data mining techniques, such as linear regression, rely on the assumption of independent and identically distributed (i.i.d) measurements, and therefore may have limited applicability when modelling ST data (Hamdi et al., 2021).

According to Tobler's first law of geography, "Everything is related to everything else, but near things are more related than distant things" (Tobler, 1970). For instance, the weather 1 mile away is likely to be more similar than 10 or 100 miles away. Yet it is an observation that can equally be extended to the temporal dimension, for example the weather tomorrow is likely to be similar to the weather today, more so than the weather a week or a month ago. These phenomena are referred to as spatial and temporal dependence, or are termed spatio-temporal autocorrelation when they occur in both dimensions. Spatio-temporal data samples further tend not to follow an identical distribution across the entire spatial and temporal domains, known as *heterogeneity*, or non-stationarity. Instead, different geographical regions and moments in time may have distinct distributions. Heterogeneity can further arise where data is

available from different sources and on different spatiotemporal scales (Amato et al., 2022).

Since many statistically based techniques; such as Linear Regression, assume *independence among observations* and *identical distributions* (the i.i.d assumption), the application of these techniques may not be well suited for this type of data. Hence, whilst analysing these data it is important to consider the effect of autocorrelation and variations in statistical properties and how these factors may impact the analysis.

Figure 4.11 represents a spatio-temporal density plot for RCF cracking. The xy axes represent the spatial and temporal domains respectively, markers indicate the location of observations, and the depth of background colour indicates the density of observations (darker blue = high density, lighter blue = low density). The data exhibit areas of densely clustered cracks, and further areas of no cracking, this is intuitive since neighbouring segments have similar characteristics which drive crack formation. Further, in the temporal domain the condition of the track segment at time $t+1$, will be related to the condition of the segment at time t. The obvious exception to this, however, is where interventions have occurred.

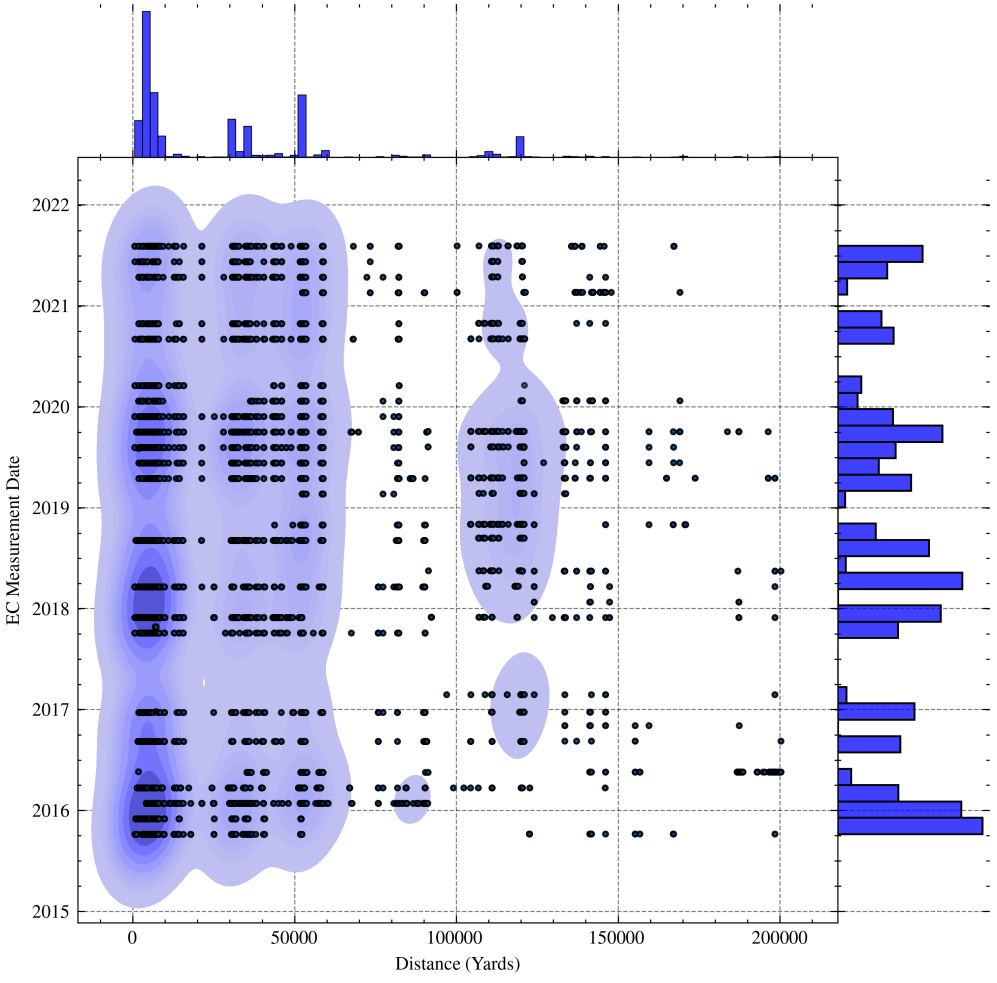


FIGURE 4.11: Distribution of EC Crack Segments: Left Rail

4.5.2 Inspection Frequencies

There are two predominant challenges with respect to the frequency of data collection:

- **Low Inspection Frequencies:** The time between inspections can vary from a few weeks up to 6-12 months between measurements. The interval depends on the category of track and the previous measurements at the site. Sparse measurements in the temporal domains make it difficult to ascertain temporal patterns and presents challenges when assessing the effectiveness of rail interventions.
- **Irregularly spaced measurements in space and time:** Controlled or synthetic environments enable data to be collected such that sensors are placed at regular spatial intervals, and that data is sampled at regular time intervals (i.e. every hour, every day, every week at the same time). This provides opportunities for leveraging novel formulations for spatio-temporal data mining (Atluri et al., 2017). For example, data instances can be expressed as 2D- time series data (vector representation, for each sequence), 3D- cross-sectional data (matrix representation at each time), 4D- spatio-temporal data (tensor representation). Where intervals are irregular, estimation of missing data is required to formulate instances as above, and therefore the data must be represented as points.

  Figure 4.12 shows the yards covered by the UTU, with green indicating that the Eddy Current equipment has successfully tested the track, and red representing testing failure. The figure also demonstrates how measurements are distributed in the temporal and spatial dimensions:

  - **Temporal:** There are some measurements which occur close together (approximately 6 weeks), whereas others have gaps $>$250 days, for example in 2017 and 2020. TL1 is predominantly 'Track Category' 1 and 2 and therefore has a planned EC measurement interval of 8 weeks (56 days), and maximum of 18 weeks (126 days) and so this is unusual. After discussions with NR staff (Personal Communication, Brian Whitney- Network Rail-Track Technical Expert, 2022) it is understood that this is largely a result of: a) the gradual addition of the Sperry RSU to all UTUs, which has only occurred more recently, gaps in the data are likely because the monitoring was carried out using a UTU with no EC system, or b) during the COVID-19 pandemic and due to social distancing rules, ultrasonic and EC operators were unable to work alongside each other, ultrasonics were seen as the priority and therefore EC measurements were not taken.
  - **Spatial:** In the spatial dimension three regions with differing inspection frequencies are distinguishable, 0-100,000 yards, 100,000-120,000 yards, and 120,000-210,000 yards with a degree of overlap, these overlapping sections

may cause some difficulties in the analysis particularly if they are spatially misaligned.

FIGURE 4.12: EC Tested/ Untested: Left Rail

4.5.3 Prevalence of Zeros/ Sparse Data

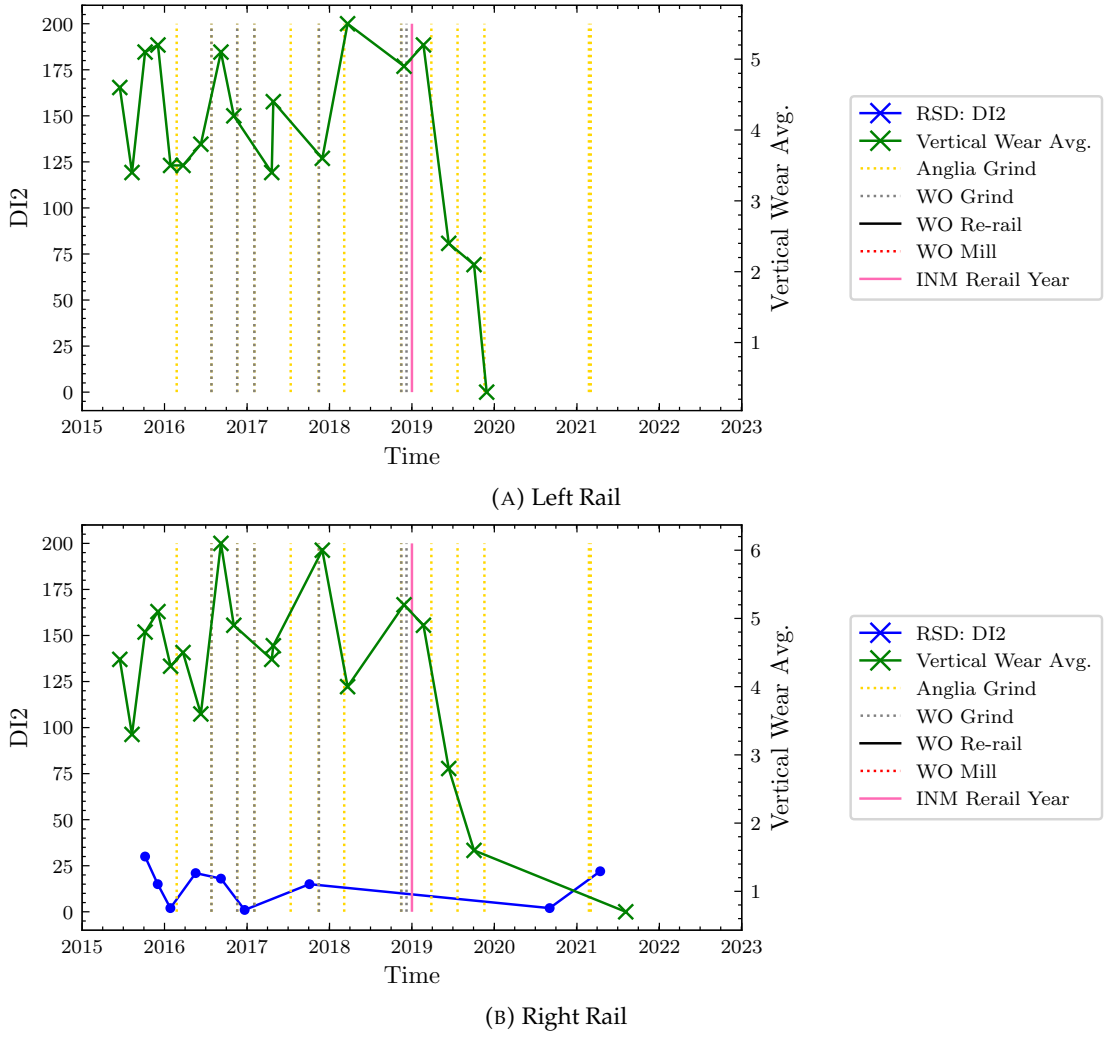
Notably, the majority of track sections present no cracking at all, and therefore there are large areas of the track with no data. Additionally, where cracks are not detected, this data is not reported by the system, it must be inferred by the lack of data, which could also be attributed to absence of measurements at the position/time, or system failure in this section, as opposed to **no** RCF.

The regions in the Figure 4.11 without scatter points indicate either missing observations (the track was not monitored) or a lack of RCF (the track was monitored and no RCF was detected). Some cases can be cross-checked with the UTU tested data shown in Figure 4.12. The red points indicate where the UTU has travelled but the EC testing failed, the green points represent successful recordings, and the yellow points represent the overlaid EC observations.

4.5.4 Interventions

Since EC data is collected directly from the operational environment, not only will it represent a system of rail degradation due to RCF, but also will be subject to changes from maintenance interventions such as re-profiling and rail replacement. When observing a segment of track it may appear as if the track is degrading, and then spontaneously it improves. It is assumed from the data that this is a result of grinding, milling, spot replacements or full track renewal, however care must be taken since this could also result from poor spatial alignment and other sources of noise.

Figure 4.13a and 4.13b show the 1D time series for the left and right rail at position 11,176 yards, with interventions represented by vertical lines, the rail surface damage by the green series, and vertical wear by the blue. The left rail has no recorded measurements at this location, but the right rail presents low levels of RCF. Following interventions such as grinding or milling, it is anticipated that the levels of RCF would reduce, or at least show decreasing degradation rates, and the vertical wear would increase. On the other hand, after rail replacement it is expected that RCF and vertical wear would dramatically change. The rail replacement data states that a replacement occurred at this location in 2019, although it is not known *when* in 2019 that this occurred. In this study all rail replacements are allocated to January 1st of the year. The effect of rail replacement on the damage indicator $DI2$ at any point in 2019 is difficult to detect since there are only measurements from 2017 and 2020. The wear data on the other hand shows four measurements in 2019 with large drops in vertical wear between measurement 1 and 2, and also between 3 and 4. It is therefore likely that the replacement also occurred in the later half of 2019. Grinding data is also presented, with data from the Anglia grinding database represented in yellow, and the work order grinding data in grey. However there doesn't appear to be any consistent effect on the wear or RCF data, in fact in some instances the vertical wear appears to reduce following a recorded grinding.



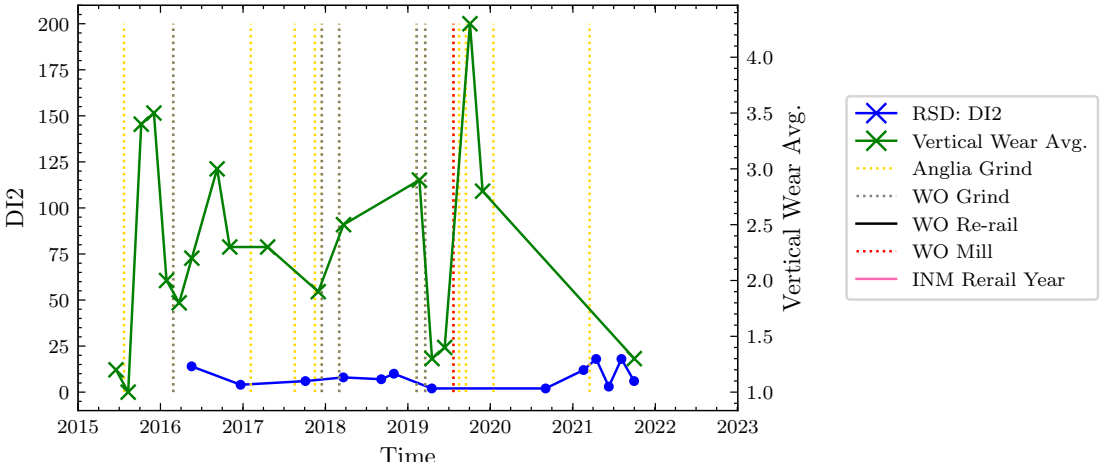
(A) Left Rail

(B) Right Rail

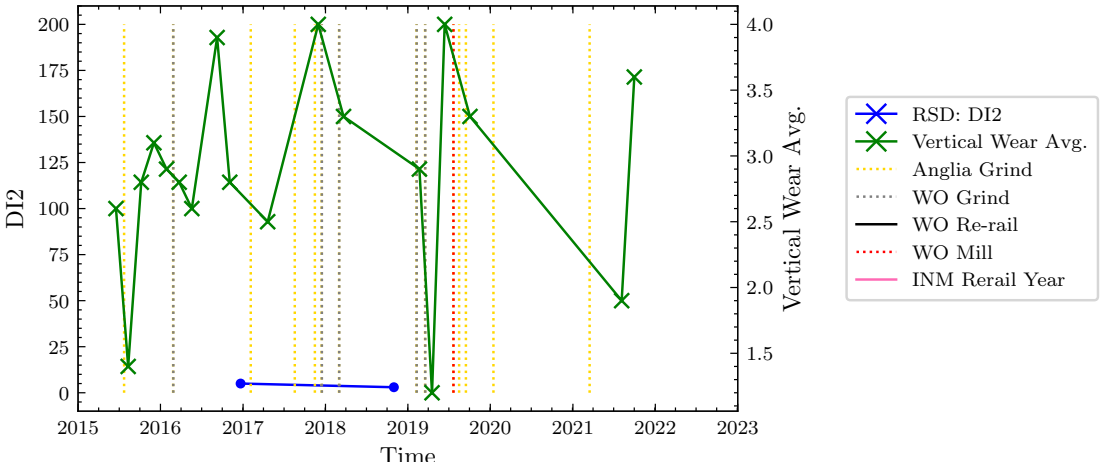
FIGURE 4.13: Illustration on the effect of rail replacement for TL1 at 11176 Yards

Additionally, Figure 4.14a and 4.14b illustrate a section of rail where there is reported milling in mid-2019. Once again it is difficult to detect a significant change in the RCF data due to temporal frequency and relatively low levels of damage, however the wear data confirms (through a spike in vertical wear) that rail has very likely been milled as stated. The reliability of the grinding data must again be questioned, since there is no obvious correlation between the wear or RCF data and the occurrence of grinding.

In summary, there are some high levels of variation in the temporal patterns of the EC data which is likely as a result of interventions, and some measurement/ processing noise. The interventions data further appears to be inconsistent and of low temporal resolution, which is problematic if the analyses rely on this data. There are instances where re-railing is indicated but the wear/ RCF data does not support this, and vice-versa instances where a significant improvement is observed in the rail RCF damage and/ or vertical wear but is not recorded in the intervention data. There is the possibility of using a combination of wear and RCF data to infer the occurrence of



(A) Left Rail



(B) Right Rail

FIGURE 4.14: Illustration on the effect of Rail Milling for TL1 at 52591 Yards

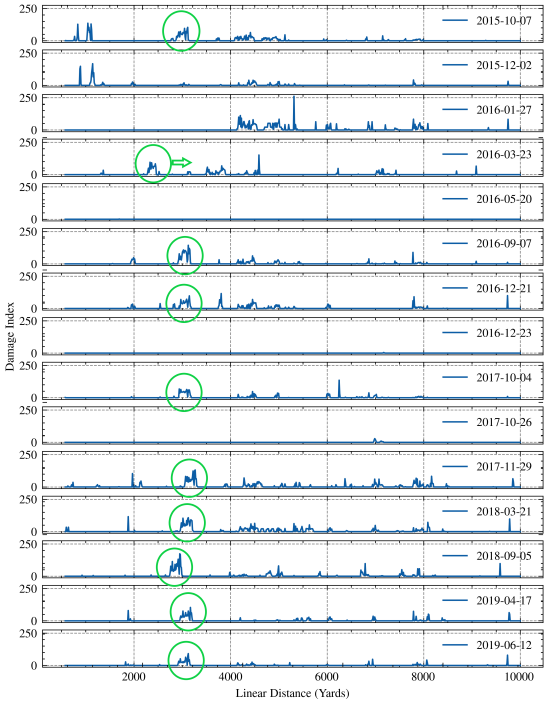
re-railing, as this does in some cases seem to be apparent, however the additional problem is the varying temporal frequencies of the wear and RCF data, and any spatial misalignments between the measurement sources.

4.5.5 Noise

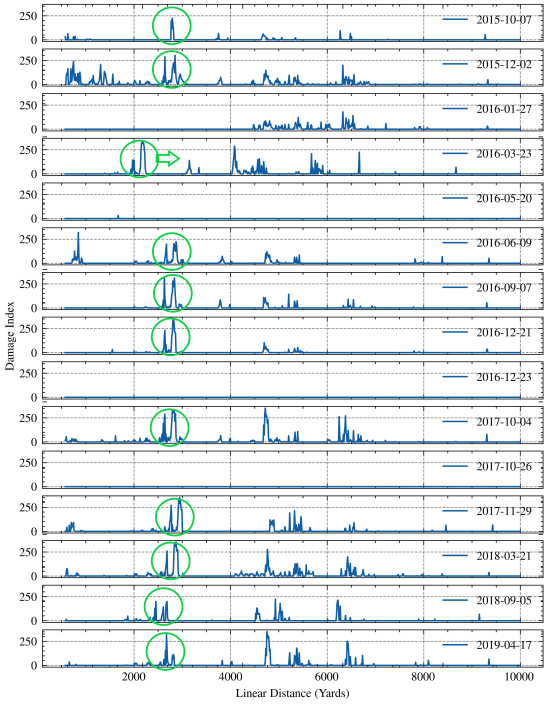
Georeferencing errors result in misalignments between cross sectional eddy current data, such that RCF cracks are reported to change location between measurement runs. The other challenge is the misalignment of data from different sources: information such as track curvature, cant, traffic density, locations of S&C and interventions data for instance all have different sources for determining their position on the track. Thus the problem is not only that RCF cracks will not align between runs, but that RCF cracks cannot be accurately positioned on the track.

Figure 4.15a and Figure 4.15b illustrate the geo-spatial misalignment between cross sections of EC data measured on separate occasions, for a subset of TL1 data. In most cases, the cross sectional peaks are clearly identifiable (shown in green), in particular for the right hand rail. However there are some instances, such as 2015-12-02, 2016-05-20, 2016-12-21 and 2017-10-26 where the data is sparse and therefore difficult to ascertain identifiable features to align. The 2016-03-23 cross section is clearly spatially shifted from some of the preceding and following cross sections, and should be moved some 500 yards farther down the track. These figures also indicate the localised regions of spatio-temporal correlation in the data, illustrating the similarities in cross-sections, and the clusters of RCF in space, the cross-sectional correlation can be utilised when re-aligning the data. Failure to re-align the data may result in highly anomalous results, especially when generating rates of change.

Further, there are numerous processing stages performed by the Sperry system on-board the UTU which generate the raw data made available for this study. Over the years, thresholds such as the lift-off distance (introduced in Section 2.3.2) which represent the distance of the probes above the rail in order to optimise for penetration ability vs. sensitivity to cracks may have changed which may present some inconsistencies over time. Additionally, there are reports of occasions where other conductive materials, such as remnants from ground rail interfere with the Eddy Current measurement system.



(A) Left Rail



(B) Right Rail

FIGURE 4.15: Spatial Misalignment

4.5.6 Modifiable Area Unit Problem (MAUP)

Scale effects in space and time are a challenging issue in spatio-temporal data analysis and mining (Venkateswara Rao, 2012). MAUP affects result when point-based measures are aggregated into districts, the resulting summary values are influenced by both the shape and scale of the aggregation unit.

As described in Section 4.3 the EC data in raw form can be represented by a point (d, s, t) where d represents the maximum crack depth over a 1 yard track section s, detected at time t. Whilst the temporal granularity will remain unchanged due to the temporal irregularity and low frequency of measurements, the data in the spatial domain are numerous, and the data at the 1 yard level of granularity are likely to result in high levels of noise and thus will be difficult to analyse. In the spatial domain therefore, aggregation is applied. However, the aggregation window will be constant and as low as practical to ensure minimum bias but also enabling a sufficient reduction in noise and the ability to integrate the data reliably with other sources.

4.6 Concluding Remarks

This chapter has introduced the data under study, some preliminary analyses, but most significantly the challenges and properties displayed in the Eddy Current data.

The Eddy Current data has dimensions in the spatial and temporal dimensions, and thus may present characteristics such as spatial and temporal dependencies (autocorrelation), and variation in statistical properties in space and time (heterogeneity). These features must be considered, particularly when making inferences using models that assume independence amongst observations, and further the use of validation schemes that also may make independence assumptions. Additionally, the data are irregularly spaced in time and space which means that learning instances for presentation to machine learning algorithms must be represented as 1D points rather than using novel representations that are available to spatio-temporal data such as time series (2D), rasters (3D) or tensors (4D). The irregularity may also cause issues with regard to correcting for other issues such as noise and misalignments. Some of the most significant problems with the Eddy Current data however include the presence of noise and interventions, which are directly addressed in the following chapter using alignment methods, smoothing and outlier detection. Spatial misalignment in particular presents an issue for analysing degradation rates, and therefore will inhibit any meaningful analysis and generate unreliable models if used in its raw form. Additionally the presence of interventions, and the inconsistent, unreliable data on rail replacements, grinding and other rail

maintenance activities mean that assumptions may be required for estimating when interventions have occurred in order to determine degradation rates.

In order to successfully implement a data driven methodology, it is fundamental that these challenges are addressed. The following chapter summarises the various pre-processing techniques that have been employed to satisfy these goals and ensure the data is suitable for extracting useful insights.

Chapter 5

Data Pre-Processing

Data pre-processing is a critical step in ensuring that the data are provided in a suitable quantity, structure and format for data mining. Data mining is a process of extracting knowledge from real data sets, which in reality are often scattered, noisy and even incomplete. The quality of these data can have a significant influence on subsequent analyses and data mining models, and thus various pre-processing steps have been developed to ensure the data meet the input requirements of the model, improve the relevance of the prediction target, and make model optimisation simpler (Li, 2019).

The proposed pre-processing methodology for the Eddy Current, Wear, Track, ACTRAFF and Defect data is depicted in Figure 5.1. Specifically, Eddy Current data processing is addressed in Section 5.1, whereas the additional data sources are covered in Section 5.2.

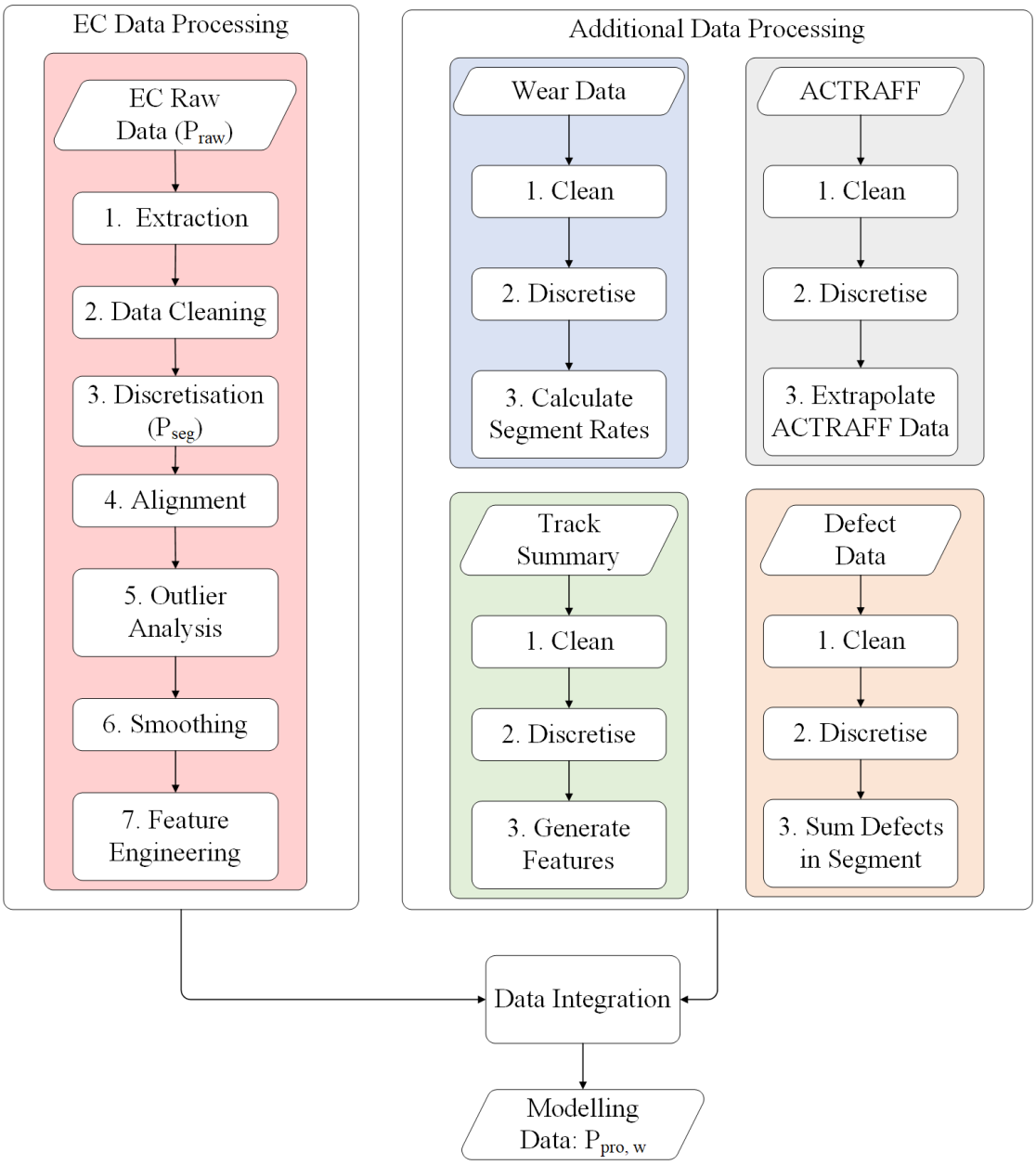


FIGURE 5.1: Data Pre-Processing Methodology

5.1 EC Data Processing

As described in Chapter 4, there are several characteristics of the Eddy Current Data which make the application of data mining techniques difficult and unlikely to yield useful results if applied directly. The purpose of this section is to describe some of the proposed techniques for addressing these problems, the raw data is denoted: P_{raw}, the discretised data: P_{seg}, and the final processed modelling data set is: $P_{pro,w}$ where w is the applied smoothing window. The full processing methodology is illustrated in Figure 5.1, here we describe the Eddy Current Data processing steps:

1. **Data Extraction:** The 1 Yard Historic EC Data are extracted from RDMS, we denote this dataset: P_{raw},

2. **Data Cleaning:** This step includes:

   - Conversions: Converting discrete data to categorical data types,
   - Missing Values: Identifying Not a Numbers (NaNs), which are values which are missing or undefined, determining a method of filling these values and also removal of duplicates,
   - Check Ranges: Ensuring data ranges are reasonable,
   - Date Compression: EC recording runs performed on consecutive dates for different lengths of track are combined in order to generate longer 'continuous' lengths of track data for analysis, and to reduce any difficulties with overlapping data when calculating rates. In order to combine signals that should be considered as occurring on the same date to improve the ease of analysis a time window must be chosen. We know that in most instances, the same site will not be reassessed more frequently than approximately 6 weeks (42 days), using this knowledge, and a preliminary look at the data 20 days was selected for this data set since it picked up the majority of clearly anomalous cases, however this may change depending on the input data set,
   - Timescale: Timestamps are converted to a continuous daily time scale, the earliest date in the dataset represents the first inspection for this sample, and is thus denoted 0 days. All the other dates are converted with reference to this first inspection date.

3. **Data Discretisation:** As noted in Section 4.3.1, the locations of cracks are described by linear yardage that assume a continuous value between the origin and total track length. i.e, cracks can be detected at 10 yards, 10.5 yards, 10.2 yards etc. Discretisation is applied in order to reduce the sample space, and hence problem complexity, and enable integration of independently collected data sources (such as the track spatial characteristics, etc.) using a common spatial reference frame. Thus each observation will represent an 11 yard spatio-temporal cell, containing Light, Moderate, Heavy, Severe and Very Severe cracks, which can be mapped to other spatial and temporal parameters. Note that the spatial dimension is regular but the temporal domain is irregular. 11 yards represents the lowest resolution for which other data sources are available, it is commonly used to discretise the NR network, and therefore is selected for ease of integrating with other data sources which are required for subsequent analysis.

   To discretise the data, the track is first divided into equally sized segments (11 yard segments) and the Eddy Current observations with common time stamps

are aggregated over each spatial segment $(s, s+1)$, where s represents the index for the starting linear position of a segment, and $s+1$ the end position. The aggregations applied to observations at time t, over spatial segment $(s, s+1)$ are as follows:

(a) Maximum Crack Depth- DI_{max}: Maximum Crack Depth (mm) of cracks in segment $(s, s+1)$)

(b) Total No. Cracks in Segment, DI_{count}: Total number of all cracks in segment $(s, s+1)$):

(c) Damage Indicator, $DI2$ is a weighted sum of cracks where each weight is dependent on the severity of the crack, this is intended to generate larger damage values where the severity is high in order to distinguish between high density and severity:

$$DI2 = \sum_{s}^{s+1} L + 2(\sum_{s}^{s+1} M) + 4(\sum_{s}^{s+1} H) + 8(\sum_{s}^{s+1} S) + 16(\sum_{s}^{s+1} VS) \tag{5.1}$$

where L, M, H, S, VS represent the total number of observations in each of the Light, Moderate, Heavy, Severe and Very Severe categories.

For example a row of data may look like:

TABLE 5.1: Example of Discretised EC Data

Start	End	L	M	H	S	VS	Sum	Max. Depth (mm)	$DI2$
11	22	1	3	0	4	0	8	4.2	39
22	33	2	3	1	2	1	9	5	44

The resulting discretised dataset is denoted as P_{seg},

4. **Alignment:** A major shortcoming affecting the EC measurements is geospatial misalignment. Largely this is historical due, in-part to some manual operations with regard to synchronising GPS signals with mileposts. Data cross-correlation is utilised to perform alignment between Eddy Current runs. Care must be taken with alignment parameters due to the temporal frequency of EC signals which results in potentially large run-run variations causing large shifts to be recommended to maximise total signal cross-correlation. The methodology is described further in Section 5.1.1,

5. **Outlier Analysis**: The Local Outlier Factor (LOF) is a distance-based approach in which the density of regions in the data are computed, and the instances in the low density regions are declared as outliers. Data points with high LOF have more sparse neighbourhoods and typically represent stronger outliers, unlike data points belonging to dense clusters that usually tend to have lower LOF values Lazarevic (2005). The methodology is described further in Section 5.1.2,

6. **Smoothing:** Also known as curve fitting, or low pass filtering, smoothing is intended to detect trends in the presence of noisy data, and includes techniques such as clustering, regression and binning (Alasadi, 2017). The outcome is essentially noise reduction, which, in its simplest form eliminates the extreme evaluates of the signal, giving in effect the reduction of average magnitude of the signal (Kowalski, 2018). Smoothing improves data quality by replacing the noisy irregular signal with the new smoothed signal which probably better describes the measured phenomena. The methodology is described further in Section 5.1.3,

7. **Feature Engineering:** In machine learning, a sufficiently descriptive feature set is critical for the explanatory or predictive power of a model. Feature engineering is the practice of constructing suitable features that lead to improved predictive and explanatory performance (Nargesian et al., 2017). Additional features derived from the EC data are described in Section 5.1.4,

5.1.1 EC Alignment

Georeferencing errors in the Eddy Current equipment are known to result in spatial misalignment between recording runs. For instance, consider subsequent damage measurements D_{s_i,t_j}, which represents an observation occurring in spatial cell s_i at measurement time t_j, and $D_{s_i,t_{j+1}}$ which corresponds to the subsequent observation in time recorded in spatial cell s_i. In the presence of spatial misalignments between recording runs, these observations may have in fact occurred in different spatial cells. If this is occurring repeatedly in the data, it will introduce misleading patterns and results which are calculated from analyses and learning algorithms based on these data.

5.1.1.1 Theory

Techniques developed in the realm of signal processing have been utilised in many domains to re-align signals. These techniques depend on the assumptions regarding the nature of the misalignment, and differing measures for determining signal similarity. For example, temporal signals can be aligned using techniques such as dynamic time-warping, which assume shifts and distortions in time to determine the optimal alignment between two sequences under certain restrictions (Müller, 2007). However in this case, we have spatial cross-sections of data corresponding to different measurement times, which are believed to be constantly shifted in space, rather than non-linearly distorted.

A basic measure of similarity between two signals as a function of the space or time-lag between them is cross-correlation. The cross correlation of the discrete time signals $X^j[m]$ and $X^k[m]$ for a spatial or time lag n is expressed as:

$$R_{X^jX^k}[n] = \sum_{m=-\infty}^{\infty} X^j[m]X^k[m+n] \tag{5.2}$$

The maximum (peak) cross-correlation occurs when the lag between the two signals is zero, and indicates the initial shift between the two signals. When the lag between the signals is zero, the positive peaks of signals $X^j[m]$ and $X^k[m]$ are aligned, and thus maximally contributing to $R_{X^jX^k}$. Similarly, when the negative peaks of signals $X^j[m]$ and $X^k[m]$ align, they also make a positive contribution to $R_{X^jX^k}$. Thus, the maximum of the cross-correlation function will occur when the two signals overlap, and will depend on the magnitude of the two signals, i.e., the peak-to-peak magnitude, or energy, of both signals (Potas et al., 2015).

The selected approach uses a 'naive implementation' of cross-correlation analysis to align *multiple* signals, where, rather than calculating the cross-correlation between all signals, the relative shifts are estimated with respect only to a *Reference signal*, X_β.

5.1.1.2 Application to EC Data

For each track length TL1 and TL2 a methodology is applied to align multiple Eddy Current signals, which will be termed spatial cross-sections or signals interchangeably. Note that we refer here to the discretised 11 yard Eddy Current data (P_{seg}).

Rather than the simple illustrated case of aligning two cross sections of data, the aim here is to align N EC signals. Assuming that these each of these N signals is shifted with respect to the others, it is not possible to estimate the set of absolute drifts $\delta_1, \delta_2, ..., \delta_N$. Rather, the relative drift of the jth and kth signal $\delta_{jk} = \delta_j - \delta_k$ is estimated. Thus, in this study we estimate the relative shifts of the $N-1$ remaining signals with respect to a selected reference signal, X_β.

The reference signal, X_β will determine the alignment of all other signals, and therefore it is important that it carries sufficient information to enable effective alignment performance. The alignment procedure here includes a method for determining a suitable reference signal which involves calculating various signal characteristics such as the damage variation and the number of peaks in the signal.

Additionally the procedure must address the problem that each signal covers different portions of the track length. Some may cover 0-50000 yards, whilst others only cover 20000 - 50000 yards. The data are zero-padded, and this will result in low

cross-correlation between these signals in the regions where one signal is assumed to be zero.

To account for this the signals are divided into spatial clusters before alignment, and each shift is constrained using a maximum shift threshold, Δ.

The parameters for these procedures, such as the *optimum reference signal* $X_{\beta}*$, *the optimal number of partitions* (clusters) required to divide the data $\kappa*$, and the *optimal maximum shift* $\Delta*$ must be estimated empirically from the data in order to maximise the quality of the resulting alignment of all signals. A popular measure in the literature for establishing the alignment accuracy between two signals is the Pearson's Cross Correlation coefficient (r) (Korifi et al., 2014). Given paired data $(x_1, y_1), ..., (x_n, y_n)$ consisting of n pairs, r_{xy} is defined as:

$$r_{x,y} = \frac{\sum_{i=1}^{N}[(x_i - \bar{x})(y_i - \bar{y})]}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2 \sum_{i=1}^{N}(y_i - \bar{y})^2}} \tag{5.3}$$

where n is the sample size, x_i, y_i are the individual observations indexed with i, $\bar{x}$ and $\bar{y}$ are the sample means for x and y (Berman, 2016).

In this study; to determine the quality of the alignment between the N signals, the *average* of the Pearson's coefficient for each signal with reference to the X_{β} is calculated, ($\rho = \sum R_j / N$). The reference signal and maximum shift threshold are selected to maximise this Performance Indicator (ρ). The optimum number of clusters κ^* can subsequently be determined by maximising the average of our performance indicator ρ across the clusters, i.e. ($\sum_{i=1}^{\kappa} \rho / \kappa$).

The full procedure for aligning Eddy Current cross sectional data is described below and in Figure 5.6. Note that we assume for each cluster the shift is constant across the cluster, and further that the left and right rail shifts are the same:

1. **Select input data for alignment:** the data is first segmented into clusters of observations which cover similar portions of the track. For this study the procedure will be followed using 1, 2, 3, 4, 5 and 6 as the total number of clusters κ, this level of segmentation should be sufficient to maximise alignment accuracy:

   - Choose number of sub-sections (clusters) from $\kappa = [1, 2, 3, 4, 5, 6]$
   - Use the K-Means clustering algorithm to compute clusters in the data based on the linear distance of each observation, an illustration of the resulting clusters where $\kappa = 3$ are shown in Figure 5.2.

   The following steps (2 and 3) are performed for each total number of clusters $\kappa = [1, 2, 3, 4, 5, 6]$, and each cluster $i \in [0, ..\kappa]$

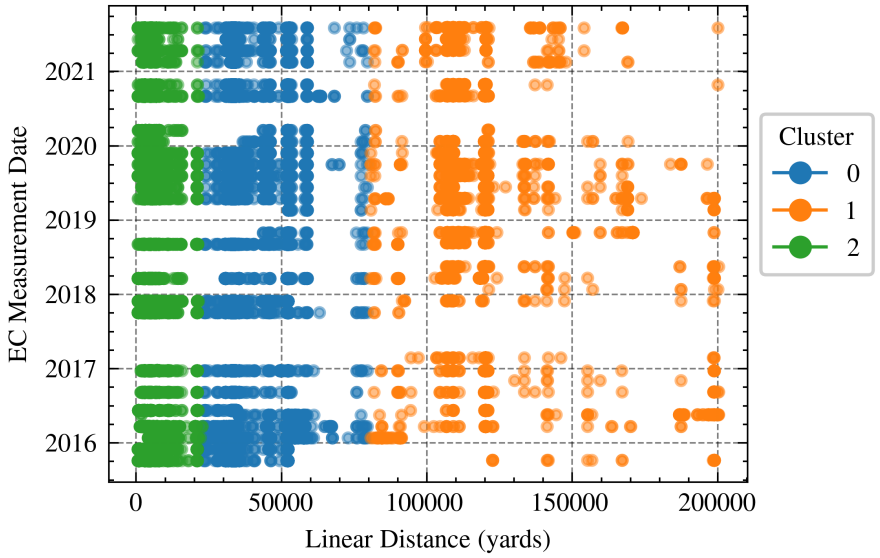


FIGURE 5.2: Data Segmented for Alignment: 3 Clusters

2. **Identify 3 'Best' Reference Signals** $[X_{\beta_{i1}}, X_{\beta_{i2}}, X_{\beta_{i3}}]$**:** the Reference signal must contain sufficient information to enable effective signal alignment,

   - **Compute Signal KPIs:** Key Performance Indicators (KPIs) are derived for each signal in cluster i in order to determine the 'best' signal for using as a reference, these are as follows:
     - **Total No. Signal Peaks:** a peak or local maximum is defined as any sample whose two direct neighbours have a smaller amplitude, the basis of using this as an indicator of a 'good' reference signal is that a signal with a high number of peaks is easier to align. In this study, based on some basic analyses, a peak is determined to exceed a threshold of DI = 3, and separated by 300 track segments (3,300 yards),
     - **Difference between Start and End Positions:** a reference signal should span a reasonable distance,
     - **Observation Count:** A reference signal should have a large number of observations which can be aligned to,
     - **Signal Variances:** $Var(X_1), Var(X_2), ...Var(X_N)$, a signal with very low variation is difficult to align to since the algorithm will find it difficult to distinguish signal characteristics,
   - **Scale Data:** Scale the performance indicators between [0, 1] using a maximum value scaler,
   - **Rank Signals:** Determine the rank of each signal based on the 4 parameters in order of importance: *[Total No. of Peaks, Signal Variance, Difference between start and end yardages, Observation count]*,
   - **Assign 3 Reference Signals for cluster** i**:** the top three ranked signals $X_{\beta 1_i}, X_{\beta 2_i}, X_{\beta 3_i}$ are selected for input to the alignment procedure,

3. **Align Signals in Cluster i, for each of the 3 reference signals:**

   - **Compute Shifts:** For each maximum shift:
     $\Delta \in [50, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000]$, and
     $X_{\beta_i} \in [X_{\beta 1_i}, X_{\beta 2_i}, X_{\beta 3_i}]$, the objective is to determine the $N-1$ spatial shifts $\delta_1, ...\delta_{N-1}$ which optimise the left and right rail cross correlation functions between signals X_j and X_k:

$$r_{j,k} = \frac{\sum_{s=1}^{S}[(X_j^s - \bar{X}_j)(X_k^s - \bar{X}_k)]}{\sqrt{\sum_{s=1}^{S}(X_j - \bar{X}_j)^2 \sum_{s=1}^{S}(X_k - \bar{X}_k)^2}} \tag{5.4}$$

     where j represents the index of the reference signal, and k the index of the remaining $N-1$ signals to align, subject to the constraints:

$$\begin{aligned} \delta_{left} &\leq \Delta \\ \cap\, \delta_{right} &\leq \Delta \\ \cap\, \delta_{left} &= \delta_{right} \end{aligned} \tag{5.5}$$

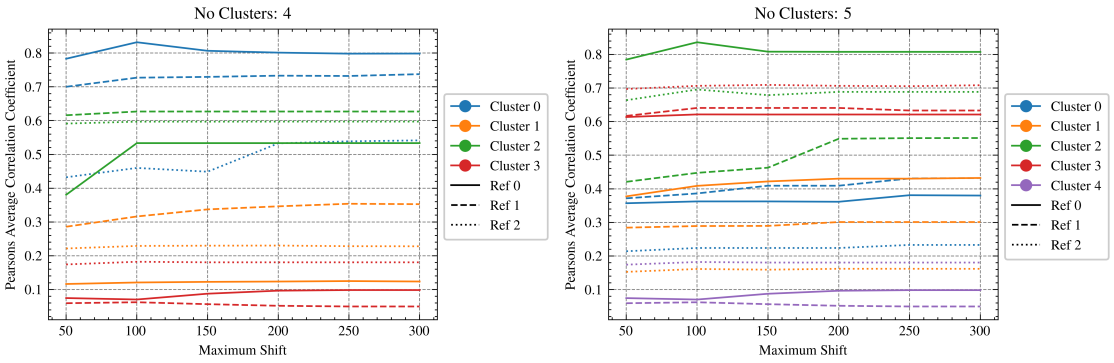
     where δ_{left} and δ_{right} are the calculated left and right hand shifts. These constraints reflect the limit on the upper bound for the spatial shift of the signal, and ensure that both the left and right rails are shifted by the same amount,

   - **Shift each Signal** shift each of the $N-1$ signals in time by calculated shifts $\delta_1, ...\delta_{N-1}$, and calculate the resulting vector of correlation coefficients for the left and right rails: $R[i, X_{\beta_i}, \Delta] : r_{\beta,1}, ...r_{\beta,N-1}$ where $\beta \in [1, 2, 3]$

4. **Calculate the performance for each set of parameters:** $[\kappa, \Delta, i, X_\beta]$**:** For each number of clusters, cluster index, reference signal and maximum shift, the Pearson's Cross Correlation coefficient is calculated between the reference signal and each of the remaining $n-1$ signals for each rail independently. These coefficients are mean averaged to determine an indicator for the left and right rails $\rho_{left} = \sum_{k=1}^{N-1} R_{k_{left}}/N$ and $\rho_{right} = \sum_{k=1}^{N-1} R_{k_{right}}/N$, which are then summed to generate a joint performance indicator ρ. Figures 5.3a and 5.3b illustrate these results for $\kappa = 4$ and $\kappa = 5$ respectively.

5. **Select Parameters which Maximise** $\sum_{i=1}^{\kappa} \rho/\kappa$**:** Select parameters $[\kappa, \Delta, X_{ref}]$ for the final alignment. Table 5.2 indicates the parameters which maximise ρ, and Table 5.3 subsequently indicates the average performance over the total number of clusters $\sum_{i=1}^{\kappa} \rho/\kappa$.

6. **Final Alignment of Data:** The final results indicate that the data should be segmented into 3 clusters ($\kappa = 3$) (see Table 5.3), alongside corresponding parameters for each cluster $i \in [0, 1, 2]$ as highlighted in Table 5.2.



(A) No. Clusters: $\kappa = 4$ (B) No. Clusters: $\kappa = 5$

FIGURE 5.3: Illustration of Performance Indicators for Alignment Parameters: X_β and Δ for each Cluster $i \in 1, ...\kappa$

Figure 5.4 and 5.5 illustrate the outcome of the alignment process for $\kappa^* = 3$, $i = 0$ for the left and right rail respectively, where the optimal maximum shift, Δ^* for this cluster is 100, and the reference signal is $\beta 2$ (which corresponds to the signal measured on 2017-10-04). From the figures, the success of the alignment procedure is clearly visible for signals dated 2016-03-23 and 2021-08-04.

TABLE 5.2: Performance Indicators for Optimal Alignment for $\kappa \in [2, 3, 4, 5]$

Total Clusters κ	Cluster Number i	Max. Shift Δ	Ref. Signal X_β	R_{left}: Left Rail	R_{right}: Right Rail	ρ
1	0	100	2	0.14	0.15	0.29
2	0	150	0	0.32	0.39	0.70
2	1	200	2	0.10	0.22	0.32
3	0	100	2	0.16	0.46	0.62
3	1	300	2	0.1	0.27	0.37
3	2	100	0	0.41	0.43	0.83
4	0	100	0	0.41	0.43	0.83
4	1	250	1	0.10	0.25	0.35
4	2	100	1	0.17	0.46	0.63
4	3	100	2	0.05	0.13	0.18
5	0	300	1	0.23	0.20	0.43
5	1	300	0	0.12	0.32	0.43
5	2	100	0	0.40	0.43	0.84
5	3	150	2	0.25	0.46	0.71
5	4	100	2	0.05	0.13	0.18
6	0	250	2	0.25	0.16	0.41
6	1	250	2	0.18	0.35	0.52
6	2	100	0	0.40	0.43	0.84
6	3	100	2	0.25	0.46	0.71
6	4	100	2	0.05	0.13	0.18
6	5	100	0	0.07	0.03	0.10

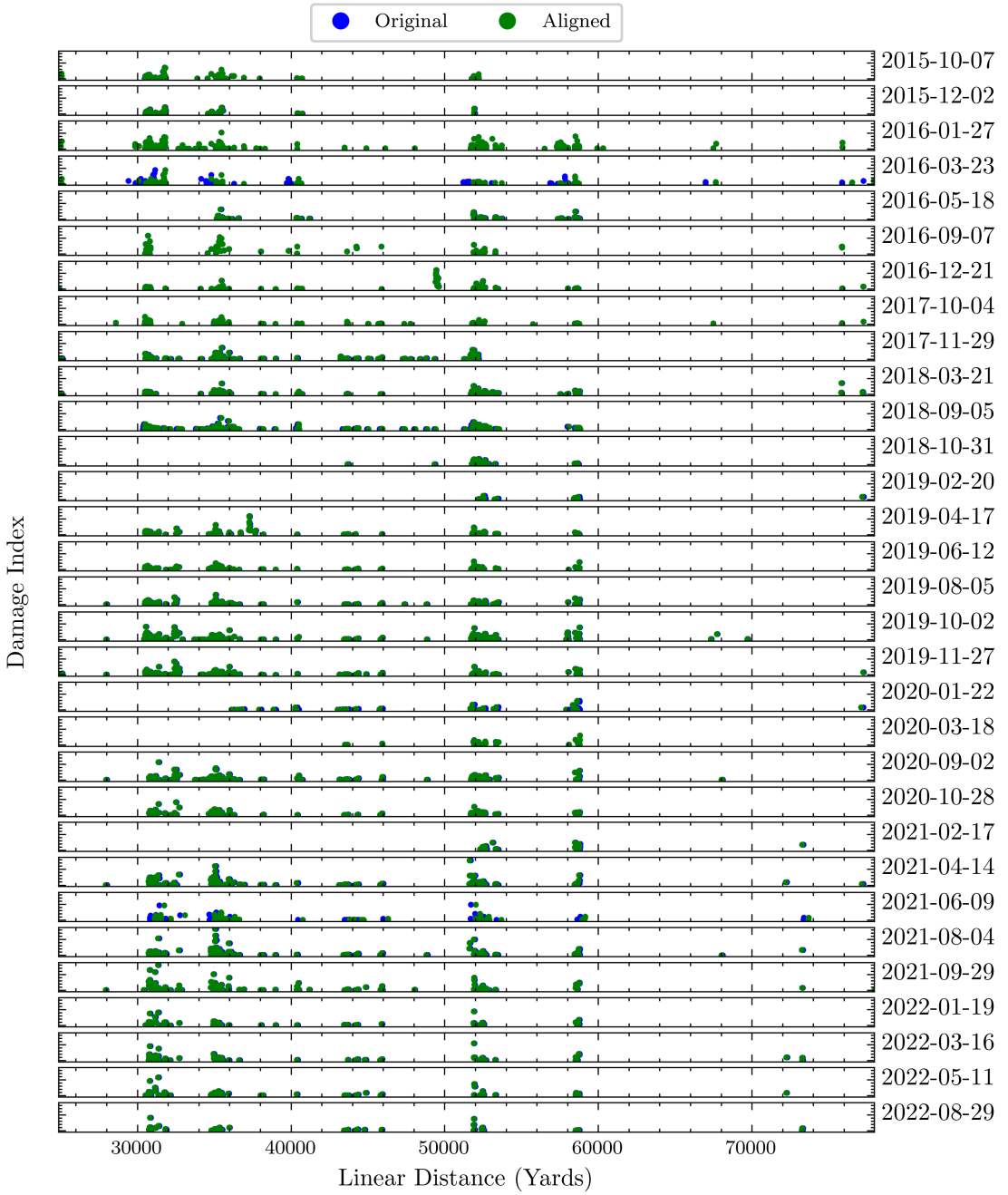


FIGURE 5.4: Optimal Alignment: Left Rail, Cluster i = 0, Max. Shift $\Delta* = 100$, Optimal Ref Signal ($\beta*$) = 2

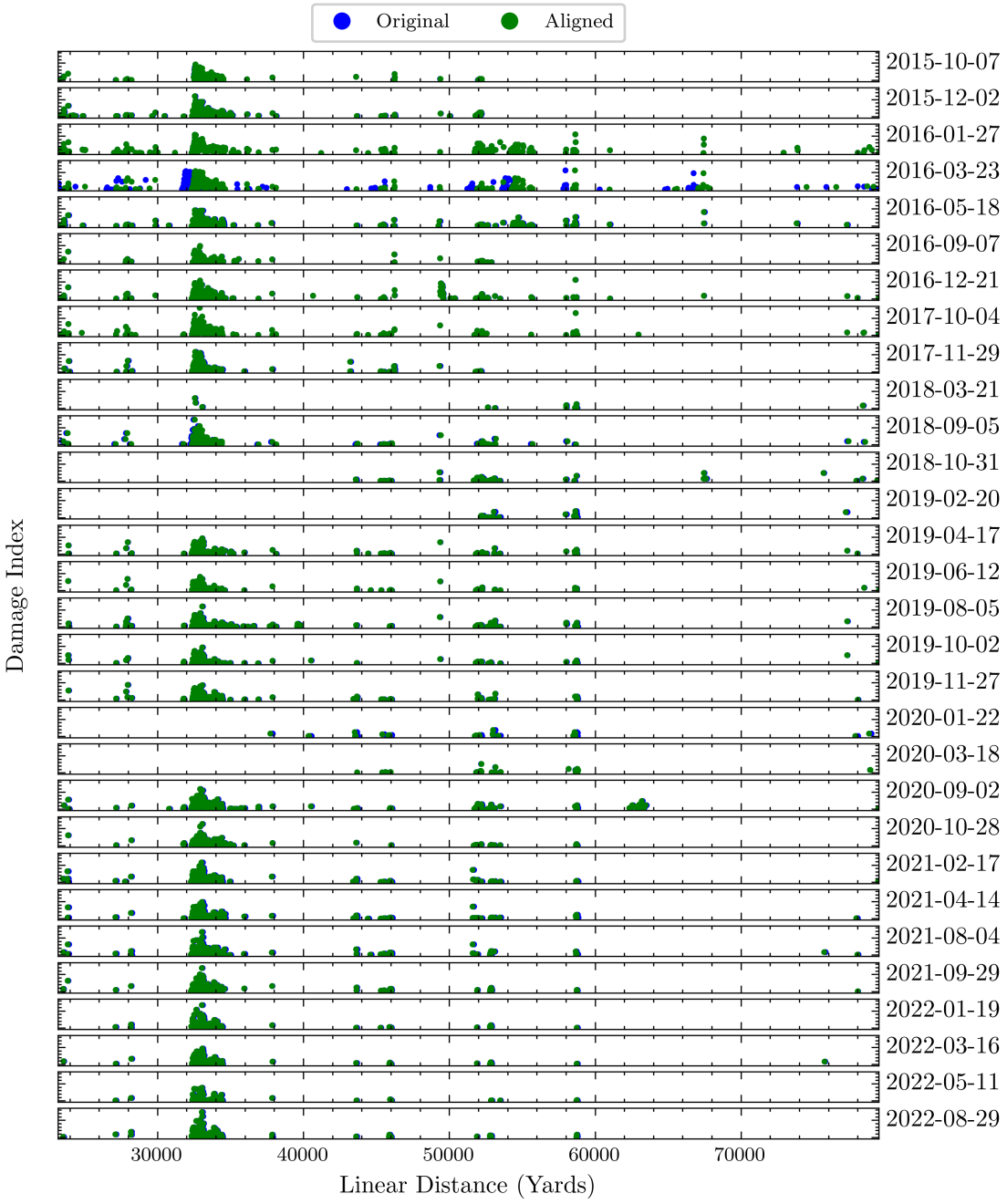


FIGURE 5.5: Optimal Alignment: Right Rail, Cluster i = 0, Max. Shift $\Delta* = 100$, Ref Signal $(\beta*) = 2$

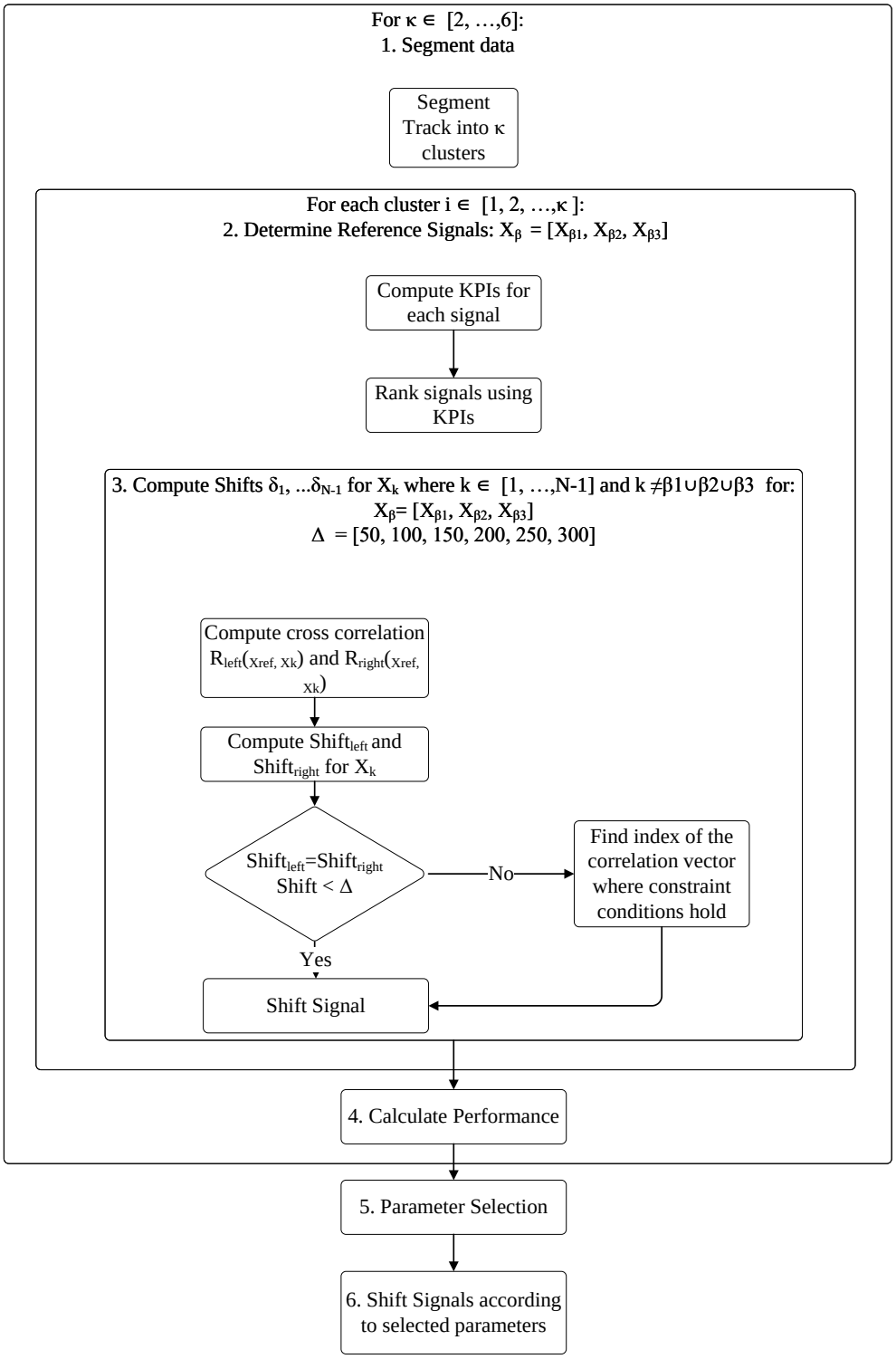


FIGURE 5.6: Spatial Alignment Procedure

TABLE 5.3: Performance Indicators: Optimal no. Clusters

Total Clusters κ	$\sum_{i=1}^{\kappa} \rho$	$\sum_{i=1}^{\kappa} \rho / \kappa$
1	0.29	0.29
2	1.02	0.51
3	1.82	0.61
4	2.00	0.50
5	2.59	0.52
6	2.75	0.46

5.1.2 Outlier Detection

Outlier detection refers to the problem of finding patterns in data that do not conform to expected normal behaviour (Chandola, 2009). It is important in this case to distinguish between *noise removal*, which aims to immunize a statistical model estimation against outlying observations, from *novelty detection*, which aims at detecting emergent patterns in the data. For example, errors due to measurement, processing and geolocation may be classed as noise, and on the other hand, observations that are subjected to an intervention, or very severe cracking events are considered as outliers in the traditional definition, but they are also interesting patterns that have significance in isolation as compared with other sources of noise. In this instance, we wish to identify noise which will otherwise influence subsequent statistical analyses.

5.1.2.1 Theory

Many outlier detection methods work on the principle that being an outlier is a binary property, however approaches such as the Local Outlier Factor assign to each observation a degree of being an outlier, which increases the information available for each observation, rather than discarding it (Breunig, 2000). The Local Outlier Factor (LOF) is a distance-based approach in which the density of regions in the data are computed, and the instances in the low density regions are declared as outliers. Data points with high LOF have more sparse neighbourhoods and typically represent stronger outliers, unlike data points belonging to dense clusters that usually tend to have lower LOF values (Lazarevic, 2005).

The approach requires a single parameter, k (or the *minimum points*) which indicate the number of nearest neighbours used in defining the local neighbourhood of the observation. For each observation x, the calculation of the Local Outlier Factor (LOF) is as follows (Sugiyama, 2016):

- **Calculate k-distances:** The distances between points using a distance function such as Euclidean or Manhattan are determined to find the k-nearest neighbours of point x,
- **Calculate the Reachability Distance:** The maximum distance between two points (x, x') and the k-distance of that point.

$$RD_k(x, x') = max(x - x^{(k)}, x - x') \tag{5.6}$$

where $x_{(k)}$ is the kth nearest neighbour of x in x_{i-1}^{n},

- **Calculate the Local Reachability Density:** For each point, determine the LRD which is a measure of the density of k-nearest points around a point x which is calculated by taking the inverse sum of all of the reachability distances of all the k-nearest neighbour points. When x is isolated from surrounding samples, the local RD takes a small value,

$$LRD_k(x) = \frac{1}{(\frac{1}{k}\sum_{i=1}^{k} RD_k(x^{(i)}, x))} \tag{5.7}$$

- **Calculate the LOF:** The ratio of the average of the LRDs of k number of neighbours of x and the LRD of x:

$$LOF_k(x) = \frac{\frac{1}{k}\sum_{i=1}^{k} LRD_k(x^{(i)})}{LRD_k(x)} \tag{5.8}$$

The intuition behind the calculation of the LOF is that if the density of the neighbours and x are almost equal, then these points are similar. If the density of the neighbours is much lower than the density of x, then x is an inlier, i.e. insider the cluster. Whereas, if the density of the neighbours is much higher than x, then x can be considered an outlier. Overall, if $LOF_k(x)$ takes a large value, then x is regarded as an outlier and it is at the discretion of the analyst to determine a LOF threshold if outliers are to be removed. Typically, a contamination parameter c is defined, which indicates the estimated proportion of outliers present in the dataset.

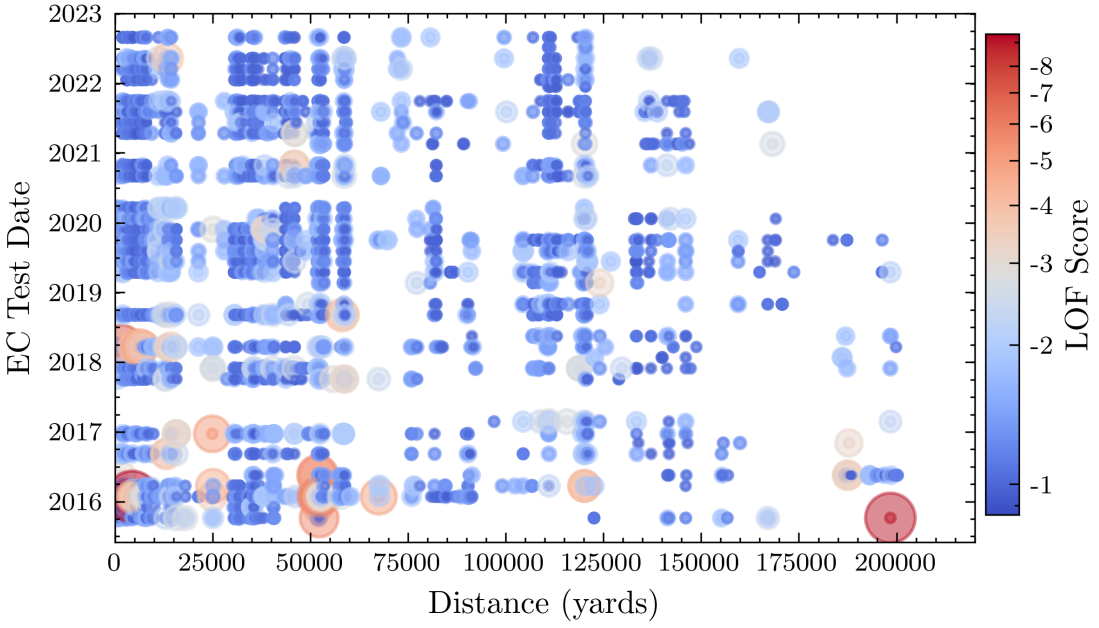
5.1.2.2 Application

To reduce the impact on subsequent analyses, we aim to identify the most significant local spatio-temporal outliers. These outliers are removed from the data set before performing regression analyses. The process implemented is described below:

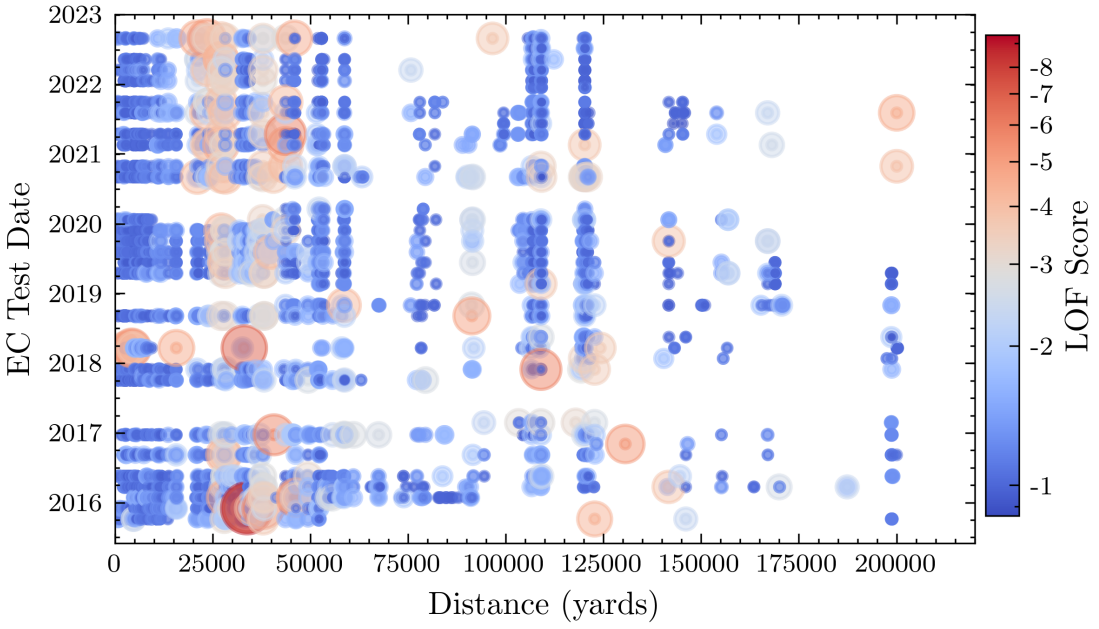
1. **Select Input Data:** The input data set is the aligned output from the previous processing step 5.1.1,

2. **Select Model Parameters:**
    - **No. Neighbours/ Min Points (k):** As suggested in the practical guidelines, a k value between 10 and 20 is suitable for many applications (Breunig, 2000). For this study, k = 10 is selected due to the sparsity of the data, the higher the k value the more local points are required for an observation to be considered an inlier, and therefore the lower bound of this recommendation was selected,
    - **Contamination:** a contamination of 3% is utilised to remove the most severe outliers,
    - **Features:** Linear Distance, Measurement Date, Rail Side, Damage Indicator ($DI2$) are selected,
3. Calculate the LOF for each observation,
4. Remove 3% of data with highest LOF.

From a dataset of 17,184 observations, 52 points are removed using this method. Figure 5.7a and 5.7b indicate the spatio-temporal locations of the aligned observations, with the colour and size of the point representing the LOF Score (Note that the colour scale is log normalised in order to show more clearly the anomalous points). There is not an obvious pattern in the locations of these outliers, but there is a degree of clustering of moderate outliers for the left rail at approximately 125,000 yards. However, many of the more significant outliers tend to occur before 2019, which may be a result of how measurement procedures have improved since then. Further, as illustrated in Figure 5.8 many of the observations with high outlier scores have lower damage scores, and all of the points removed from the data set correspond to damage values: $DI2 < 50$. It is therefore, unlikely that we are removing any potentially significant observations of 'unexpected hotspots'.



(A) Left Rail



(B) Right Rail

FIGURE 5.7: Spatio-temporal Representation of Outliers Identified using the LOF, n = 10

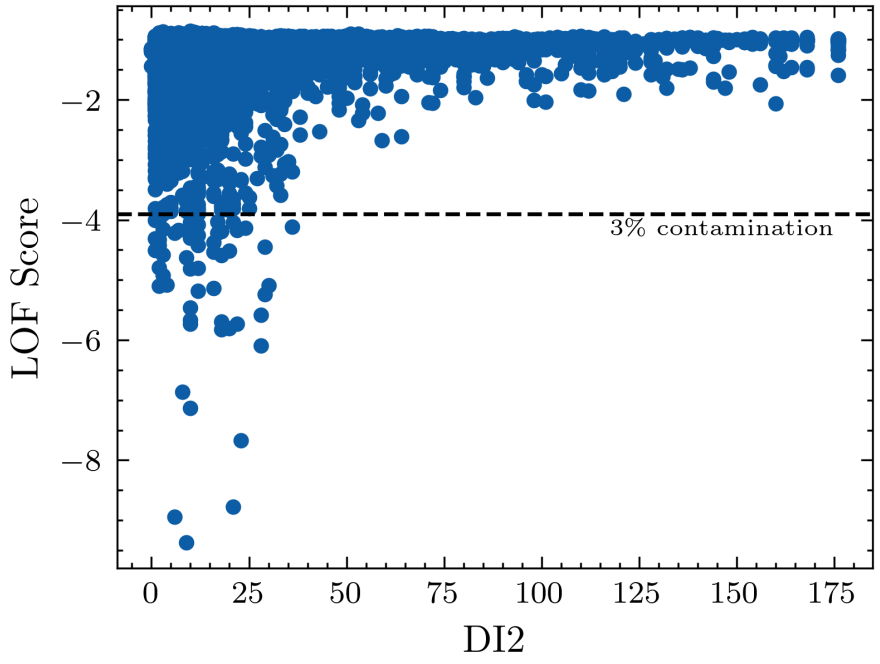


FIGURE 5.8: Illustration of LOF variation with Damage Score, n = 10

5.1.3 EC Spatial Smoothing

Noise reduction can be performed using methods such as filtering, smoothing and prediction (estimation), but the choice of method is dependent on the application. In the case of the EC data, we wish to smooth the EC cross sectional data in the spatial domain, to reduce localised noise particularly with the aim of calculating rates of change between measurements.

Smoothing improves data quality by replacing the noisy irregular signal with the new smoothed signal which may better describe the measured phenomena. Methods include mean filters, median filters, Kalman filters and Gaussian filtering. Due to computational efficiency, and applicability to the target data- spatial data where ordering must be preserved, a mean filter is applied here.

The mean filter is one of the simplest smoothing methods (Kowalski, 2018), and the output value is calculated by computing the average of all samples from a given window. Its' formula has the following form:

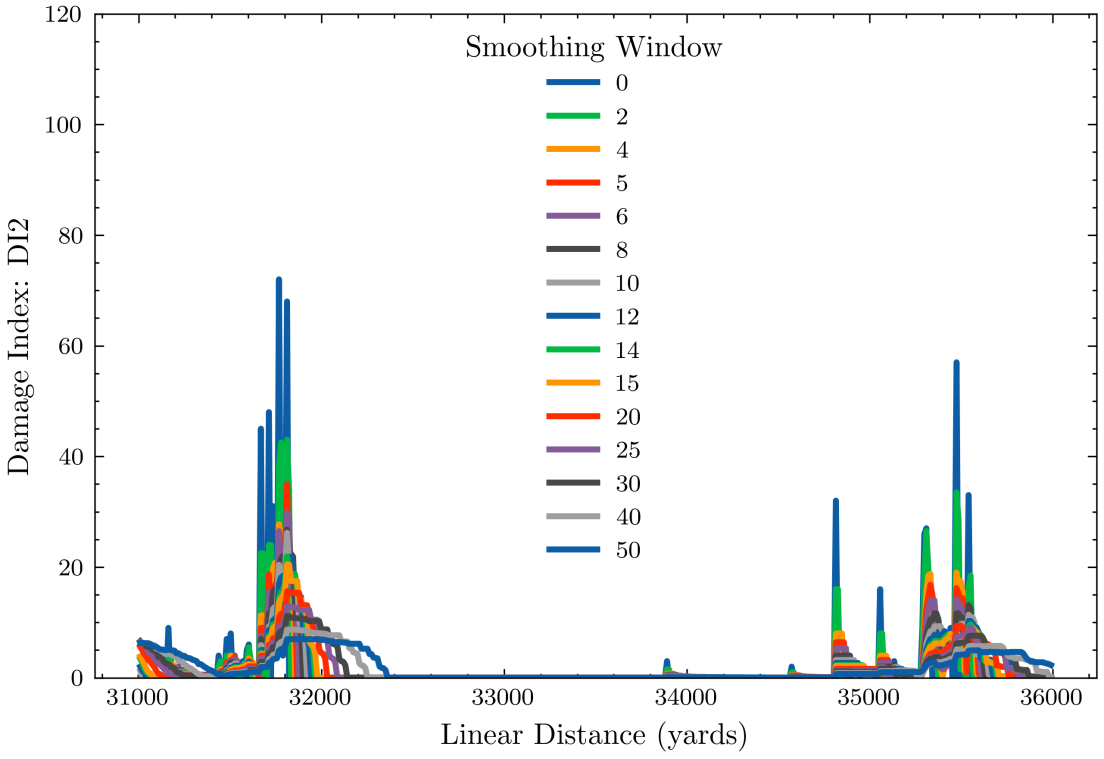
$$z_j = \frac{\sum_{i=-n}^{n} x_{j+i}}{2n+1} \tag{5.9}$$

Where x_i is a sample of the input signal, $2n+1$ is the window length, z_j represents the output value, and j is the current index of the output value.

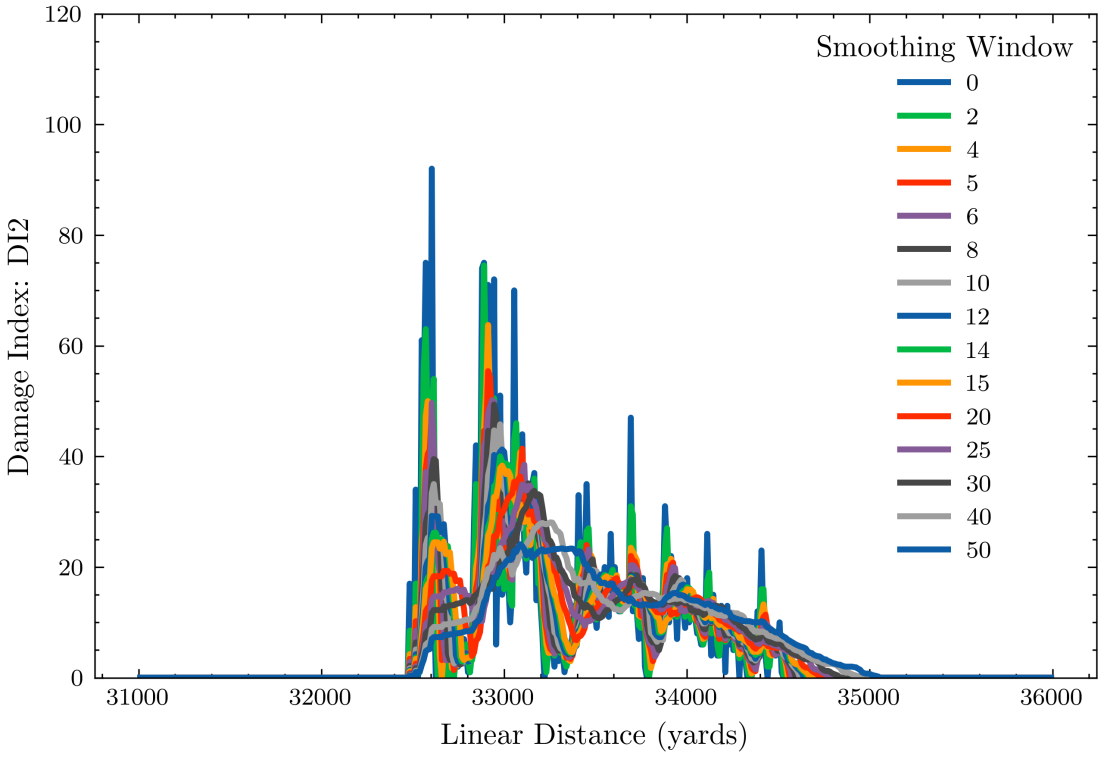
5.1.3.1 Application

The mean filter is applied to each spatial cross-section of data $X_1, X_2, ...X_N$ for the left and right rail independently, utilising the window sizes, $w = [5, 10, 15, 20, 25, 50]$[1]. The window sizes are arbitrarily chosen to provide a reasonable range of windows without reducing the magnitude of signal peaks too considerably. Note that a window size of 5 applied to the 11 yard data means that damage is averaged over 55 yard intervals across the data, whereas a window size of 10 applied to 11 yard data corresponds to averaging over a 110 yard interval. Figure 5.9 illustrates the effect of spatial smoothing on the 11 Yard damage signal for recordings collected on a particular date. The effects of peak truncation are apparent as the smoother window is widened especially where the window exceeds 15 segments and the 11 yard damage is concentrated over a small distance. Additionally Table 5.4 shows the effects of the smoothing window size on key signal parameters such as the minimum and maximum damage, the damage variance and mean damage. Each of the parameter values reduce sharply to begin with, however as the smoothing window increases the reduction in value levels off. When developing a regression model, a suitable smoothing parameter will be selected in Chapter 7. The appropriate degree of smoothing should balance the effects of model accuracy, correct data representation and model computation time.

[1] Although these are the windows illustrated here, for the final analysis carried out in Chapter 7, additionally windows $w = [2, 4, 6, 8, 10, 12, 14]$ were tested to increase the resolution for smaller window sizes



(A) Left Rail



(B) Right Rail

FIGURE 5.9: Spatially Smoothed Damage Data

TABLE 5.4: Data Characteristics after Smoothing

Rail Side	Window	Min. Damage	Max Damage	Variance	Mean
Left	0	0	176	389.07	14.21
Left	2	0	176	231.97	10.35
Left	4	0	164	142.8	7.38
Left	5	0	166.4	121.95	6.58
Left	6	0	166	106.84	5.97
Left	8	0	150	85.89	5.1
Left	10	0.1	134.6	71.61	4.5
Left	12	0.08	128.58	61.24	4.05
Left	14	0.07	123.07	53.3	3.7
Left	15	0.07	117.93	50.01	3.56
Left	20	0.05	104.25	37.85	3.01
Left	25	0.04	91	29.99	2.65
Left	30	0.03	76.03	24.61	2.39
Left	40	0.025	69	17.94	2.04
Left	50	0.02	69	14.01	1.82
Right	0	0	176	607.15	18.96
Right	2	0	176	394.2	14.84
Right	4	0	168.5	263.59	11.46
Right	5	0	161.6	228.99	10.46
Right	6	0	156.67	202.92	9.67
Right	8	0	150.5	164.78	8.48
Right	10	0	125.6	138.62	7.63
Right	12	0	108.5	119.85	6.97
Right	14	0	95.07	105.98	6.45
Right	15	0	89.93	100.37	6.23
Right	20	0.05	70.3	80.07	5.38
Right	25	0.04	73.04	66.89	4.79
Right	30	0.03	63.83	57.55	4.36
Right	40	0.025	59	45.14	3.75
Right	50	0.02	59	37.44	3.33

5.1.4 EC Feature Engineering

Once the aligned and smoothed data sets are generated, additional features are calculated to improve resultant analyses and regression modelling. These features are a result of iterative analyses and prior subject knowledge.

As previously described, the eddy current data are spatio-temporal, presenting characteristics such as autocorrelation and heterogeneity amongst others (see Section 4.5). Spatio-Temporal Data Mining (STDM) is a particular branch of data analytics which explicitly models this type of data. In the literature, the benefit of using STDM is to capitalise on novel formulations and methods that arise from these data due to their dimensions (Atluri et al., 2017). However, predominantly these studies assume regular gridded data, such that observations are available for each

(s, t) cell in the dataset. For the EC data presented here, this is clearly not the case and without the use of estimation or interpolation techniques, these approaches cannot be taken advantage of. An alternative to using ST models whilst still accounting for the natural ordering, and relationships within this type of data is to construct a feature set that represents the ST variation in the data.

Further, due to rail maintenance, either through grinding, milling or re-railing, the data does not represent a process of pure degradation, since there are restorative processes occurring. As described in Chapter 4 obtaining reliable and sufficiently detailed interventions data is a problem since we cannot determine accurately where the track segment is degrading naturally or there has been some other process that has modified the rate of degradation. It is therefore necessary to determine when these interventions have occurred, and generate new temporal features describing a degenerative process only, without the inclusion of data which has experienced improvements.

- **Track Segment Damage Rates** Rates of deterioration are useful in identifying hot spots, and also may provide increased insight. Rates of change can be calculated for each track segment s between two subsequent inspections [t, $t+1$]. For instance, take observation $DI_{s,t-1}$ and $DI_{s,t}$, where s is the track segment, [$t-1$, t] is the measurement interval, then the rate of change is calculated:

$$RateDI_{s,t} = \frac{DI_{s,t} - DI_{s,t-1}}{(T_{s,t}) - (T_{s,t-1})} \tag{5.10}$$

- **Lagged Damage Variables**: To represent the autocorrelation in the data, it is assumed that rail damage in a segment of track is dependent on previous measurements of rail damage. These Lagged Damage Variables are continuous features which represent the Damage Index: $DI2$ from the previous inspection for the spatial cell (track segment),

- **Time Since Intervention** The EC RCF cracking data does not represent a system of solely rail degradation due to RCF, but is influenced by the presence of maintenance activities. These improvements in rail condition (a reduction in the damage indicator : $DI2$), are visible from the data set (See Figures 4.13 and 4.14 from Chapter 4),

  In order to model the degradation process, in this study it is assumed that an improvement in the rail condition indicates an intervention and therefore should not be represented in the dataset. For each rail and track segment, the rate of change of $DI2$ between subsequent measurements is calculated, the *measurement time*: T_{ec} is then converted to an *Intervention Time* T_{int}. Clearly, if an intervention is detected over an interval: $[t, t+1]$ the precise T_{int} is unknown. However, in

the absence of reliable interventions data it is assumed in this study that; in this case, the T_{int}= 0 days, and subsequent measurements (where degradation occurs) are re-indexed accordingly with reference to this measurement. All records in the data set represent a system of degradation, note that where there are less than 2 measurements on a track segment, rates cannot be obtained and therefore these records are removed.

- **High/ Low Rail Indicator:** The high rail, due to super-elevation in curves, is hypothesised in many studies to have higher rates of RCF due to increased contact forces experiences at the leading outer wheel-set when curving. An indicator for whether the rail is the low or high rail can be determined using curvature and rail side information (from the EC data: it is assumed that the left/ right indicator is consistent for all EC inspections over a single length of track).
  - If $RailSide = Left$ and $Curvature \leq 0 \Rightarrow High/Low = Low$
  - If $RailSide = Left$ and $Curvature \geq 0 \Rightarrow High/Low = High$
  - If $RailSide = Right$ and $Curvature \leq 0 \Rightarrow High/Low = High$
  - If $RailSide = Right$ and $Curvature \geq 0 \Rightarrow High/Low = Low$
  - If $Curvature \geq 0$ and $Curvature \leq abs(1e-4) \Rightarrow High/Low = Tangent$, 1e-4 is currently set as an arbitrary tangent/ curve boundary
  - If $Curvature = 0 \Rightarrow High/Low = Unknown$, this category represents 0 curvature, it has previously been mentioned in conversations that curvature is this category is most likely non-zero (Personal Communication, Julian Williams, Network Rail- Principal Analyst Whole Life Cycle Costing, 2022).

5.2 Additional Data Sources Processing

The points below describe the processing stages for each of the supporting data sources utilised in the study, each data source must be cleaned and discretised as a first step, followed by generation of additional required features, and then integrated with the Eddy Current data. The approach for data integration depends on the spatio-temporal nature of the data source. For data sources with space and time attributes, the features must be mapped to each EC observation, whereas for spatial data, such as track characteristics and segment wear rates, these features can be mapped to the EC spatial cell. Table 5.5 contains a final set of the parameters included in the study, and an indication of the data type.

The resultant data is a modelling dataset $P_{pro,w}$ for varying smoothing window size w.

- **Wear Data**

1. **Data Extraction:** The wear data is extracted directly from the NR network drive containing *csv* files for each recording run. The wear parameters of interest are: Vertical Wear (Max. Average), Gauge Face Wear (Max., Average), Field Wear (Max., Average), Head Width Remaining (Max., Average), Gauge Face Remaining (Max., Average) and the Estimated Rail Depth (Minimum),
2. **Basic Processing:** cleaning such as removal of duplicate entries is required for the wear data, note that where a duplicate is detected the most recent entry is retained in the data set,
3. **Discretisation and Aggregation:** the wear data are mapped to the 11 yard track segments, where an 11 yard track segment overlaps with two wear segments the average of the wear values in both wear spatial cells are used,
4. **Calculate Deterioration Rates:** the average deterioration rates are then calculated for each of the wear parameters,

- **ACTRAFF Data**

1. **Data Extraction:** The ACTRAFF parameters selected are the Period ID, Start Yards, End Yards, Total Tonnes, Passenger Tonnes, Freight Tonnes, Total No. Trains, No. Passenger Trains and No. Freight Trains,
2. **Basic Processing:** Missing periods are interpolated using an average of the periods either side of the missing period,
3. **Estimation:** As previously stated, the ACTRAFF data is not available from approximately 2019, however this varies depending on track location. The future traffic must therefore be extrapolated in order to leverage ACTRAFF data in the study. A basic estimate is employed for determining Passenger, Freight and Total Traffic. These estimates are based on discussions with Network Rail staff, Network Rail published reports (as stated) and assumptions that traffic from the years pre-pandemic were roughly constant between corresponding periods from year to year. We know that following the pandemic (as stated in Chapter 1) that there was a considerable reduction in traffic on the network. The estimates used are as follows:
   - Where the estimated date is pre-COVID, the period estimate from the previous year is utilised, i.e. 2018-2019 year,
   - Where the estimated date is from April 2020 - April 2021, we employ a 75% reduction in traffic for each period from pre-pandemic levels (78% reported in (Department for Transport, 2021)),
   - For April 2021-April 2022, a 50% reduction is used (41.8% reported in (Network Rail, 2022)),

- For April 2022- April 2023, a 25% reduction is used based on the assumption that uptake improves again by approximately 25%.

An illustration of these estimates is provided in Figure 5.10 for TL1.

4. **Discretisation and Aggregation:**, the ACTRAFF data are discretised into 11 yard segments, where an 11 yard segment overlaps two ACTRAFF segments the traffic data closest to the start yardage of the 11 yard segment are used (i.e. the data to the left)

- **Defect Data:** A number of studies have concluded that the existence of rail surface defects intensifies the deterioration of tracks, and has the potential to result in rail failure due to increased vertical dynamic forces of wheelsets onto rails (Zhang et al., 2022). Therefore, as a key risk factor, the presence of other defects detected using ultrasonics/ visual monitoring practices is incorporated into the feature set. There are two key features generated for each observation, 1) the total number of defects detected on the section s over the course of monitoring 2) the existence of a defect on the section s since the last measurement date $t-1$ (binary).

1. **Data Extraction:** Historic defect data are extracted from the RDMS,
2. **Basic Processing:**
3. **Data Discretisation and Aggregation:** The total number of defects are summed over each spatial cell,

- **Track Summary Data:**

1. **Data Extraction:** the primary parameters of interest include track cant, curvature, maximum line speed, tonnage, track category, rail material, track type, location of S&C, stations and tunnels,
2. **Basic Processing:** Conversion of categorical data types,
3. **Feature Generation:**
   - **Centripetal Force** $F_{centripetal}$,

$$F_{centripetal} = mv^2 \tag{5.11}$$

   where m is the tonnage and v is the maximum line speed.
   - **Cant Deficiency** C_D (in mm),

$$C_D = \frac{11.82 * v^2}{R} - C_A \tag{5.12}$$

   where V is the maximum line speed, R is the curve radius and C_A is the cant applied, the 11.82 is defined for normal gauge and takes into account g (gravitational acceleration), S (the cross-level standardised

reference for rail heads centreline distance (mm)) and the speed unit conversion from m/s to km/h, see (Constantin, 2015; Cope, 1993) for a full description,

- **Distance to Features:** traction and braking forces are related to the generation of frictional (creep) forces responsible for RCF, it is therefore hypothesised that where trains are likely to be braking or accelerating more frequently will experience more instances of RCF. The distance of each cell from the nearest Stations, Tunnels, Switches and Crossings (SC) is therefore calculated,
- **Time since Rail Replacement:** the track summary data includes the INM values for when each segment of rail was replaced. Since the data is only available at a yearly resolution, the replacement is assumed to have occurred on the 1st January of that year,

4. **Discretisation and Aggregation:** the track summary data are discretised and mapped to the 11 yard track segments, where the 11 yard segment overlaps with 2 TS segments:
   - for the continuous value parameters (such as curvature, speed, cant) the average value is used,
   - for the feature distances- the minimum values are used, i.e we assume the worst case,
   - for discrete binary parameters such as the location of tunnels and stations one segment will indicate 0- no station, and the other 1- a station present, the aggregated value always assumes 1,
   - for discrete parameters such as the age of the rail, the value which covers the greatest proportion of the cell is used.

TABLE 5.5: Features

Feature	Type	
Track Characteristics		
Curvature (1/m)	Continuous	Spatial
Cant (mm)	Continuous	Spatial
Cant Deficiency (mm)	Continuous	Spatial
Line Speed (mph)	Continuous	Spatial
EMGTPA (MGT)	Continuous	Spatial
Station Indicator	Binary	Spatial
Linear Distance to Station (Yards)	Continuous	Spatial
S&C Indicator	Binary	Spatial
Linear Distance to S&C (Yards)	Continuous	Spatial
Tunnel Indicator	Binary	Spatial
Linear Distance to Tunnel (Yards)	Continuous	Spatial
Track Category	Discrete	Spatial
Wear Data		
Vertical Wear Average (mm)	Continuous	ST
Vertical Wear Max. (mm)	Continuous	ST
Gauge Face (GF) Wear Average (mm)	Continuous	ST
Gauge Face (GF) Wear Max. (mm)	Continuous	ST
Field Wear (FW) Average (mm)	Continuous	ST
Field Wear (FW) Max (mm)	Continuous	ST
Head Width Remaining Avg. (mm)	Continuous	ST
Head Width Remaining Min. (mm)	Continuous	ST
GF Remaining Avg. (mm)	Continuous	ST
GF Remaining Min. (mm)	Continuous	ST
Estimated Rail Depth Min. (mm)	Continuous	ST
ACTRAFF		
Total Tonnes (MGT)	Continuous	ST
Passenger Tonnes (MGT)	Continuous	ST
Freight Tonnes (MGT)	Continuous	ST
Total Trains (MGT)	Continuous	ST
Passenger Trains (MGT)	Continuous	ST
Freight Trains (MGT)	Continuous	ST
Defect Data		
Total Defects	Discrete	ST

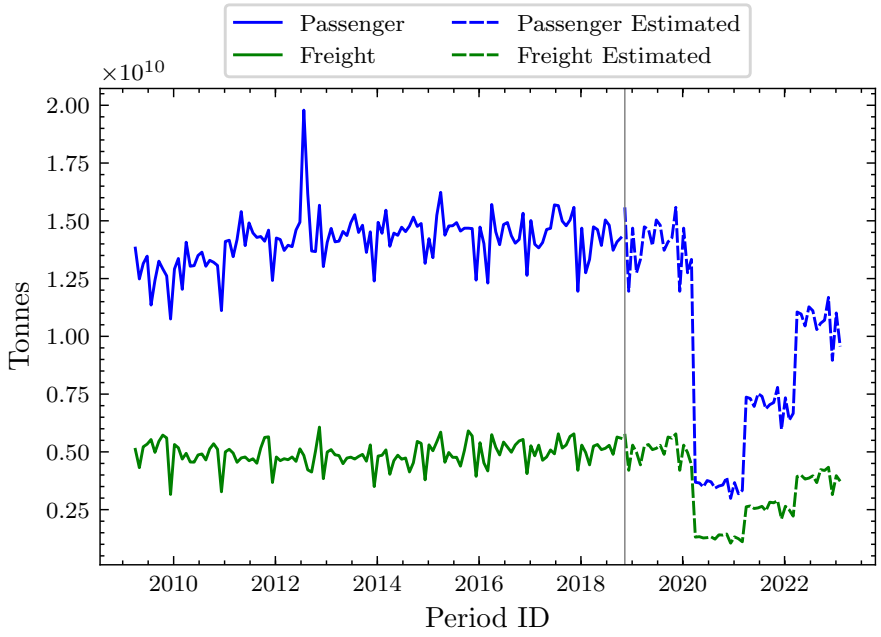


FIGURE 5.10: ACTRAFF Projected Passenger and Freight Tonnes over TL1

5.3 Concluding Remarks

This chapter presents the essential pre-processing tasks that are required to ensure the data is suitable for further analysis, and additionally is formulated such that it can be used as training data for a machine learning algorithm. The Eddy Current data is extracted directly from the RDMS which represents the maximum crack depth measured (mm) over a 1 yard section of the network at a particular measurement date. This data are mapped to the 11 yard network whereby a damage index is constructed which represents the density and severity of RCF cracking over an 11 yard section. As shown in Chapter 4, the data are often misaligned in space between recording runs, and therefore a novel process for re-aligning Eddy Current signals is developed using cross correlation methods. Further, to address noise in the EC data, outliers are removed using a density based parameter known as the Local Outlier Factor (LOF) and smoothing is applied to reduce the effects of localised noise in the spatial domain. The spatial smoothing parameter is varied such that the effect of modifying this window can be assessed when developing a ML model for RCF prediction (see Chapter 7). As the smoothing window is increased, the signal converges to the global average and is thus easier to predict, however this would not be a very useful model and therefore we must determine a balance between model accuracy and correct data representation. Subsequently, additional features such as Lagged Damage Indicators; which reflect the known autocorrelation in the data, rate variables and parameters reflecting the estimated time since an intervention are generated. Finally, the processed Eddy Current data are integrated with supporting data sets such as track

characteristics, wear, defect and cumulative traffic parameters to generate a final modelling data set with representative features for mining information.

The benefits of processing the data are two-fold, firstly without too much complex analysis the processed data set could now be used by maintenance teams to conduct practical analysis of the Eddy Current Data. The focus of Chapter 6 is specifically aimed at using the processed data to augment the current strategies for preventive and corrective RCF intervention measures introduced in Chapter 2. Secondly, the comprehensive cleaning and preparation of the data ensures the best possible outcome with regard to building representative data-driven models for RCF prediction. Without a thorough pre-processing methodology, it is unlikely that any resulting models would yield results that were accurate or reliable. The RCF regression modelling work is presented in Chapter 7.

Chapter 6

Practical Applications of Data Analysis

Chapter 5 provides a summary of the techniques applied to the raw Eddy Current and supporting data sources to generate a modelling data set which is suitable for data mining applications. This chapter uses that data and expands on the preliminary analyses provided in Chapter 4 to demonstrate simple methods which capitalise on the processed data ($P_{pro,w}$) and could be used to supplement existing maintenance strategies. The section is structured as follows:

- **Rate Analyses:** an introduction to damage rates of change is presented, including terminology and parameters which form the foundations for application within preventive and corrective maintenance strategies,
- **Preventive Applications:** describes an approach for the use of pre-processed EC data in determining a revised method for segmenting the track into categories with similar damage properties,
- **Corrective Applications:** utilises the pre-processed EC data to identify RCF hot spots and rank these according to custom Key Performance Indicators (KPIs).

Note that all of the following analysis is performed using the processed data set $P_{pro,0}$ (i.e. no smoothing is applied).

6.1 Rate Analyses

After cracking events are mapped and aggregated over 11 yard segments, as indicated in Section 5.1.4, the rate of change for each spatial segment can be calculated. For each

track segment s, rates of change of $DI2$, VS, S, M, H, L can be calculated between two subsequent inspections $[t, t+1]$. For example, take two observations $DI_{s,t-1}$ and $DI_{s,t}$, where s is the 11 yard segment, $[t-1, t]$ is the measurement interval, and DI represents the measured damage (either $DI2$, VS, S, M, H, L). Then the rate of change can be calculated:

$$R_{s,t} = \frac{DI_{s,t} - DI_{s,t-1}}{(T_{s,t}) - (T_{s,t-1})} \tag{6.1}$$

Note that where $R_{s,t} > 0$ (positive rate), this represents degrading rail condition, whereas $R_{s,t} < 0$ (negative rate) indicates an improvement in rail condition. In this section, when discussing the *degradation* rate, we refer only to rates which are greater than or equal to 0, i.e. $R_{s,t} \geq 0$. Figure 6.2a and 6.2a illustrate the monthly rate of change for damage index $DI2$ corresponding to the left and right rail, respectively. The positive rates are indicated by yellow-red tones, and negative rates are represented by green- blue tones. Comparatively the right rail appears to experience higher rates of degradation than the left rail, with more points exceeding rates of 60 units. Table 6.1 and Figure 6.1 illustrate the distribution of track curve radius for the TL1 track section. A negative curve radius represents a Left Hand (LH) curve, whereby the RHR would act as the high rail, and a positive curve radius (Right Hand (RH) curve) results in the LHR acting as the high rail. There are considerably more track segments where the RHR acts as the high rail and has a curve radii in the range 1000-2000 m (1281) compared with 705 for the LHR. This region is known to be where RCF is induced, and may be a factor explaining the prevalence of higher rates in the RHR.

TABLE 6.1: Distribution of Curve Radii for TL1

Curve Radius	**No. 11 Yard Segments**
$\leq$ -3000.0	529
(-3000.0, -2000.0]	513
(-2000.0, -1500.0]	494
(-1500.0, -1000.0]	787
(-1000.0, -500.0]	217
(-500.0, -0.01]	0
Tangent Track	9992
(0.01, 500.0]	46
(500.0, 1000.0]	89
(1000.0, 1500.0]	292
(1500.0, 2000.0]	413
(2000.0, 3000.0]	1130
$>$ 3000	1002

Additionally, Table 6.2 demonstrates the distribution of monthly degradation rates for all observations in the processed data set. The data are divided into six categories, with group size increasing with degradation rate due to the negative skew in the data

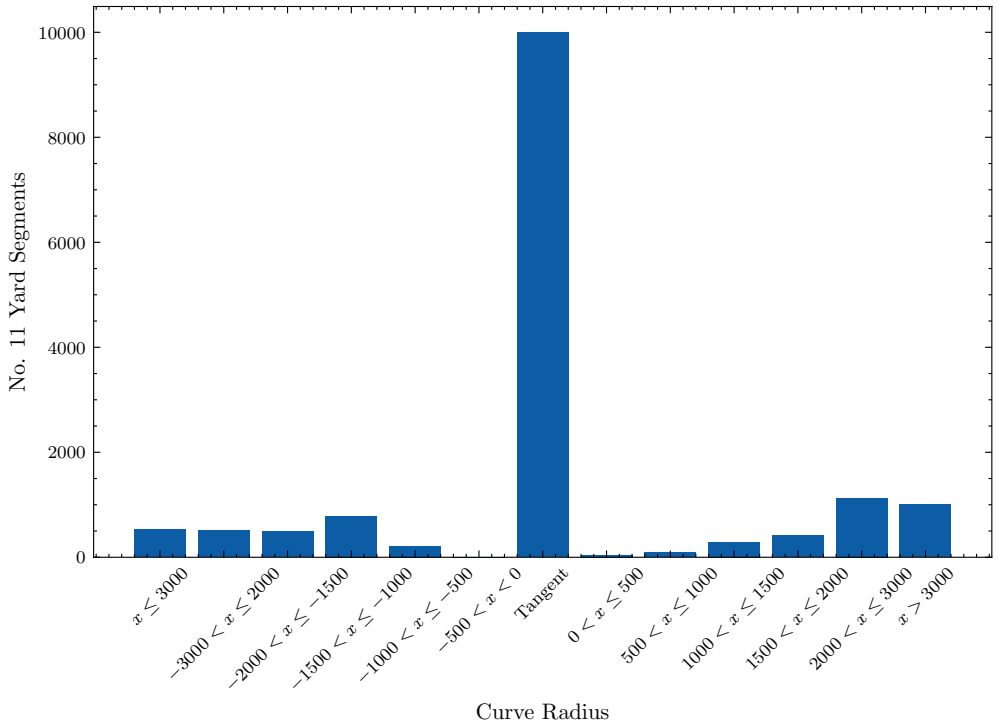
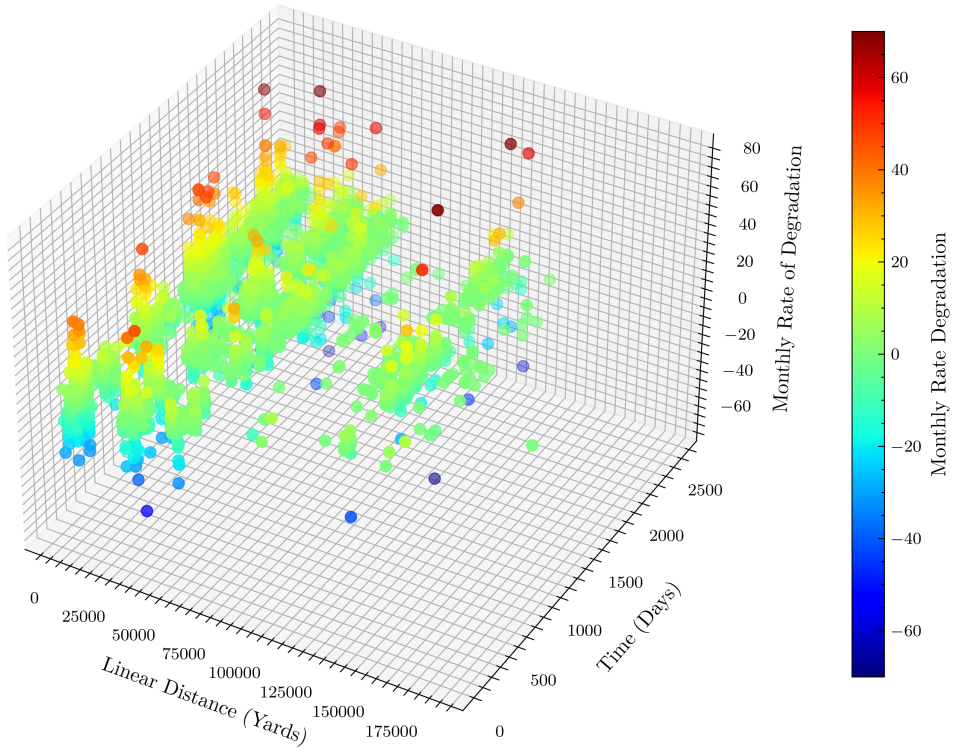


FIGURE 6.1: Curve Radius Summary for TL1

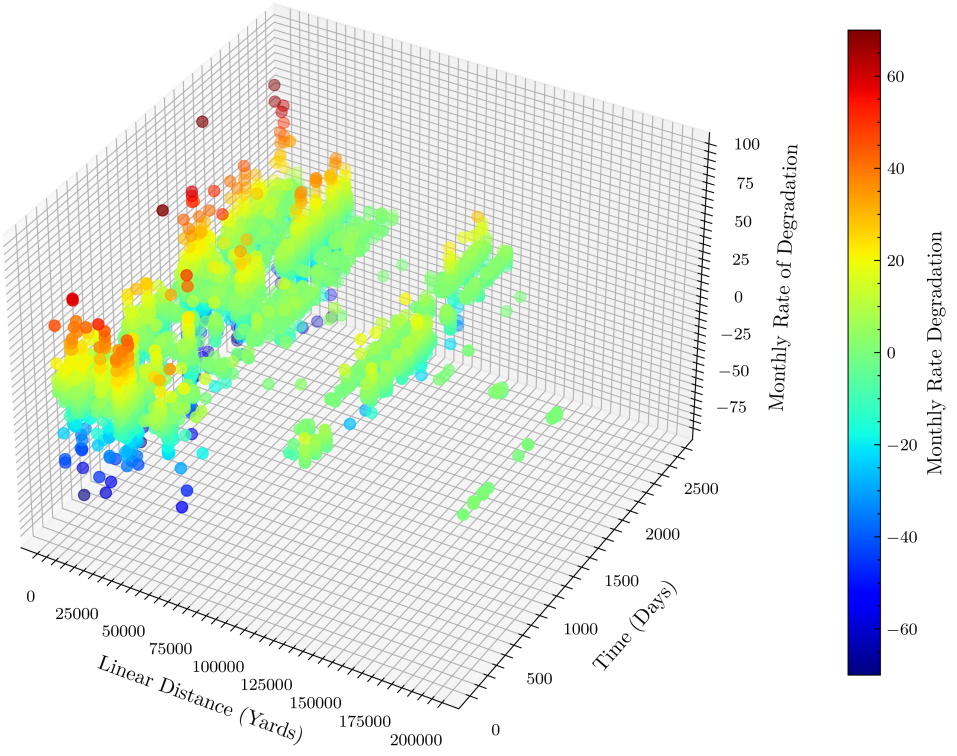
(note that Category 0 essentially represents a rate of 0). The table supports the observation that the right rail presents higher rates of degradations amongst the observation, with the quantity of observations exceeding the left rail in all categories. Figure 6.3a and 6.3b show a histogram of the information presented in Table 6.2.

TABLE 6.2: Monthly Degradation Rate Categories

Monthly Degradation Rate Category	**Category No.**	**Left Rail: Count**	**Right Rail: Count**	**Both Rails**
(0.0, 0.1]	0	151	94	245
(0.1, 2.5]	1	1740	2360	4100
(2.5, 5.0]	2	518	796	1314
(5.0, 10.0]	3	379	590	969
(10.0, 20.0]	4	213	382	595
(20.0, 100.0]	5	123	186	309

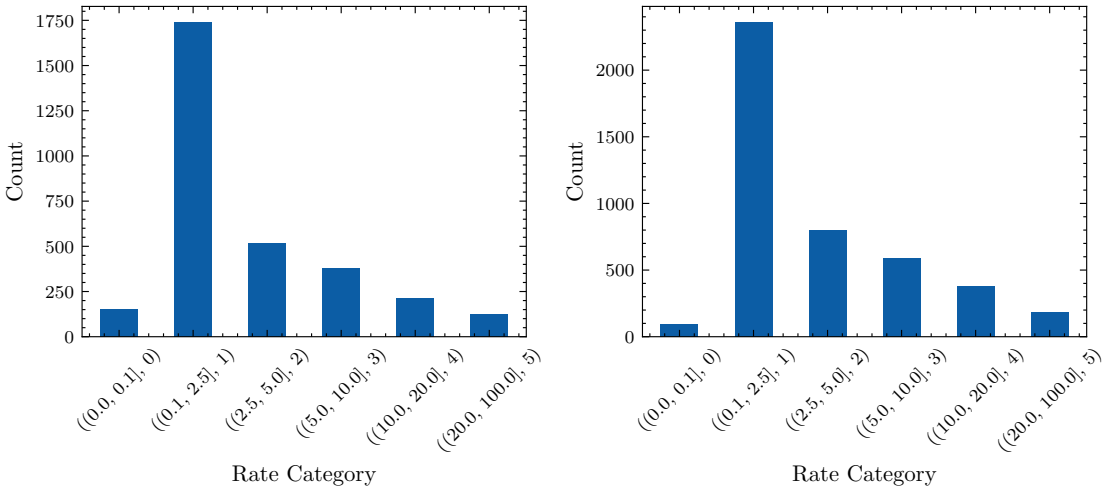


(A) Left Rail



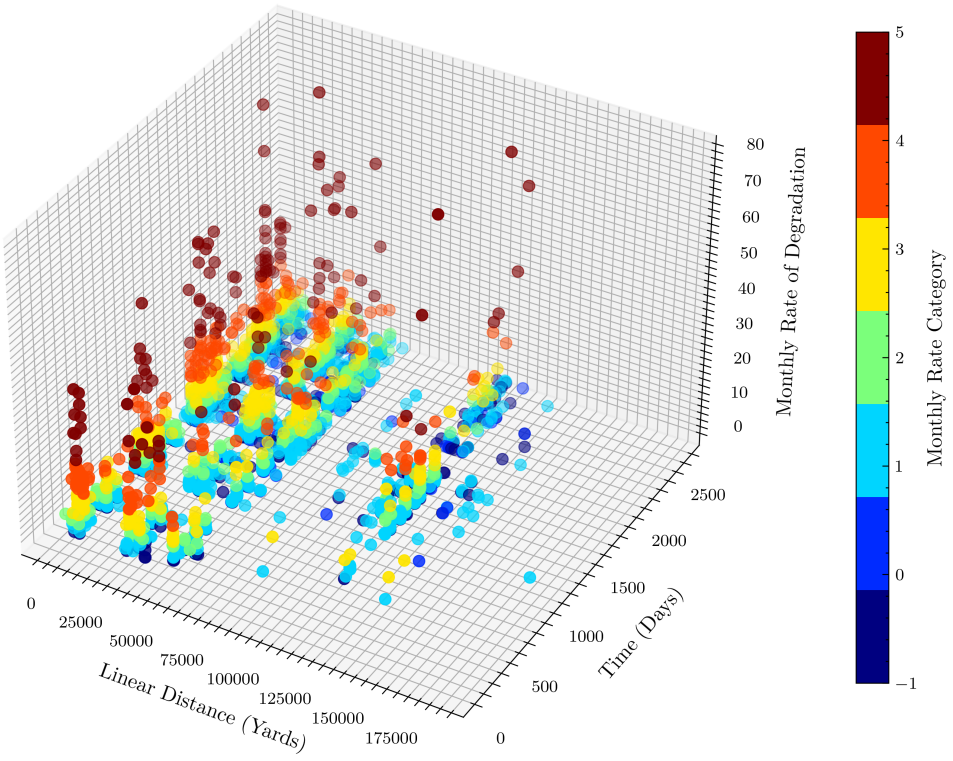
(B) Right Rail

FIGURE 6.2: Monthly Rates of Change of *D*I2

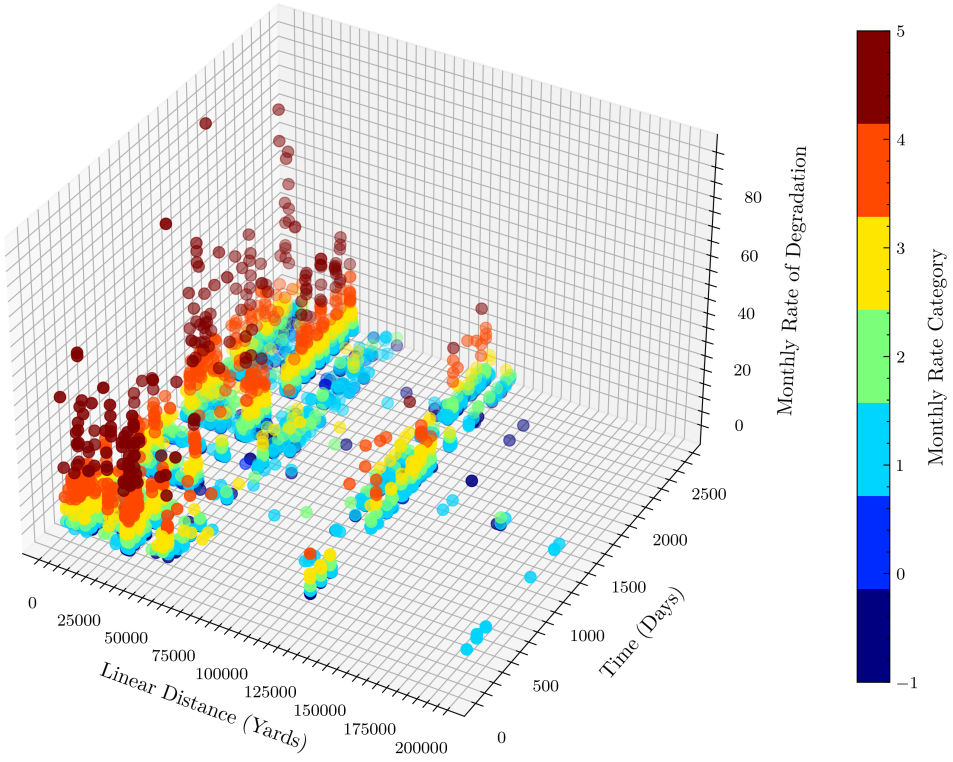


(A) Left Rail (B) Right Rail

FIGURE 6.3: Histogram for the Monthly *DI*2 Degradation



(A) Left Rail



(B) Right Rail

FIGURE 6.4: Monthly Rates of Change of *DI2*

6.1.1 Segment Rates

In much of the following work we refer to specific *KPIs* for each track segment s, which are defined:

- **Mean Segment Degradation Rate:** the mean of all positive degradation rates for segment s:

$$KPI_s R_{mean} = \frac{\sum_{i=1}^{a_s} R_i}{a_s} \tag{6.2}$$

  where a_s is the total number of observations in segment s where $D_{i+1} - D_i \geq 0$

- **Max Segment Degradation Rate:** the maximum of all positive degradation rates for segment s,

$$KPI_s R_{max} = Max(R_i) \tag{6.3}$$

  where $i = 1, ...a_s$ are the observations in segment s where the rate of change is positive or equal to zero,

- **Segment DI2 Sum:** the sum of damage over all observations in segment s:

$$KPI_s DI_{sum} = \sum_{i=1}^{b_s} D_i \tag{6.4}$$

  where $j = 1, ...b_s$ are indices for all observations in segment s, (i.e. where there are no constraints on $D_{i+1} - D_i$)

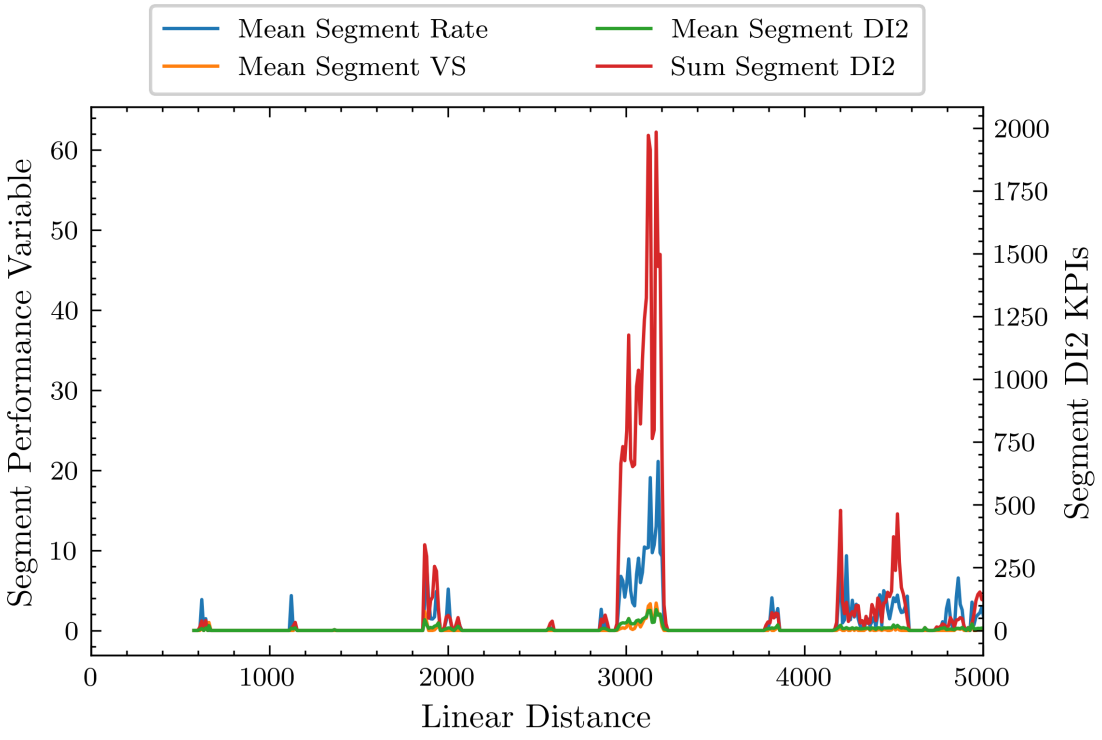
- **Mean Segment DI2 Sum:** the average of the sum of all observations in segment s:

$$KPI_s DI_{mean} = \frac{\sum_{i=1}^{b_s} D_i}{b_s} \tag{6.5}$$

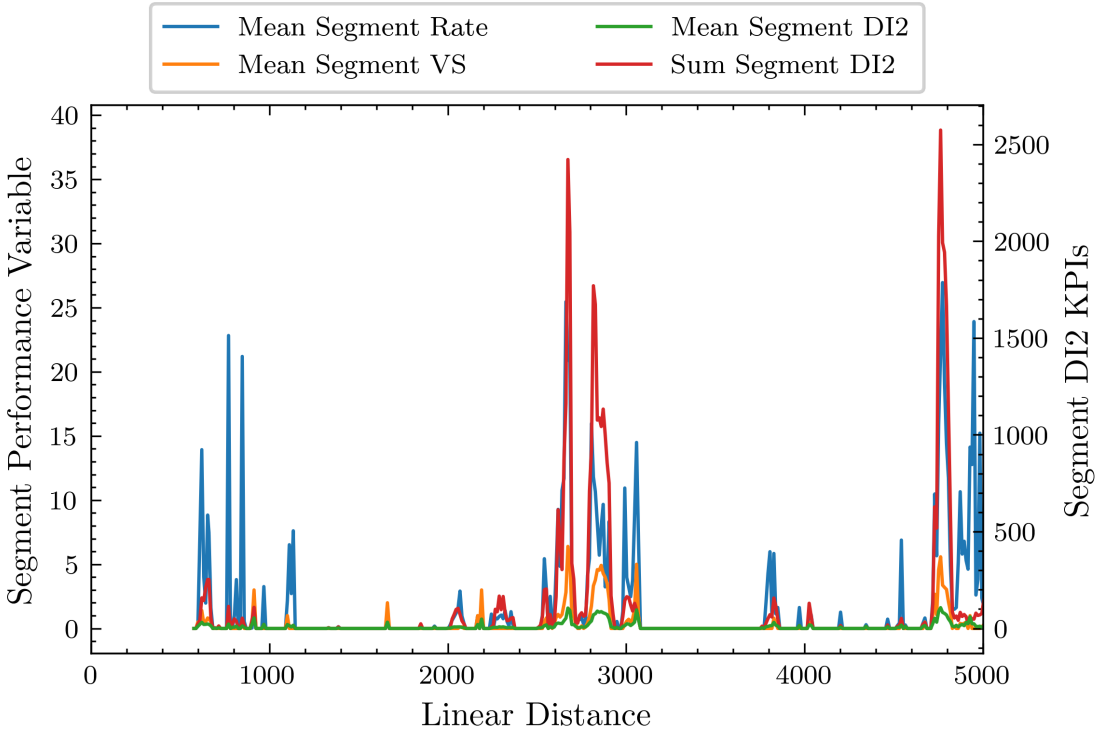
  where $j = 1, ...b_s$ are indices for all observations in segment s, (i.e. where there are no constraints on $D_{i+1} - D_i$)

Figure 6.5a and Figure 6.5b illustrate these segment performance indicators for the left and right rails respectively between 0 and 5000 yards. The *Mean Segment Degradation Rate* reflects the mean of all positive rates over the segment, and is clearly the most variable of all parameters. This is a result of the susceptibility to outliers, particularly where the temporal frequency for the segment is low. For instance, at approximately 1000 yards on the RH rail, there are spikes in the *Mean Segment Degradation Rate* which are much smoother for *Segment DI2 Sum* or *Mean Segment DI2 Sum* indicators. This is likely to be due to the low number of measurements available at this location, which reduces the reliability of the rate *KPI* ($KPI_s R_{mean}$) at these locations. Figure 6.6a and Figure 6.6b illustrate the correlation between *Mean Segment Degradation Rate* and *Segment DI2 Sum*, the marker colour represents the number of measurements in the

segment, with red indicating high, and blue is low. There is a weak positive correlation where, on the whole, the higher the rate the higher the sum, but there is a clear dependence on the number of measurements in the segment. Intuitively, where the measurement frequency is low, *Mean Segment Degradation Rates* are high and *Segment DI2 Sums* are low, therefore it is important that the analyses consider these factors.

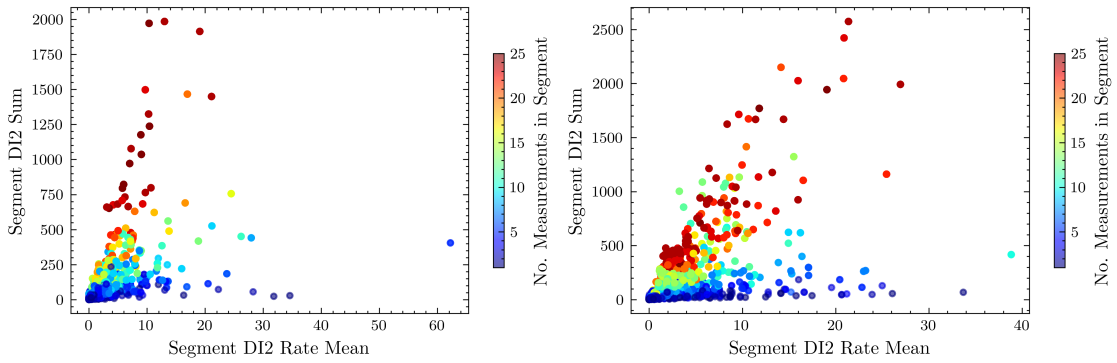


(A) Left Rail



(B) Right Rail

FIGURE 6.5: Segment Aggregated Rates of Change



(A) Left Rail (B) Right Rail

FIGURE 6.6: Segment KPIs

6.2 Application for Preventive Strategy

The preventive strategy presently employed by Network Rail is introduced in Chapter 2. The process uses a simple segmentation method, whereby the track is divided into curved and tangent track, and these are cyclically ground after 15 Mega Tonnes (MGT) and 45 MGT respectively due to the recognition that curves degrade faster than tangent track. The process is illustrated once more in Figure 6.7. Whilst this strategy is simple, it is based on theory developed in the early 2000's and not tested on the UK network, and has not been validated since its introduction. Rather than segmenting the track based purely on the track curvature, is is proposed that we capitalise on the available Eddy Current data and utilise historic measurements to develop a new method for segmenting the track.

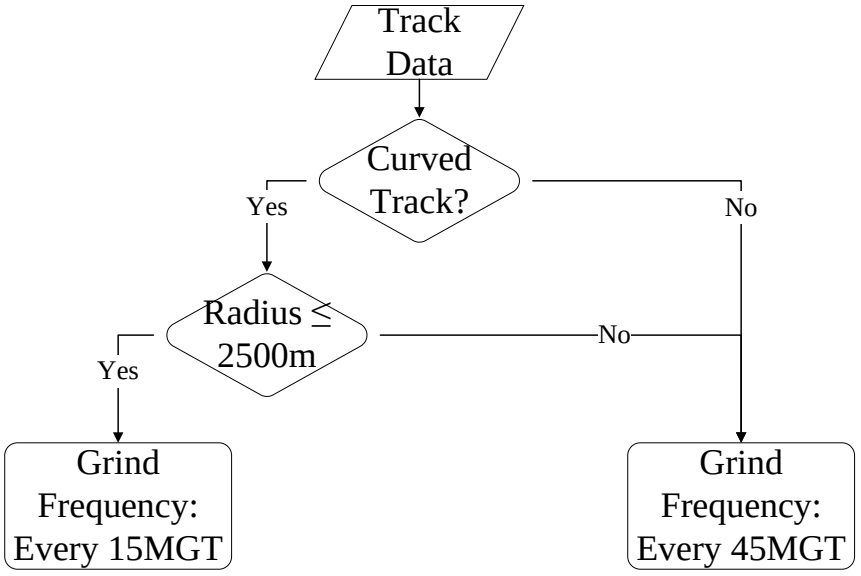
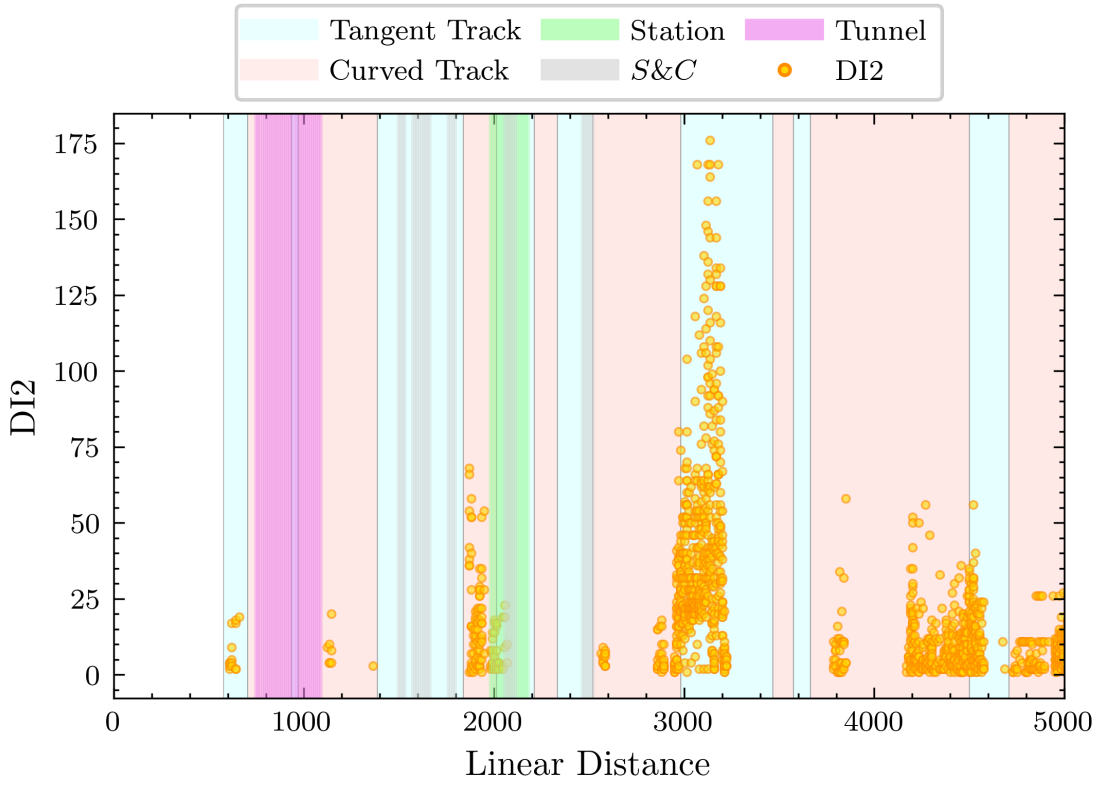


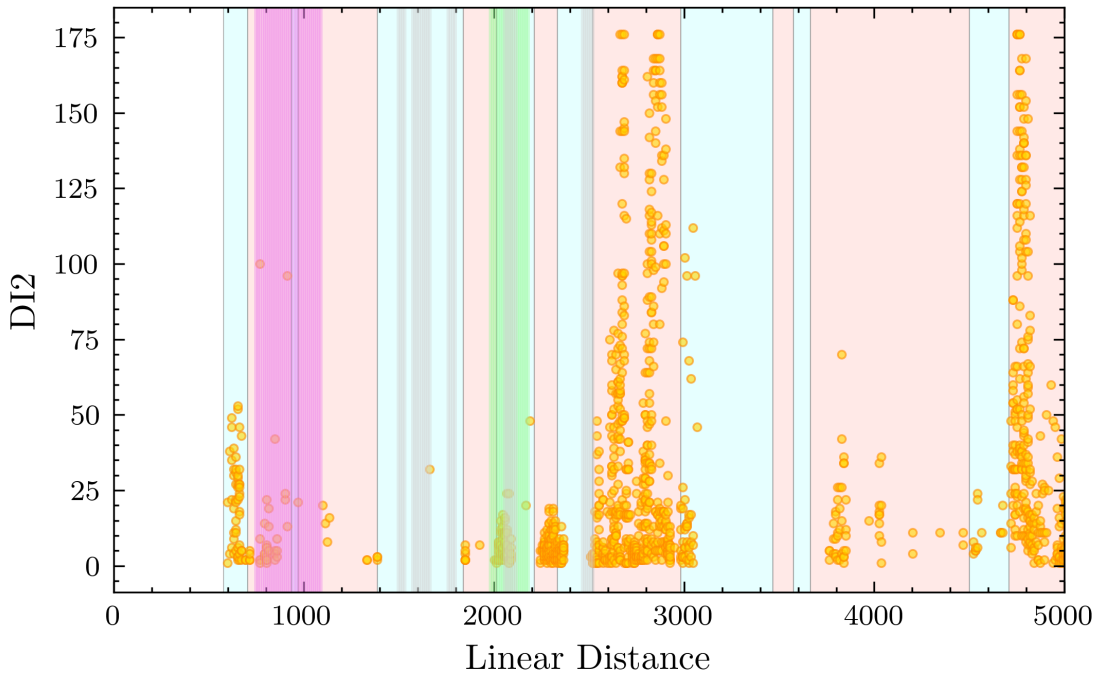
FIGURE 6.7: Current Preventive Grinding Strategy

Additionally, Figure 6.8a and Figure 6.8b illustrate the curved and tangent track sections which are utilised currently to determine the grind frequencies for the LHR and RHR. The historic EC data are overlaid using yellow markers (note that this is the processed data set with no smoothing, i.e. $P_{pro,0}$), and tunnels, stations and S&C locations are specified for reference (using the Track Summary data). The charts indicate that not all curves experience the same levels of damage, and therefore it may be beneficial to derive track segments based on the damage data as well as track curvature alone.

Further, for the LHR at approximately 3,000 yards the chart indicates that there is a tangent section of track experiencing high levels of damage. In this case, it is believed that this is likely due to spatial misalignments between the Eddy Current data and the track itself defined by the track summary data. Whilst the alignment process specified in Chapter5 aligns the Eddy Current data internally, it cannot align the Eddy Current data with the track. This is a problem where track parameters are captured using different sources, i.e. Eddy Current monitoring and measurement of track curvature are captured with different systems and are therefore not aligned.



(A) Left Rail



(B) Right Rail

FIGURE 6.8: Preventive: Initial Segmentation

6.2.1 Proposed Method for Improving Segmentation

A simple extension to the current strategy is to understand whether curved/ tangent track segments have low or high levels of RCF damage, which can be assessed using the processed data (P_{pro}). The basis of this approach is to understand if the curved/ tangent section may benefit from more or less intervention than is being implemented by current strategies. The following steps define the approach:

1. **Divide Track:** Split the track length into contiguous curved and tangent segments, and index each new section,
2. **Calculate Section KPIs:**For each curved/ tangent section, three performance indicators are calculated in the spatial and temporal domains. These KPI are extensions of those introduced in Section 6.1.1, which refer only to the temporal dimension to produce a KPI for each track segment. The following KPI represent the damage over the tangent/curved section in space and time:
   (a) **Mean of Segment Mean Degradation:** the segments means are averaged over the entire section,

$$KPI_{section}R_{mean,mean} = \frac{\sum_{m=1}^{M} KPI_s R_{mean}(m)}{M} \tag{6.6}$$

   where M is the total number of 11 yard segments in the site,
   (b) **Mean of Segment DI2 Sum:** the segment DI2 sum are averaged over the entire section,

$$KPI_{section}D_{sum,mean} = \frac{\sum_{m=1}^{M} KPI_s D_{sum}(m)}{M} \tag{6.7}$$

   (c) **Mean of Segment DI2 Mean:** the segment DI2 mean are averaged over the entire section,

$$KPI_{section}D_{mean,mean} = \frac{\sum_{m=1}^{M} KPI_s D_{mean}(m)}{M} \tag{6.8}$$

3. **Calculate Final Section Score:** Based on a sum of the scaled *Mean of Segment Mean Degradation*, *Mean of Segment DI2 Mean* and the *Total number of measurement*, the final score is calculated for the section.
4. **Rank Sections and Assign:** Order the sections by score and allocate the sections to three categories:
   - **Category 1- Low:** 33.3̇% of the sections with lowest KPI score,
   - **Category 2- Medium:** 33.3̇% of the sections with moderate KPI score,
   - **Category 3- High:** 33.3̇% of the sections with highest KPI score,

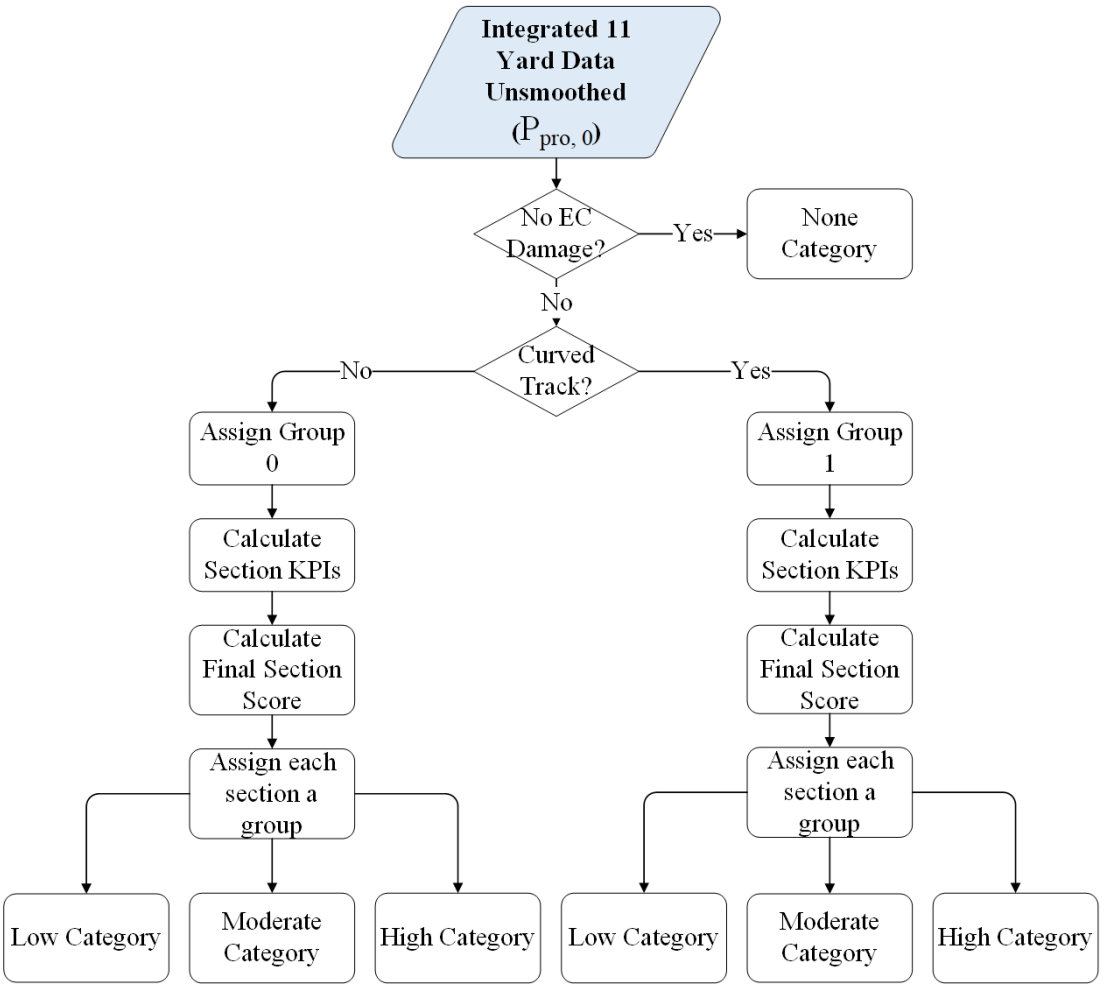


FIGURE 6.9: Flowchart for the Proposed Track Segmentation

6.2.2 Results

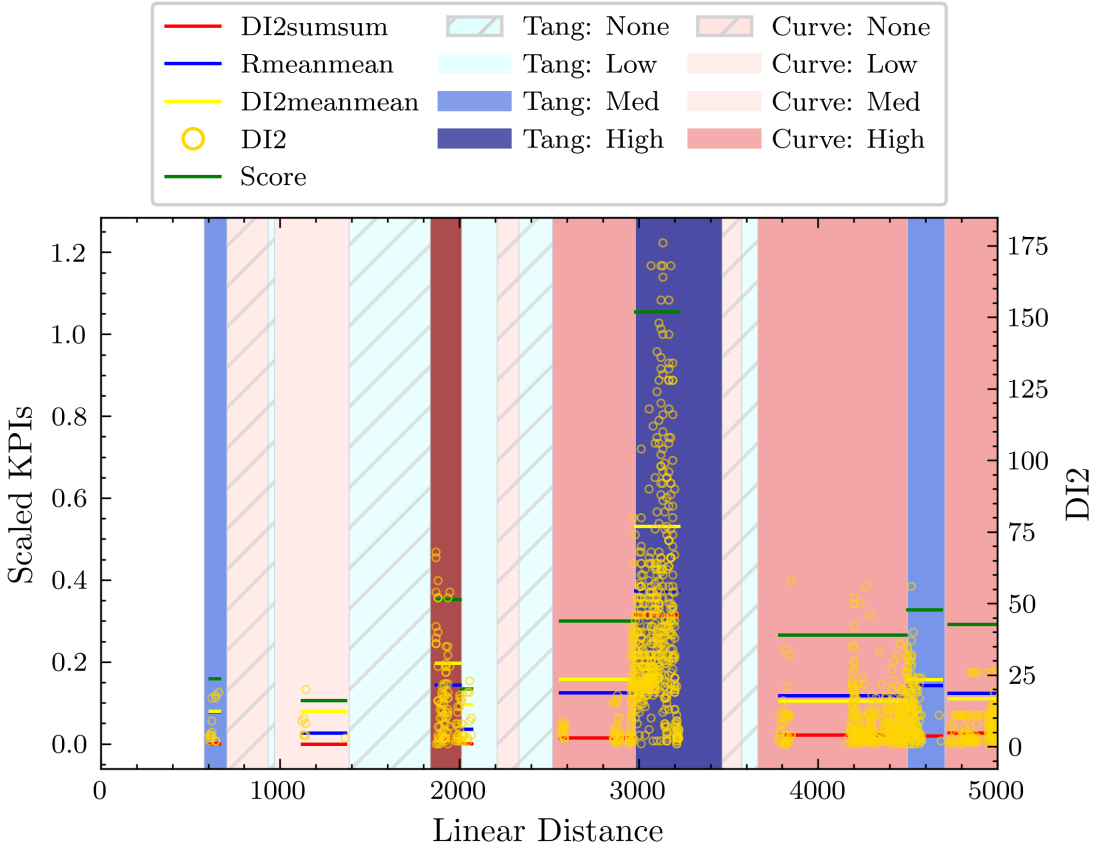
The output of the segmentation process shown in Figure 6.9 is to assign a new category to each tangent and curved section of track: (Low, Medium, High or None (no EC Damage)). In total, there are 172 curved and tangent sections for TL1 and Table 6.3 illustrates their resulting category assignment, with 27 sections having no recorded RCF damage, 5 of these are tangent and only 5 are curved sections. As one might expect, it is predominantly curved sections assigned to the 'Higher' category, however there are 17 tangent sections that are experiencing similar levels of damage and are therefore assigned to this category. On the other hand, there are a significant number of curved sections which, historically, are exhibiting low levels of RCF damage and thus it may be worth revising/ reducing the interventions carried out on these sections.

Additionally, Figure 6.10a and Figure 6.10b illustrate the new classifications; [high, moderate, low, none] and their relative positions along the track for 0- 5000 yards. The

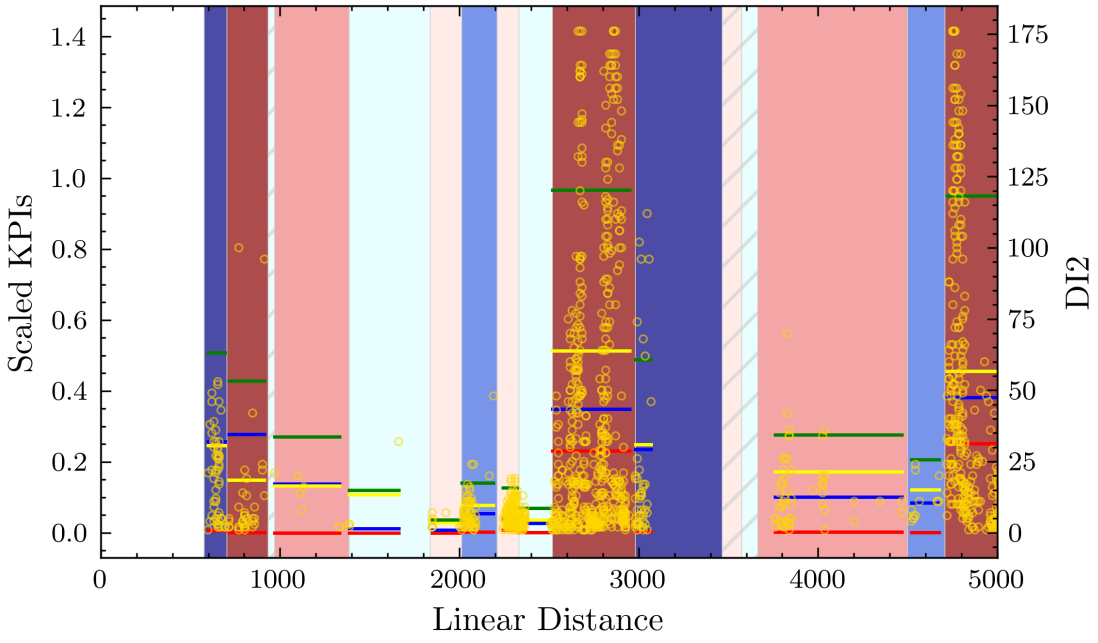
background colours represent the section type, with red tones indicating curved track and blue tones for tangent track. the darker the tone the higher the damage category.

TABLE 6.3: Section Category Allocations

Section Type	Section Score Category Defn.	Category	No. Sections
Tang.	-	None	22
Tang.	[0, 0.134]	Low	21
Tang.	(0.134, 0.33]	Mod	26
Tang.	(0.33, 1.415]	High	17
Curve	-	None	5
Curve	[0, 0.134]	Low	28
Curve	(0.134, 0.33]	Mod	22
Curve	(0.33, 1.415]	High	31



(A) Left Rail



(B) Right Rail

FIGURE 6.10: Preventive: Proposed

6.3 Application for Corrective Strategy

With respect to corrective grinding and milling, as discussed in Section 2.4, in the first instance routes must comply with Standard NR/L2/TRK/001/mod07 (Network Rail, 2018a). However, some routes such as Anglia have implemented their own methods to assign risk scores to RCF sites. The approach however relies heavily on expert judgements with regard to degradation rates and their correlation with factors such as traffic and curvature, the process is once again shown in Figure 6.11. The following proposal offers a demonstration for how the data generated by the proposed pre-processing process described in Chapter 5 ($P_{pro,0}$) can be used to inform a revised version of the existing Anglia Process.

6.3.1 Proposed Method for Corrective Strategy

The steps are described below and illustrated in Figure 6.12. Note that the examples presented do not include wear, traffic or other parameters at this stage for determination of site rank (highlighted in grey in Figure 6.12). It is intended here that the emphasis is placed on presenting the leveraging of rates and parameters determined solely from the processed EC data. Additional parameters can be readily incorporated at a later stage since they are already present in the *Integrated Data Set* $P_{pro,w}$.

1. **Data Selection:** The input data to this process is the *Integrated Data Set* with no smoothing applied, $P_{pro,0}$. The user must select the set relevant to the track section under analysis, for example: TL1: LTN1, 2100, Start Yards: 0, End Yards: 5,000, Track Type: Plain Line,
2. **Filtering:** Since we wish to see the most recent severe cracking events we wish to filter by:
    - **Date:** Only observations within 1 year of the analysis date are selected,
    - **Severity:** Only segments including at least one Severe or Very Severe crack are to be included,
3. **Determine RCF Sites:** Using a threshold distance of 66 yards, 11 yard segments are combined to make a site if they are located within the threshold distance of the 11 yard starting yard. The threshold distance represents the maximum distance between sites where no EC damage is indicated, any segments within 66 yards of each other are clustered together into a single site. Note, that at this stage the threshold distance (66 yards) is simply a first guess which provided reasonable results however in reality it will depend on what route maintenance teams decide is practical to consider a ‘site’,

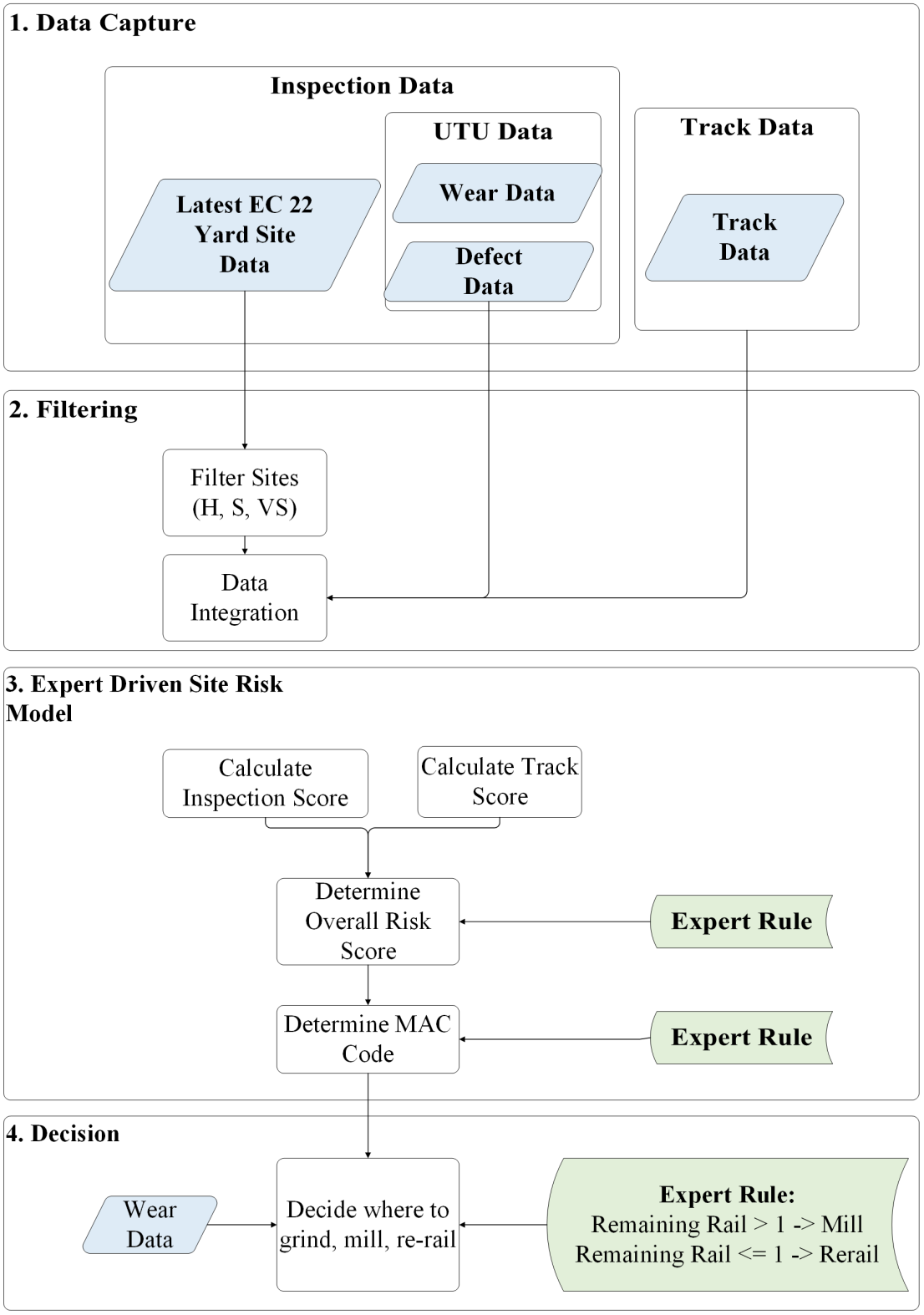


FIGURE 6.11: Flowchart for Current Anglia Corrective Maintenance Process

4. **Determine Site KPIs:** For each identified site, several KPIs are calculated for the user:
    - **EC Degradation KPIs:**
        - **DI2 Monthly Degradation Rate Average ($KPI_{corr,1}$) :** For each segment in the site an average monthly degradation rate is calculated, this is then subsequently averaged over all segments in the site,
        - **DI2 Monthly Degradation Rate Maximum ($KPI_{corr,2}$) :** the maximum of the segment monthly averages is taken over the site,
    - **EC Absolute KPIs:**
        - **DI2 Total Sum ($KPI_{corr,3}$) :** The total sum of DI2 for each segments is calculated,
        - **Normalised DI2 Total Sum ($KPI_{corr,4}$) :** The *DI2 Total Sum* is divided by the total number of measurements per segment, and yards in the site to generate a normalised value.
    - **Site Size:** Size of the RCF Site (in yards),
    - **Total Site Measurements:** This includes measurements in space and time over the site, it is a key indicator which guides the validity of the site. For example, some sites may identify high rates of change, however this is based on a single rate of change measurement between two observations. This site should therefore be given less weight than another with 100 observations over the site,
    - **Spatio-Temporal Measurement Ratio ($KPI_{corr,5}$) :** The number of yards in site affected with RCF/ the total number of measurements in the site. The lower the ratio the more trust we have that the site KPIs are supported with sufficient data,
5. **Rank Sites:** Currently the ranking is performed on a basic weighted combination of the DI2 Monthly Degradation Rate Mean ($KPI_{corr,1}$), Spatio-Temporal Measurement Ratio ($KPI_{corr,5}$), Normalised DI2 Total Sum ($KPI_{corr,4}$), with weights 45, 35, and 20 respectively. At this stage, these weights are selected simply to represent the perceived relative importance of each of these variables. For example, we are primarily concerned with sites which are deteriorating quickly (high *Average Degradation Rate*), but also must consider the robustness of this measurement through understanding the density of measurements in this site and prioritise these large high density sites (*Spatio-temporal measurement ratio*), and finally understand the sum of damage over all historic measurements in the site (i.e. has the site always had high levels of damage? which is determined using *Normalised DI2 Total Sum*). These weights can be adjusted following conversations with track experts and a thorough analysis of the sites generated. The resulting ranking is shown for a section of track 0-5,000 yards for both rail sides.

Future revisions of this procedure could incorporate KPIs such as:

- **Wear Degradation KPIs:**
  - Vertical Wear Monthly Degradation Rate Average,
  - Gauge Face Wear Monthly Degradation Rate Average,
  - Remaining Depth Monthly Degradation Rate Average,
- **Wear Absolute KPIs:**
  - Vertical Wear Average,
  - Vertical Wear Maximum,
  - Gauge Face Wear Average,
  - Gauge Face Wear Maximum,
  - Rail Remaining Depth Minimum,
  - Rail Remaining Depth Minimum,
- **Traffic KPIs:**
  - Average Passenger traffic over site during last period,
  - Average Freight traffic over site during last period,
- **Defect KPIs:** Total Defects in site at last measurement,
- **Track Geometry KPIs:**

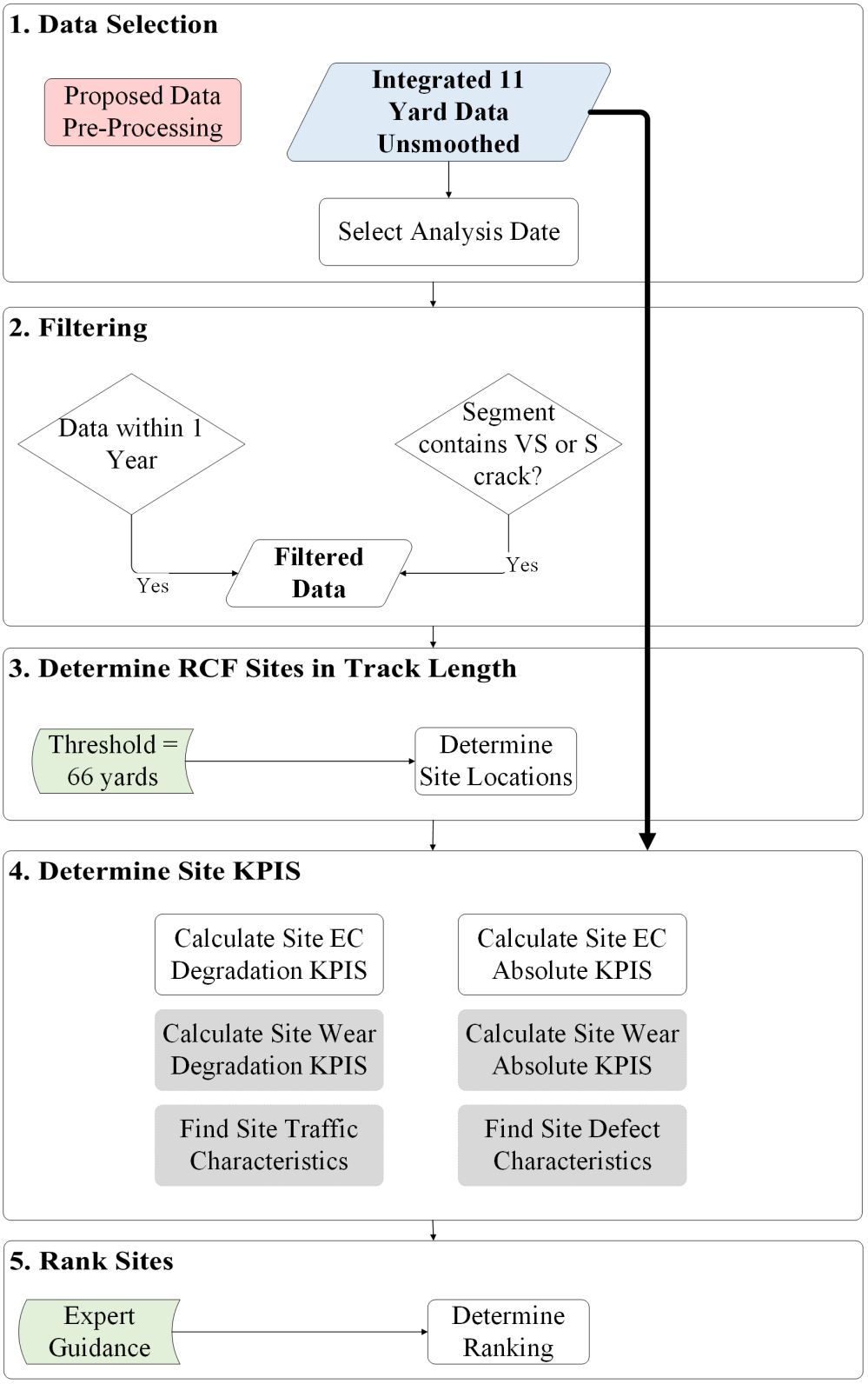


FIGURE 6.12: Flowchart for the Proposed RCF Severe Site Identification

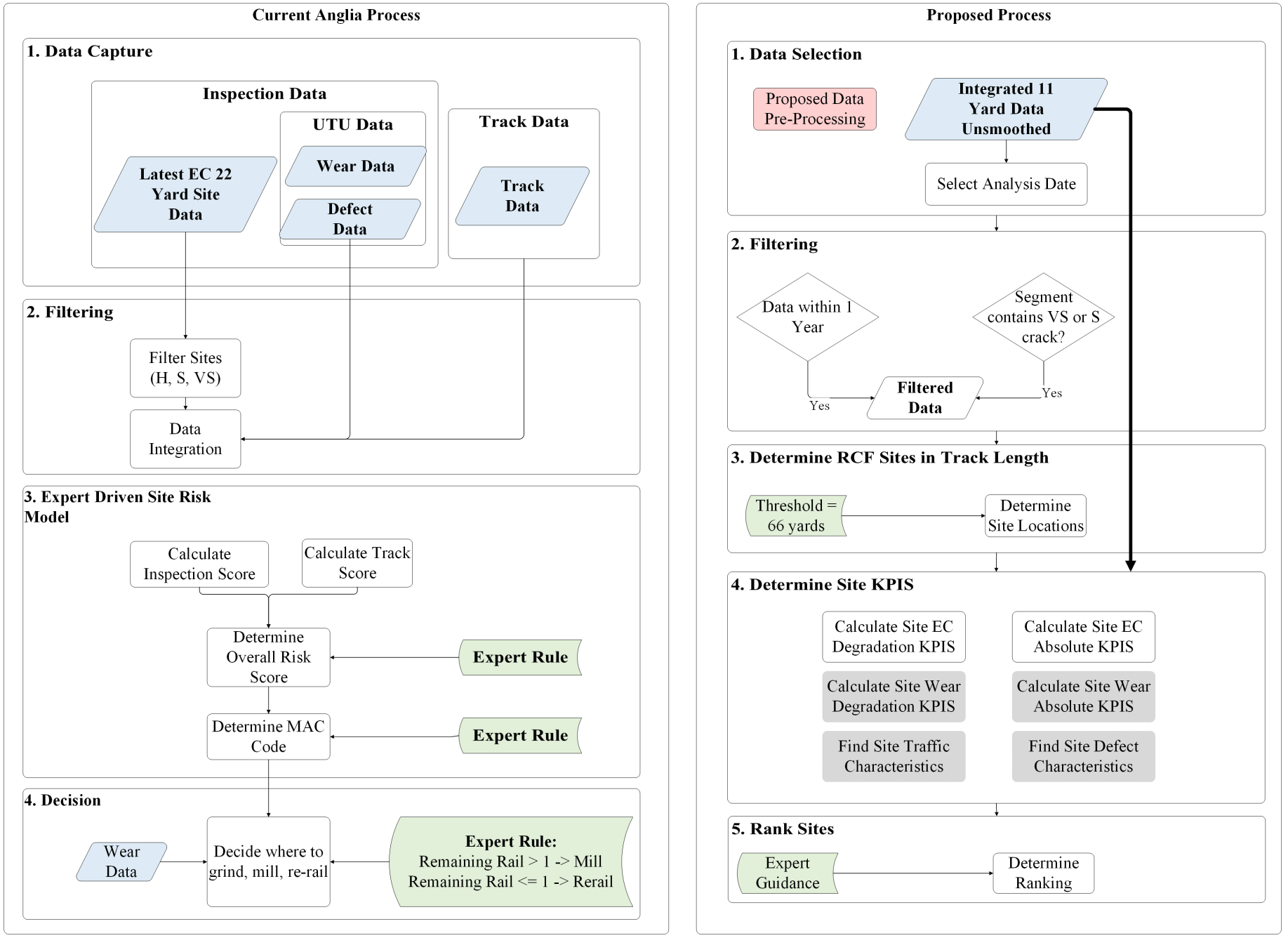


FIGURE 6.13: Comparison of Corrective Strategies: Existing vs. Proposed

6.3.2 Results

The proposed method is demonstrated for the TL1 integrated data set between 0 and 5,000 yards for the left and right rails, where 17 sites are identified. The sites and their KPIs which govern the site priority are highlighted in Table 6.4:

- **Rank:** the *Rank* for each of the sites based on the calculated KPI,
- **Site Start:** the linear distance defining the start location of the site,
- **Rail Side:** left/ right rail,
- $KPI_{corr,1}$**:** the *DI2 Monthly Degradation Rate Average*,
- $KPI_{corr,4}$**:** is the *Normalised DI2 Total Sum*,
- **VS Sum Av. per yard:** represent the normalised total sum for the number of VS cracks,
- **S Sum Av. per yard:** represent the normalised total sum for the number of S cracks,
- **Site Length:** represents the length of the identified RCF site in yards,
- **No. Site Measurements:** refers to the total number of observations identified for the RCF site (over space and time),
- **No. Yards RCF:** the number of yards in the site affected with RCF,
- $KPI_{corr,5}$**:** the *Spatio-Temporal Measurement Ratio* is the ratio of the *Total No. Site Measurements* and the *Total No. Yards Measurements*.

For example, row 1 shows the highest ranking RCF site determined using this procedure, which is located at 4,741 yards and occurs on the right hand rail. This site has the highest monthly rate of degradation ($KPI_{corr,1}$) and the averaged sum of DI2 over all ST cells $KPI_{corr,4}$, 7.16 and 7.48 respectively compared with all other sites in the table, and the average sum of VS and S cracks are also relatively high at 0.38 and 0.09. Additionally, the site is made up of 88 yards (8 segments), has 176 recordings of RCF in space and time, and 100% (8) segments are affected with RCF, which increases the trust in the generated scores.

The site ranked second corresponds to a site at 2882 yards on the left rail. Although there are considerably more measurements across the site (597), and it spans 341 yards (31 segments) where 27 of these segments are affected with RCF, the severity of the degradation is less concentrated than the top ranking site.

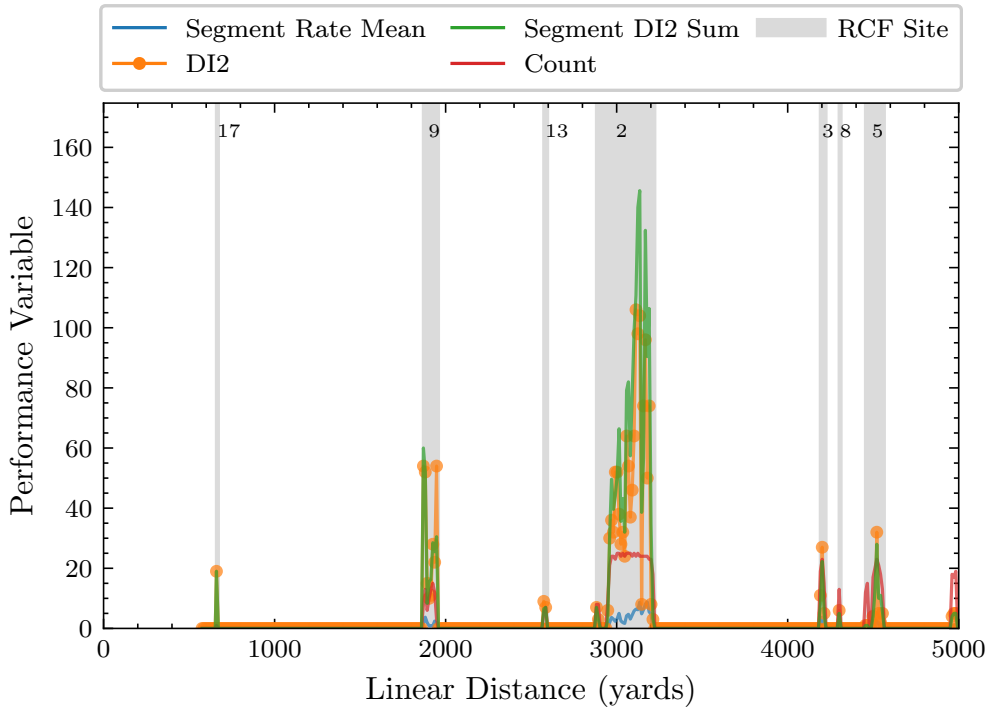
On the other hand, the site ranked 10th is located at 847 yards with a high average degradation rate ($KPI_{corr,1}$), in fact it is the second highest compared with all other sites. However, when we look at the other measures such as $KPI_{corr,4}$, and see that there are only 4 measurements corresponding to this site with only 1 segment reporting RCF damage it significantly reduces the priority of the site since it is such as small site and likely that the high degradation rate includes an outlier.

Figure 6.14a and 6.14b highlight the hot spots and labels their ranking. Additionally the *Segment Rate Mean* (the mean degradation rate is calculated for each segment), and the *Segment DI2 Sum* (DI2 is summed for each segment across all temporal measurements) are illustrated in blue and green respectively. Further the Count (number of segment measurements) is shown in red and the most recent DI2 in orange. Finally, the 17 RCF hotspots are shown in grey, and labelled according to their calculated ranking.

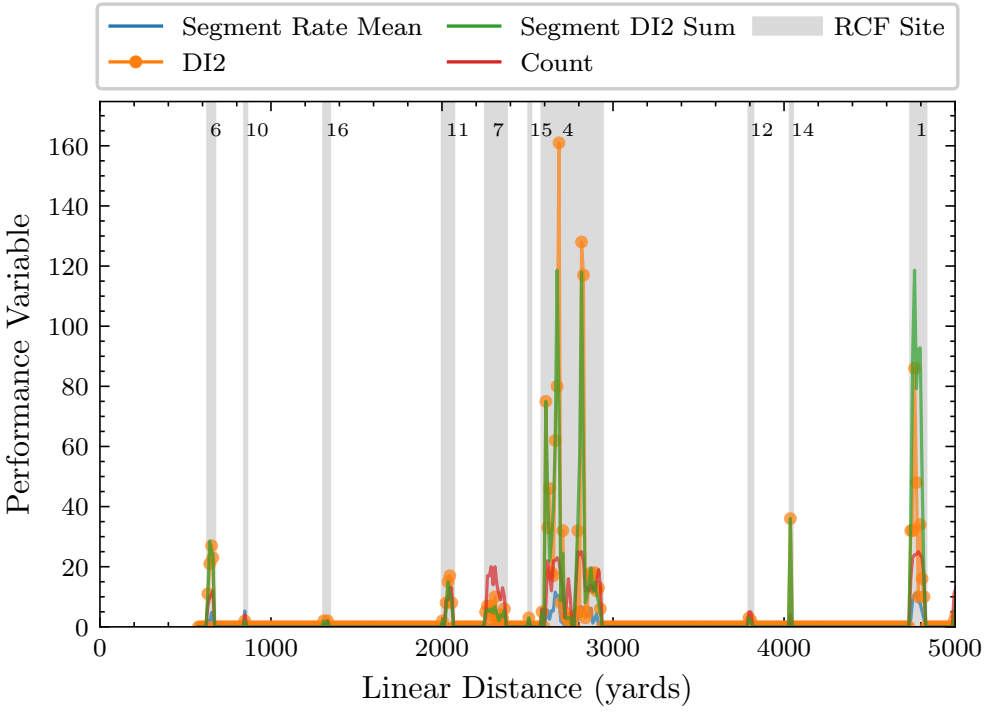
TABLE 6.4: RCF Site Identification and Ranking Procedure Results: TL1: 0- 5000 yards

Rank	**Site Start**	**Rail Side**	$KPI_{corr,1}$	$KPI_{corr,4}$	**VS Sum Av. per yard**	**S Sum Av. per yard**	**Site Length (Yds)**	**No. Site Measurements**	**No. Yards RCF**	$KPI_{corr,5}$
1	4741	Right	7.16	7.48	0.38	0.09	88	176	8	22
2	2882	Left	4.03	5.03	0.19	0.08	341	597	27	22.11
3	4191	Left	2.17	1.60	0.01	0.06	33	57	3	19
4	2585	Right	3.80	2.64	0.13	0.01	352	454	29	15.66
5	4455	Left	1.63	0.73	0.01	0.01	110	139	8	17.38
6	627	Right	3.01	3.05	0.10	0.02	44	40	4	10
7	2255	Right	0.36	0.48	0	0	121	148	11	13.45
8	4301	Left	0.90	0	-	-	11	13	1	13
9	1870	Left	1.76	3.07	0.10	0.01	88	80	8	10
10	847	Right	5.30	0	-	-	11	4	1	4
11	2002	Right	0.40	0.75	0	0	66	45	5	9
12	3795	Right	2.05	0.45	0	0	22	10	2	5
13	2574	Left	0.13	1.23	0	0	22	12	2	6
14	4037	Right	0.80	0	0	-	11	4	1	4
15	2508	Right	0	0	-	-	11	2	1	2
16	1309	Right	0	0.18	0	0	33	3	2	1.5
17	660	Left	0	0	0	-	0	11	1	1

To optimise this process it is recommended that the method is fine tuned with track experts to meet their needs. However the principle is demonstrated whereby a relatively simple aggregation of data could be used to assist in selecting sites for maintenance activity; given that the data has been appropriately processed. Wear, traffic, defect and track geometry indicators could also be readily incorporated.



(A) Left Rail



(B) Right Rail

FIGURE 6.14: Corrective: Hotspots

6.4 Concluding Remarks

This chapter has proposed some simple approaches for utilising the processed modelling data set to inform preventive or corrective maintenance procedures.

Firstly, we introduced a methodology for allocating curved and tangent sections to an enriched set of groups based on historical RCF damage. The new groups represent no damage, low, moderate and high levels of damage, and provide the foundations for implementing various interventions schemes depending on group, rather than simply by curvature. For example, sections in the high damage category may indicate that a route may need to significantly revise their intervention strategies, through more frequent grinding/ milling, more lubrication, or even the potential for future rail replacements of high performance rail to reduce the incidence of RCF. On the other hand, sections assigned to the low category, or where no RCF damage occurs may look at reducing their interventions, presenting the opportunity for cost savings. This approach is focussed on the segmentation of the track based on historic data rather than proposing new interventions/ grinding frequencies. In order to better understand the temporal aspect and time to intervene, it is imperative that the quality of the interventions data collected is improved. For instance, understanding of the effectiveness of each intervention and how this impacts the initiation/ propagation of RCF cracks must be gained. Parameters such as grinding depth, number of passes, and desired profile etc., should be collected in addition to simply the start and end points of machine grinding. Additionally, for rail replacements, the temporal resolutions are currently accurate only to the year, which is not helpful for understanding rates of degradation. Simultaneously, the frequency of the Eddy Current measurements should reflect these interventions, such that RCF measurements are available before and after each intervention. And finally, a replacement for ACTRAFF data is required to understand the cumulative tonnage over each track section between EC measurements.

Secondly, a method for highlighting RCF hotspots and ranking these based on historical damage data is described. The results presented here are intended as a demonstrator for how the degradation data can be used to influence how risk is assigned to a hotspot, rather than utilising other track characteristics defined by track experts. It is recommended that collaboration with route experts is still required to fine tune the process and define some of the parameters defined in the approach, such as the choice of additional data sources when defining site scores, and the thresholds for site definition.

In summary, two simple methods have been presented to demonstrate the potential for utilising pre-processed historic data to augment maintenance strategies. In order to work towards more consistent, evidence-based approach to maintenance, it is important to develop these strategies and use experts to supplement and fine tune

data driven approaches. The following chapter goes a step further into the realm of data-driven modelling by investigating the use of regression analysis and machine learning algorithms to predict RCF damage using the pre-processed data.

Chapter 7

Machine Learning for RCF Prediction

The third and final part of this thesis aims to assess and develop a data-driven model which predicts the evolution of RCF. The problem of predicting or classifying an outcome of interest (in this case rail damage due to RCF) can be tackled using supervised learning methods. Classification and regression are two specific forms of prediction, the former predicts a variable of discrete nature, whereas the latter involves continuous value output (Huang et al., 2020). In addition to traditional statistical methods, machine learning offers new tools for solving supervised learning problems especially in instances where other methods are unsuitable (Jiang et al., 2020). This chapter will firstly discuss the problem formulation and the methodology employed to develop and evaluate a data-driven model. Secondly the modelling results are presented which comprise of three parts:

1. **Results 1:** A comparison of the full set of model configurations and different algorithms, a subset of model configurations are determined for hyper-parameter tuning,
2. **Results 2- Determine Optimal Parameters:** Hyper-parameter tuning is performed for each of the algorithms given the chosen smoothing parameter and input features. Additionally some analysis of these optimised models are presented,
3. **Track-Ex Comparison Analysis:** Demonstrates a comparison of the optimised models with physics-based approach WLRM using the Track-Ex implementation.

7.1 Problem Formulation

As described in Chapter 3, we seek to solve a supervised learning task. If we consider two *input data sets* $P, P' \in \xi$ representing eddy current cracking observations, described by the location, time, and other defining parameters for each observation. The labels, or *output data* are denoted by $DI, DI' \in X$ which indicate the degree of damage associated with P and P' respectively. Machine learning algorithms aim to 'learn'; a functional relationship between data P (training dataset) and label DI, in order to predict unseen labels DI' from the new data set P' (test data set). More formally the supervised learning problem is to find a function $f = f(P, DI, \theta)$, where θ denotes the set of adjustable model parameters that may be estimated during the training phase. The estimated function can be used on a new dataset to predict unseen labels (Valente, 2008).

$$\hat{DI'} = f(P', DI, \theta) \tag{7.1}$$

where $\hat{DI'}$ denotes an estimate of the labels DI'. In this study, we consider labels with continuous values, a regression problem.

7.2 Proposed Modelling Methodology

The regression methodology is illustrated in Figure 7.1 and described below, the process utilises each of the pre-processed data sets generated by the process defined in Chapter 5 for smoothing windows: [0,2,4,5,6,8,10,12,14,15,20,25,30], the entire process produces 2184 different configurations. The use of the IRIDIS High Performance Computing Facility at the University of Southampton was imperative in completing these runs.

Additionally, as explained in Appendix B, there are numerous possible hyper-parameter settings that could be used for model training and the approach is often undergo a process known as 'tuning'. At this stage, the initial algorithm hyper-parameter settings are identified and shown in Table 7.4. These parameters are a result of preliminary analysis, recommended 'default' algorithm settings, the degree of noise and sparsity in the data, and with the objective to minimise over-fitting. Once the initial results are generated for each run configuration, the best feature set and smoothing parameter will be identified. The algorithms will then be tuned using these settings to determine the 'best' model. This approach is used in order to reduce the number of runs required.

Note that ideally hyper-parameter tuning is conducted using a separate verification set of data, in order to reduce the bias in the test results, however in this case due to the limited number of observations in the temporal dimension a separate verification set is not used, and the final results are presented for the same test set for which the algorithms are tuned. In future, when more data are available the process should be repeated.

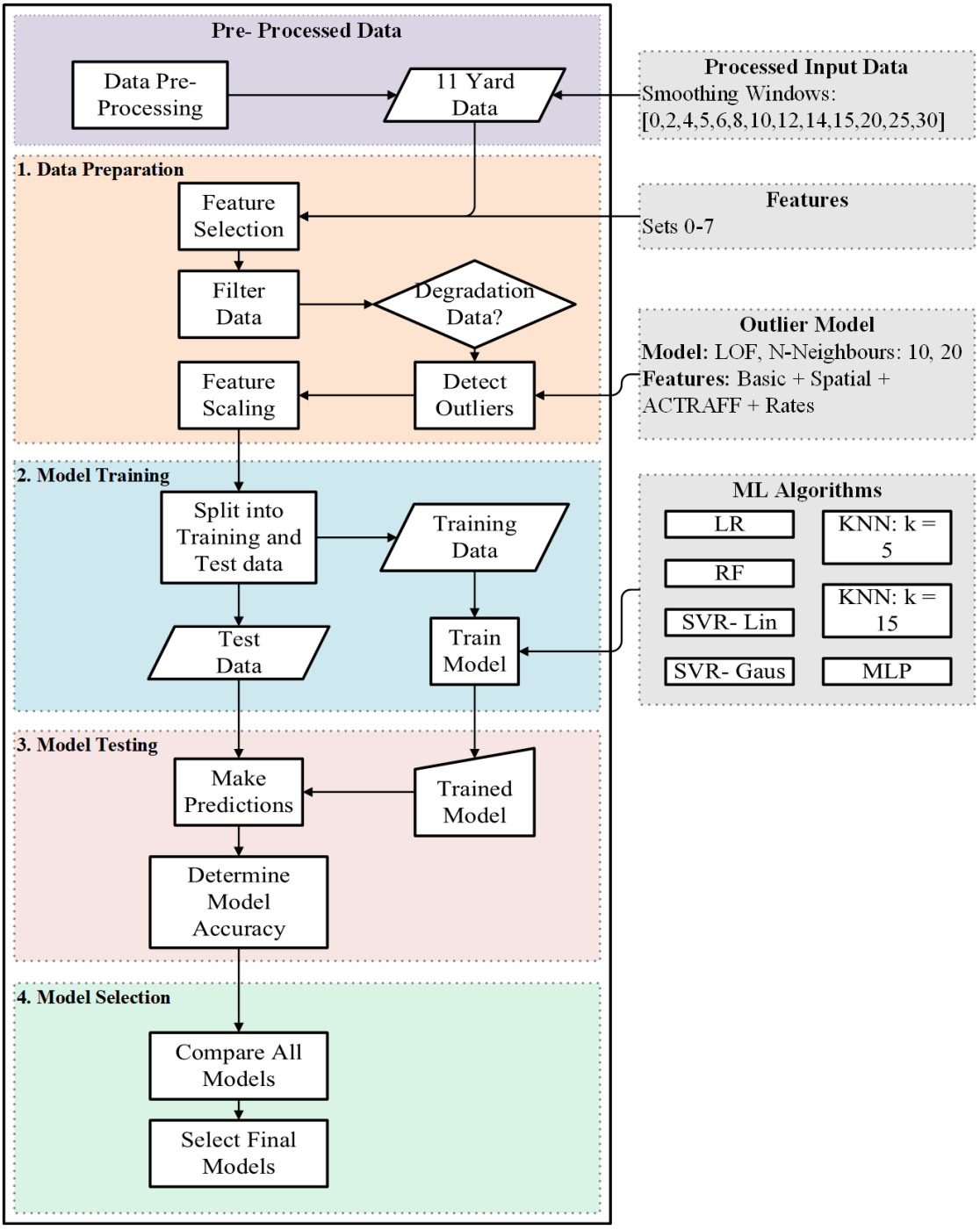


FIGURE 7.1: Modelling Methodology

1. **Pre-processing:** The pre-processing stage refers to the preparation of the modelling dataset with respect to a set of run configurations. To determine the effect of modifying specific input parameters, numerous configuration are tested as defined in Table 7.1.

    - **Feature Selection:** Feature Sets 0-7 are utilised as input to the model, these are defined in Table 7.2 and Table 7.3,
    - **Filter Data:** Only the degradation data is retained for regression analysis, observations which have detected an improvement since the last measurement in the segment are removed from the data set,
    - **Outlier Detection:** The Local Outlier Factor (LOF) is utilised to determine an outlier feature for each observation, each of the following configurations are tested:
        - None: No outlier detection is performed, outlier features are not included in the feature set,
        - LOF: N-Neighbours= 10,
        - LOF: N-Neighbours = 20,

2. **Scale Continuous Data:** In data-driven approaches, it is common to scale the continuous features in the dataset before learning. The motivation behind scaling is that variables measured at different scales do not contribute equally to the model fitting and therefore might cause bias. This is also model dependent, with some distance based models such as support vector regression often dependant on scaling of the input data. There are many methods used for scaling; we use the following:

    - **Robust Scaling**: As the name suggests, robust scaling methods are robust to outliers, the method for scaling is similar to min/max scaling but uses the interquartile range (rather than the range to scale the data). It follows the formula:
    $$x_{scaled} = \frac{x_i - Q_1(x)}{Q_3(x) - Q_1(x)}$$
    where Q_1 and Q_3 are the first and third quartiles respectively

3. **Training/ Test Split:** The data are divided into a training set and a test set by randomly splitting the data into training and testing sets. 10 different randomly generated splits will be generated, with the training data apportioning 66% of the data and 33% for the test data,

4. **Train Models:** Training the learning algorithm with input data.

    (a) Model 1: Linear Regression (LR),

    (b) Model 2: Support Vector Regression (SVR) (kernel- linear),

(c) Model 3: Support Vector Regression (SVR) (kernel- gaussian),

(d) Model 4: Random Forest (RF): ,

(e) Model 5: K-Nearest Neighbours (KNN): k = 5,

(f) Model 6: K-Nearest Neighbours (KNN): k= 15,

(g) Model 7: Multi Layer Perceptron (MLP): 50 layers,

These models comprise some of the most commonly applied algorithms utilised within the literature for this type of regression problem. The implementation of these algorithms was performed using the python library 'scikit-learn' (Pedregosa et al., 2011). These algorithms are discussed in detail in Appendix B.

5. **Make Predictions:** Trained models are used to make predictions using the test data,

6. **Model Evaluation:** For each run configuration the predictions are evaluated using performance metrics aggregated across the 10 data splits, note that $\hat{y}_i$ is the predicted value of the i-th sample and y_i is the corresponding true value for total n samples:

   (a) **Mean R^2:** The mean average of R^2 for all splits. R^2- the coefficient of determination, represents the proportion of the variation in the dependent variable that is predictable from the independent variable. The best possible score is 1.0 and negative values are possible since the model can be arbitarily worse than choosing the expected value. A constant model that always predicts the expected value of y ($\bar{y}_i$), disregarding the input features, would get a score of 0.0. The coefficient of determination for a training sample, size n is defined:

   $$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \tag{7.2}$$

   where n is the number of observations in the sample, $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ and $\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} \epsilon_i^2$
   The mean $\bar{R}^2$ over all splits k is thus:

   $$\bar{R}^2(y, \hat{y}) = \frac{\sum_{k=1}^{K}(R_k^2)}{K} \tag{7.3}$$

   where K is the total number of data splits, in this study $K = 10$,

   (b) **Range R^2 :** The range of the R^2:

   $$\text{Range}(R^2(y, \hat{y})) = \max(R^2) - \min(R^2) \tag{7.4}$$

(c) **Mean Run Time:** The run time for each split is calculated and averaged over the 10 splits,

7. **Model Selection:** At this stage we select the best set of input features and smoothing parameters,

TABLE 7.1: Modelling Runs

Model Input	**Parameters**
	tbd
Smoothing Window	[0,2,4,5,6,8,10,12,14,15,20,25,30]
Models	1: Linear Regression, 2: Random Forest, 3: Support Vector Regression: Linear, 4: Support Vector Regression: Gaussian, 5: KNN: 1, 6: KNN: 2, 7: MLP
Input Features (X)	Feature Set 1 Feature Set 2 Feature Set 3 Feature Set 4 Feature Set 5 Feature Set 6 Feature Set 7
Outlier Analysis	None, LOF- k = 10, LOF- k = 20
Scaling	Robust

TABLE 7.2: Modelling Input Feature Set

Feature Set No.	**Modelling Input Feature Set**
0	Basic Features
1	Basic Features, Spatial Features
2	Basic Features, Spatial Features, ACTRAFF Features
3	Basic Features, Spatial Features, ACTRAFF Features, Wear Features
4	Basic Features, Spatial Features, ACTRAFF Features, Wear Features, Wear Rate Features
5	Basic Features, Spatial Features, ACTRAFF Features, Wear Features, Wear Rate Features, Time Since Intervention
6	Basic Features, Spatial Features, ACTRAFF Features, Wear Features, Wear Rate Features, Time Since Intervention, Defect Features
7	Basic Features, Spatial Features, ACTRAFF Features, Wear Features, Wear Rate Features, Time Since Intervention, Defect Features, Lagged Damage

TABLE 7.3: Modelling Feature Definitions

Name	Feature
Basic	Linear Distance, Time Since Measurement, Number of Segment Measurements, Rail Side
Spatial	Track Curvature, Cant, Speed, EMGTPA, High/ Low Rail, Track Category, Station, Tunnel, Distance to Station, Distance to Tunnel, Distance to S&C
ACTRAFF Features	Total Tonnes, Passenger Tonnes, Freight Tonnes, Total Trains, Passenger Trains, Freight Trains
Wear features	Avg. Gauge Face Wear, Max. Gauge Face Wear, Avg. Vertical Wear, Max. Vertical Wear, Avg. Field Side Wear, Max Field Side Wear, Avg. Head Width Remaining, Min Head Width Remaining, Avg. Gauge Face Remaining, Max. Gauge Face Remaining, Avg. Remaining Rail Depth, Min Remaining Rail Depth
Wear Rate Features	Monthly Rates, Segment Rates as above.
Time Since Intervention	Time since Estimated Intervention
Defect Features	Defect Occurred
Lagged Damage	Shifted Segment Damage Index

TABLE 7.4: Algorithm Hyper-parameters

Index	Algorithm	Hyper-parameters
1	Linear Regression (LR)	N/A
2	Random Forest (RF)	Implementation: Breiman Cutler, No. Trees in the forest: 100, Size of Feature Set Considered at each split: All Features, Max. Depth tree: 10, Splitting Criterion: Squared-Error,
3	Support Vector Regression (SVR) Linear	Kernel: Linear, Regularisation: 1, Epsilon: ,
4	Support Vector Regression (SVR) Gaussian	Kernel: Gaussian, Regularisation: 1, Width (Gamma): 0.1, Epsilon:
5	K-Nearest Neighbours (KNN) 1	k = 5, Distance Metric: Euclidean Weights: Uniform
6	K-Nearest Neighbours (KNN) 2	k = 15, Distance Metric: Euclidean Weights: Distance
7	Multi Layer Perceptron (MLP)	No Hidden Units: 100, Solver: ADAM, Regularisation: [0.0001]

7.3 Results 1: Regression Model Comparisons

This section presents the results of model runs defined in Table 7.1. The full set of results are presented in order to determine a number of optimal models for further development.

Figure 7.2 presents the results for all configurations where Outlier Features are not included in the modelling. The smoothing window is indicated along the x-axis and the model mean accuracy R^2 is shown along the y-axis. Each marker and colour represents the ML algorithm and feature sets respectively. The inclusion of the Lagged Damage indicator variables clearly has the largest impact on model accuracy since typically the highest accuracies are achieved for models trained with Feature Set 7. On the other hand the lowest accuracies are generally achieved when only including the basic feature set- Feature Set 0, which includes the observation linear position, time and segment frequency. Additionally; as anticipated, in most cases model accuracy increases with smoothing, however a balance must be struck between model accuracy and correct data representation. Whilst smoothing helps to reduce noise, it also blurs the data, reducing peaks and details that may be imperative for describing the underlying phenomena. The smoothing process also introduces additional data points at locations which would have otherwise been zero, for example in the original dataset there are 6,921 observations, but this increases to 34381 observations when the smoothing window is 30. The increase in data set size is reflected in the computation times, which are shown in Figure 7.3. In particular, some algorithms, such as KNN and RF are especially sensitive to increasing data set size.

Figure 7.5 presents box plots for each algorithm and smoothing level representing the range in model accuracy (R^2) across all 10 splits of the data. The model accuracies, particularly for the LR, SVR and MLP algorithms are generally low but increase with smoothing parameter. The spread in R^2 value is also low indicating low sensitivity to the change in input data set, for algorithms such as linear regression this is unsurprising due to its low-variance nature. The instance-based (K-Nearest Neighbours (kNN)) and tree-based methods however have much higher degrees of variability, and are considerable more unstable without clear patterns that accuracy is improving with smoothing parameter. Ideally it is desirable to choose a model that indicates a much lower sensitivity to the data set, since it is more likely to be robust to new, unseen data sets.

The same results are presented in Figure 7.5 with Feature Set 7 as input. In this case it is clear that there is a significant improvement in prediction accuracy in all cases, and a reduction in variability, except in the case of the MLP algorithm. It is likely that in the case of the MLP algorithm that this is due to the hyper-parameter settings. The MLP algorithm can run into convergence issues due to initial settings (data set), such as high learning rates or a low number of iterations, in which case the solution may

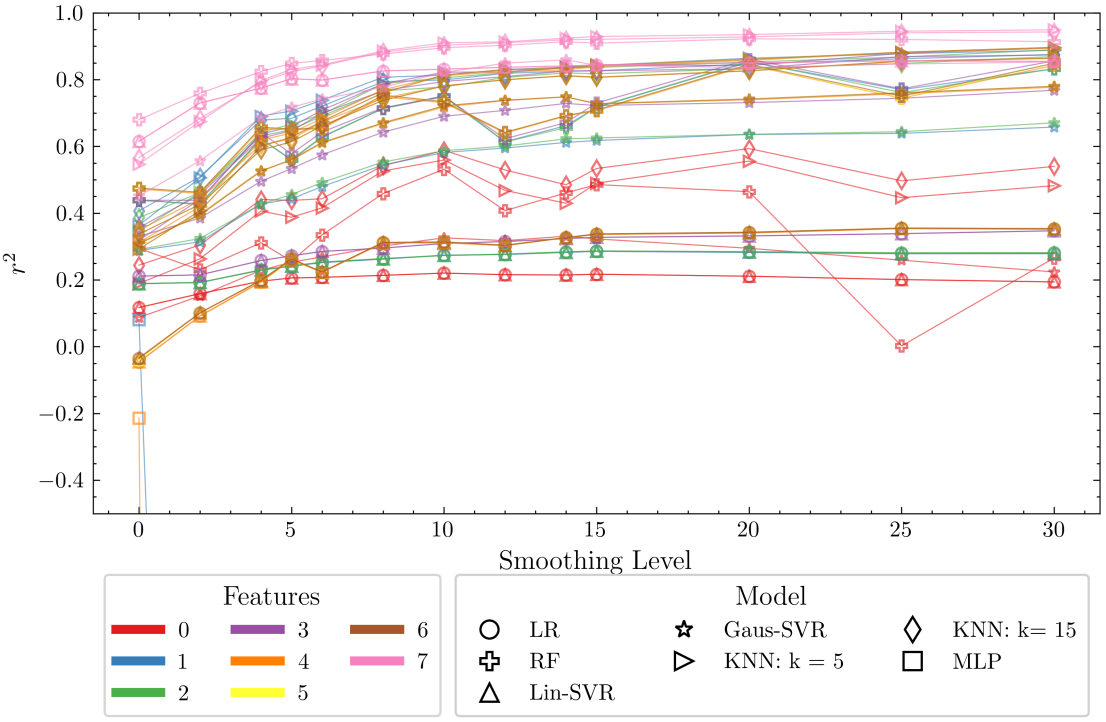


FIGURE 7.2: Results: Model Accuracy, Outlier Model = None

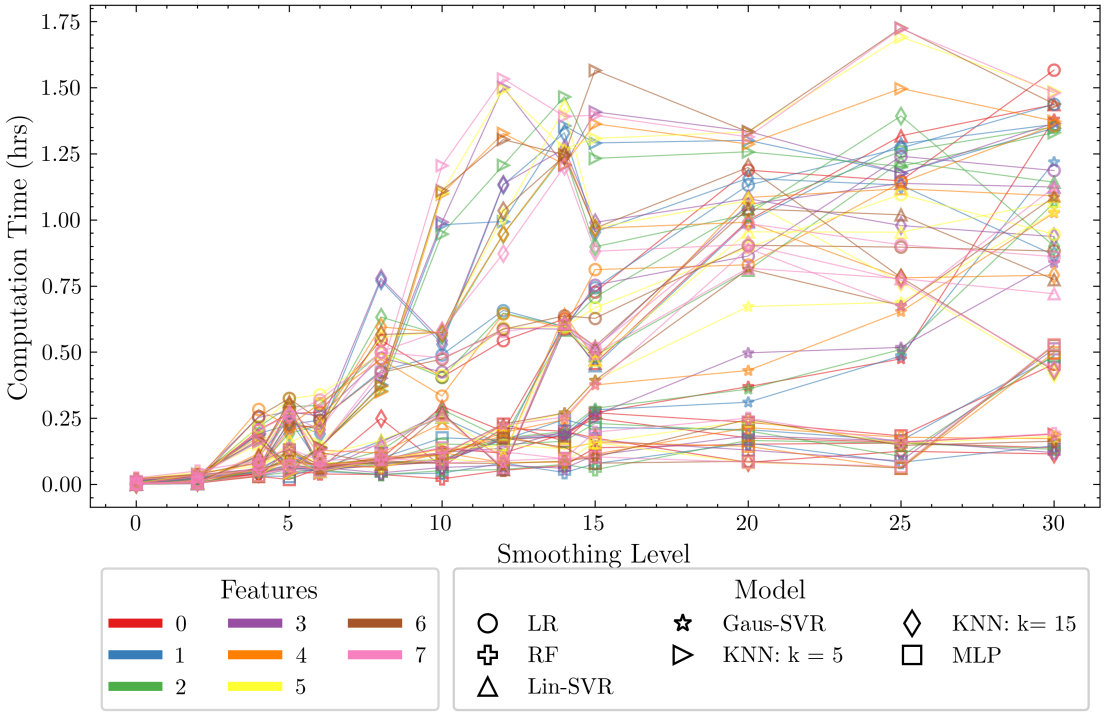


FIGURE 7.3: Results: Computation Time, Outlier Model = None

converge to local minima (Lo et al., 2012). In this case it is highly desirable to tune the hyper-parameters to determine a set which are more optimal, for example increasing regularisation, increasing step size or the maximum number of iterations may help the algorithm converge consistently to a global optimum.

To select the optimal smoothing parameter, we use Figures 7.2 and 7.3, and a heuristic known as the 'elbow method'. Often used in determining the optimum number of clusters in k-means clustering (Syakur et al., 2018), the elbow method interprets the turning point of a graph representing a model parameter of interest vs. the optimal parameter to be selected (whether it be the number of clusters, k, or here, the smoothing parameter, w). The method is used to identify the point where diminishing returns (with respect to model accuracy (R^2) and data set sensitivity) are no longer worth the additional cost, which in this case is computation time and correct data representation. A smoothing window $w = 5$ is thus selected, which in real terms represents 5 segments- a smoothing window of 55 yards.

Clearly, Feature Set 7 results in the highest accuracy. The full set of tabulated results are presented in Appendix C in Table C.1. The following section presents the results of hyper-parameter tuning where the smoothing parameter is set to 5 and input feature set 7 is used, these parameters will thus be used in the Track-Ex comparison analysis.

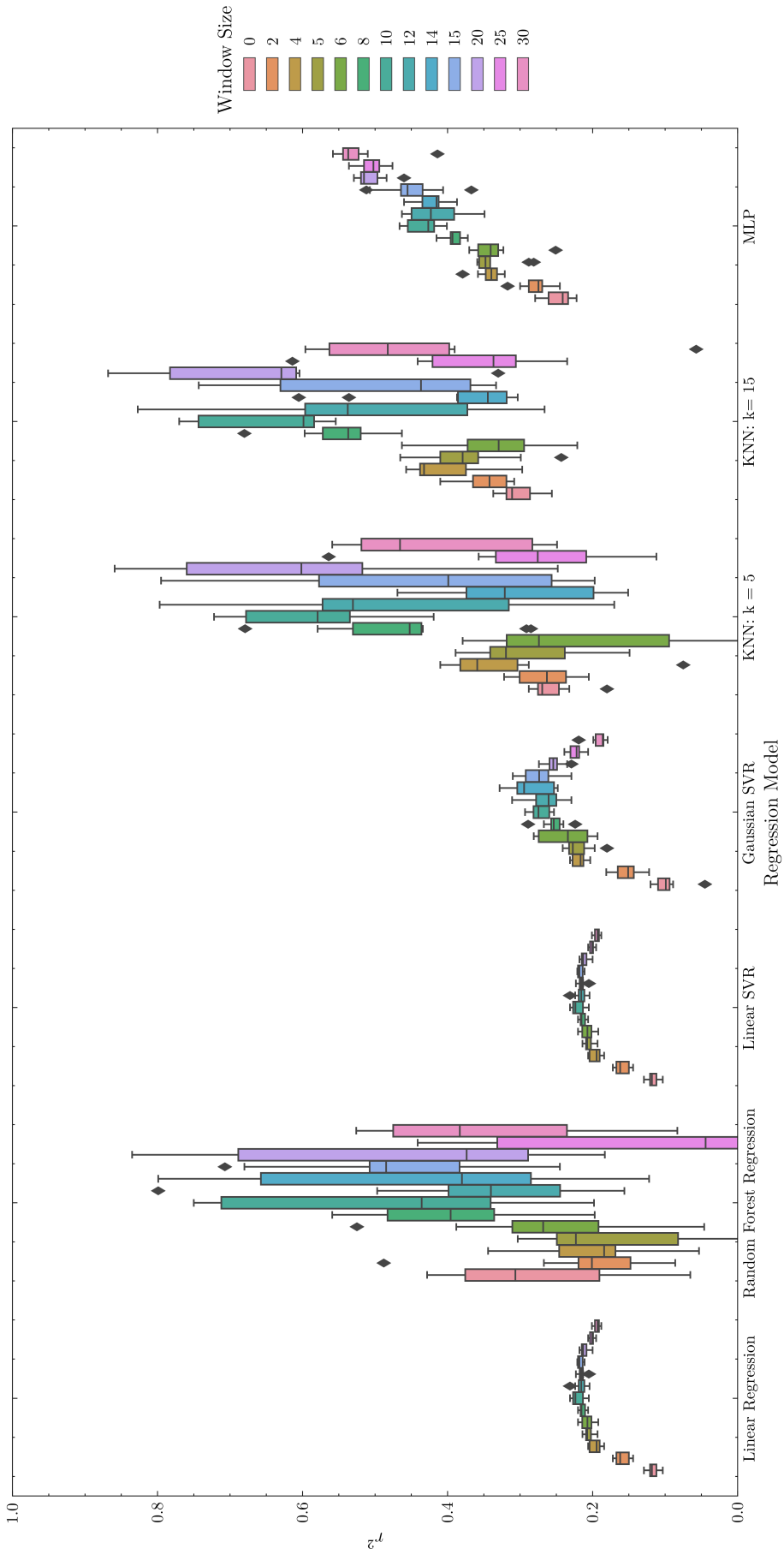


FIGURE 7.4: Results: Variation in R^2 Score Across 10 Data Splits, Outlier Model = None, Feature Set 0

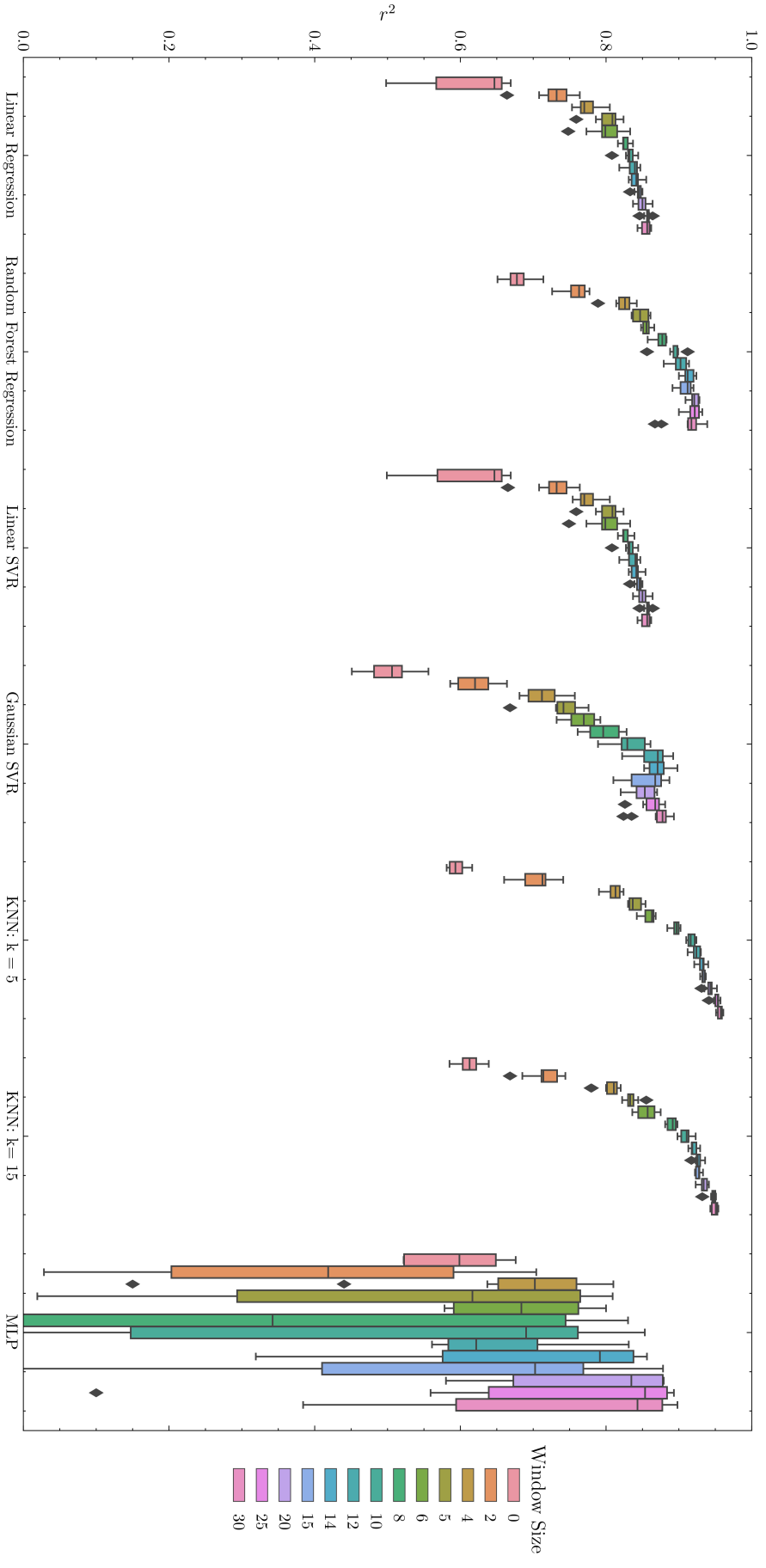


FIGURE 7.5: Results: Variation in R^2 Score Across 10 Data Splits, Outlier Model = None, Feature Set 7

7.3.1 Results 2: Regression Models with Optimal Parameters

Continuing from the preceding section, here we illustrate the results of hyper-parameter tuning where the smoothing parameter is set to 5, and input feature set 7. The hyper-parameter variations are indicated in Table 7.5 and the optimal parameters along with new results fo each algorithm are contained in Table 7.6 (note the full results are contained in Appendix C.2). Additionally an overview of feature importances and model residuals are included.

TABLE 7.5: Algorithm Hyper-parameter Tuning

Index	Algorithm	Hyper-parameters
1	Linear Regression (LR)	N/A
2	Random Forest (RF)	Implementation: Breiman Cutler, No. Trees in the forest: [25, 50, 75, 100], Size of Feature Set Considered at each split: [50% of features, All Features], Max. Depth tree: [5, 10, 15], Min. No Samples Per split: [20, 40, 60, 80], Splitting Criterion: Squared-Error,
3	Support Vector Regression (SVR) Linear	Kernel: Linear, Regularisation: [0.1, 1, 10, 100], Epsilon: [0, 0.01, 0.1]
4	Support Vector Regression (SVR) Gaussian	Kernel: Gaussian, Regularisation: [0.1, 1, 10, 100], Width (Gamma): [0.001, 0.01, 0.1] , Epsilon: [0, 0.01, 0.1]
5	K-Nearest Neighbours (KNN) 1	k = [5, 10, 20, 40, 80, 160], Distance Metric: Euclidean Weighting: Uniform
6	K-Nearest Neighbours (KNN) 2	k = [5, 10, 20, 40, 80, 160], Distance Metric: Euclidean Weighting: Distance Based
7	Multi Layer Perceptron (MLP)	No Hidden Units: [(10,), (50,), (100,), (150,), (10, 10), (50, 50), (100, 100), (150, 150), (10, 10, 10), (50, 50, 50), (100, 100, 100), (150, 150, 150),], Solver: ADAM, Regularisation: [0.0001, 0.001, 0.01, 0.1, 1] Max Iterations: [500, 1000]

Table 7.6 indicates the optimal parameters, the R^2 before optimisation (BO) and after optimisation (AO), additionally the table shows the range in R^2 for each fold of the data to illustrate the model stability. Notably the R^2 accuracy has increased for each algorithm, and in particular for the MLP a greater level of stability has been achieved due to hyper-parameter tuning. The range over all 10 splits is now only 0.03, and the model R^2 is significantly increased. Stability is achieved through increasing regularisation (α), the number of iterations and the number of hidden layers and neurons within each hidden layer. Additionally the accuracy of the Gaussian SVR has

TABLE 7.6: Optimal Algorithm Hyper-parameters

	Algorithm	**Hyper-parameters**	**Max R^2: BO**	**Range R^2: BO**	**Max R^2: AO**	**Range R^2: AO**
1	LR	N/A	0.8	0.068	0.8	0.068
2	RF	Implementation: Breiman Cutler, No. Trees in the forest: 100, Size of Feature Set Considered at each split: All Features, Max. Depth tree: 10, Splitting Criterion: Squared-Error Min Samples Split: 20	0.85	0.029	0.856	0.026
3	SVR Linear	Kernel: Linear, C: 0.1, dual: False, ϵ: 0, Loss Function: Squared Epsilon Insensitive, Max. Iterations: 1000, Tolerance: 0.0001,	0.8	0.067	0.803	0.066
4	SVR Gaussian	Kernel: Gaussian, C: 100, ϵ: 0.1, γ: 0.01, Tolerance: 0.0001,	0.72	0.104	0.868	0.047
5	KNN 1	k = 5, Distance Metric: Euclidean Weights: Uniform	0.83	0.022	0.86	0.025
6	KNN 2	k = 5, Distance Metric: Euclidean Weights: Distance	0.82	0.036	0.86	0.03
7	MLP	α: 1, Hidden Layer Sizes: (150, 150, 150), Max. Iterations: 1000, Solver: ADAM,	0	18936	0.86	0.068

also been significantly improved by increasing regularisation and decreasing the γ parameter. γ is the kernel coefficient which defines how much influence a single training example has, if γ is too large, then the radius of the area of influence of the support vectors only includes the support vector itself and even with considerable regularisation the model cannot be prevented from over-fitting to the data. On the other hand if γ is too small, then the model is too constrained and cannot capture the complexity in the data. The Random Forest, Gaussian Support Vector, kNN and MLPs have the highest accuracies ($\sim$0.86) amongst the models which is unsurprising due to their ability to model highly complex functions. The optimal choice of algorithm will be made once a comparison is made with the physics-based approach in the following section.

7.3.1.1 Model Residuals

Model Residuals provide an overview of regression model performance, in particular the distribution of the model errors and how these highlight any violations in classical statistical assumptions. Table 7.7 tabulates the mean, standard deviation and variance for each of the optimal models using a single split of the data (hence accuracy scores will not match the averaged scored provided in Table 7.6). Figures 7.6 to 7.12 illustrate how the model predictions correlate with the observed values and how these errors are distributed, and the marker colours represent the rail type, i.e. high rail, low rail or tangent rail.

In the case of a linear regression model, the resultant model also provides a function which represents the contribution of each independent variable (or feature) with the target variable. However, in this case there are indications of *multicollinearity*, such that two or more of the features are dependent on one another. Instinctively, it is highly likely to be the case that the 'Lagged Damage Indicator' is highly co-dependent on other features, since we know from previous research and the literature that damage is a function of features such as curvature, wear, traffic etc. therefore it follows that the lagged damage is also. Additionally, there is significant variation in coefficient estimates across all of the 10 folds of data, and coefficient signs are sometimes counter-intuitive. For a linear regression model, multicollinearity affects the accurate determination of model parameters (coefficients), since the estimation procedure struggles to distinguish between the effect of co-dependent terms, this in turn affects utilising the model for making further inferences.

Another observation from the residuals is that in some cases the errors are not constant in size, i.e. the residuals are increasing with the observed damage suggesting that the models predict damage more accurately at lower values. This characteristic is known as *heteroscedasticity*, errors which are non-constant, and is most apparent in the case of the Linear Regression and Support Vector based models. Once again, heteroscedasticity is particularly an issue for Linear Regression, since the ordinary least squares method (used to determine optimal model parameters) assumes that all residuals are drawn from a population that has a constant variance.

Additionally, for the models developed using the Linear Regression and Linear Support Vector algorithms, the residual plots indicate negative predictions of damage, which are possible due to the nature of the algorithms, hence generating some very large errors in these cases. Since none of the training examples have negative target values, this is not possible when using any of the other algorithms. On the whole, each of the algorithms appear to under-predict the expected RCF damage.

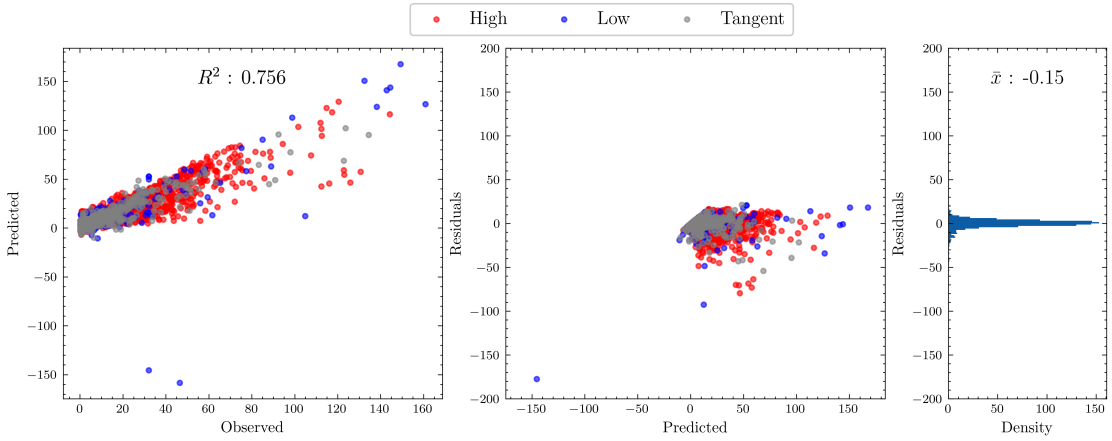


FIGURE 7.6: LR Residuals

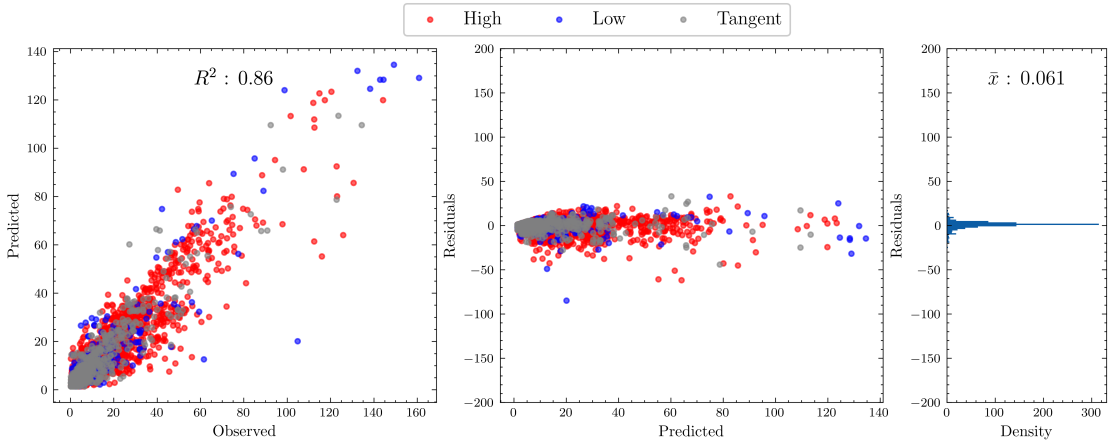


FIGURE 7.7: RF Residuals

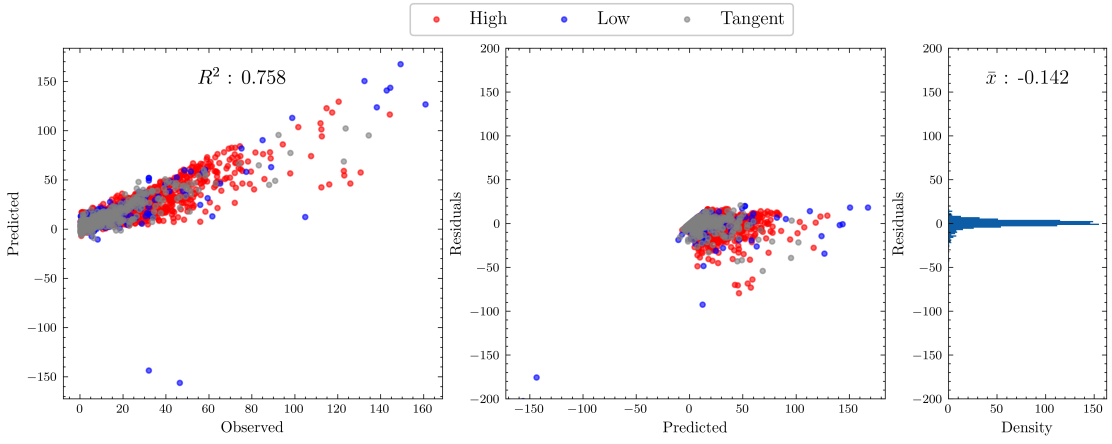


FIGURE 7.8: SVR-Lin Residuals

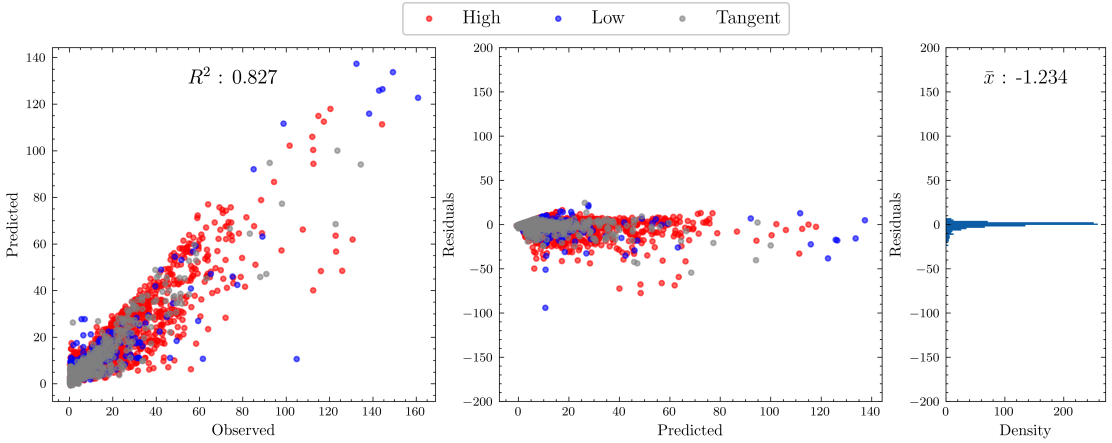


FIGURE 7.9: SVR-Gaus Residuals

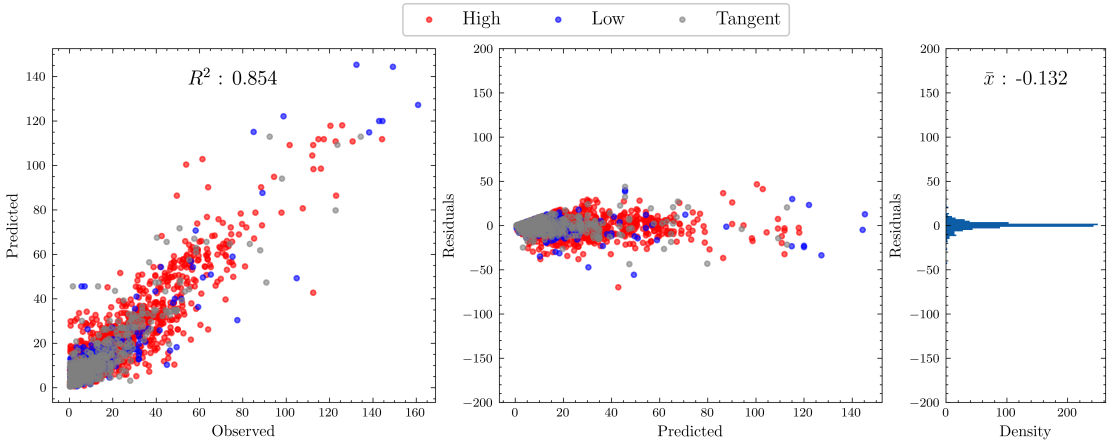


FIGURE 7.10: KNN1 Residuals

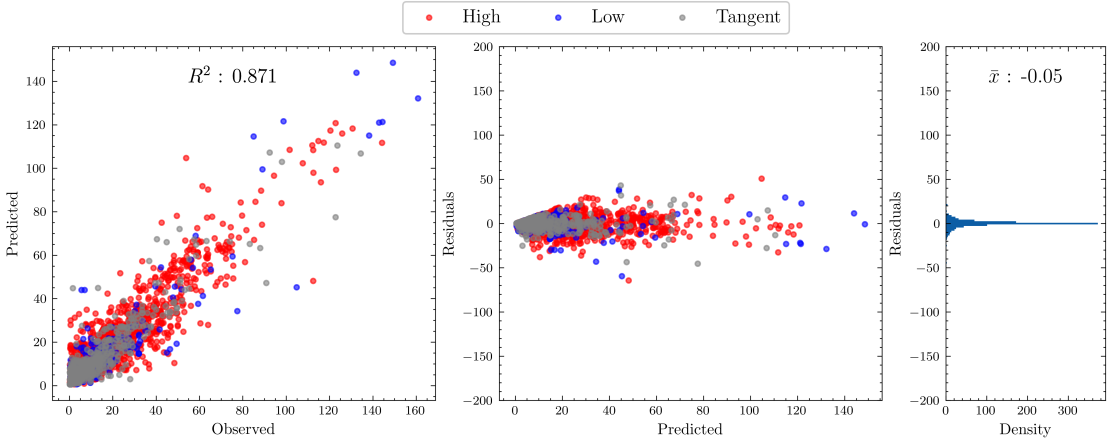


FIGURE 7.11: KNN2 Residuals

TABLE 7.7: Model Residuals Summary

Model No.	Model	Mean	SD	Var
0	LR	-0.150	8.271	68.409
1	RF	0.061	6.269	39.295
2	SVR-Lin	-0.142	8.246	67.996
3	SVR-Gaus	-1.234	6.856	47.005
4	KNN 1	-0.132	6.394	40.882
5	KNN 2	-0.050	6.013	36.152
6	MLP	0.204	6.043	36.523

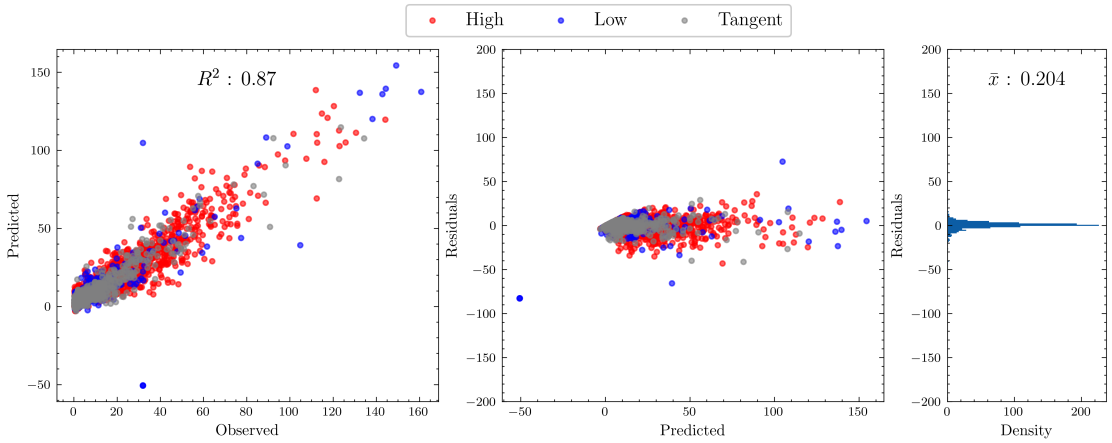


FIGURE 7.12: MLP Residuals

7.3.1.2 Feature Importances

Whilst machine learning models are able to map very complex relationships between inputs and outputs, they are often termed 'black boxes', such that it is difficult to understand the decisions that drive the model predictions. Traditional parametric statistical models such as linear regression enable the analyst to make statements about how inputs and outputs are related, and determine the uncertainty of these estimates using probability theory. However, neural networks can be made of hundreds of neurons, and random forests can be constructed using hundreds of trees with numerous nodes, and these techniques make it almost impossible to derive such insights analytically.

Deriving numerically determined feature importances is hence a useful technique in understanding how much a model's predictions depend on certain inputs. One method for determining relative feature importance is *permutation feature importance*. Once a model has been trained, the technique determines the decrease in a model score when a single feature is randomly shuffled. Ideally, random reordering of a column ought to result in reduced accuracy, since the new data has little or no correlation with real-world statistics. Model accuracy suffers most when an important feature, that the model was quite dependent on, is shuffled. The benefit of this

technique is that it is model agnostic, and can be calculated many times with different permutations of the feature.

The permutation importance algorithm is indicated in Algorithm 8:

Algorithm 1: Permutation Importance Algorithm (Breiman, 2001)

Input: Fitted Predictive Model m, training dataset D, input features $\mathscr{F}_j$

Output: Feature Importances $\{i_1, ...i_J\}$

Compute the reference score s of the model m on data D (i.e. R^2);
for *each feature* $\mathscr{F}_j$ *in* $\{\mathscr{F}_1, ..., \mathscr{F}_J\}$ **do**
  for *each repetition* k *in* $\{1, ..., K\}$ **do**
    Randomly Shuffle column j of dataset D to generate a corrupted version of the data $D_{k,j}$;
    Compute the score $s_{k,j}$ of model m on correupted data $D_{k,j}$;
  end
  Compute importance i_j for feature $\mathscr{F}_j$ defined as $i_j = s - \frac{1}{k}\sum_{k=1}^{K} s_{k,j}$
end

Since there are numerous model configurations given the 10 folds for cross validation and the 7 different algorithms, feature importances were calculated across these configurations. Firstly for each configuration, the relative importance of each feature was calculated by running the model and removing each feature in turn and re-training the model. Following this the average importance across all configuration is calculated as well as the sum of the feature rank in each configuration, these are illustrated for the top 20 ranking features in Figure 7.13 and Figure 7.14 respectively. The full set of results are tabulated in Table C.9 and Table C.8. In the case of the average importance, a larger score indicates higher importance, whereas for the cumulative rank a lower ranking score indicates a higher importance.

In both cases, the lagged damage indicator ranks highest overall, and generally the ACTRAFF and wear features are highly ranked for both cases, which correlates well with prior knowledge of RCF initiation and degradation. It should be observed, however, that the strength of the dependence on the lagged damage indicates the presence of autocorrelation in the data. This characteristic may present issues with any independence assumptions asserted by the models, in particular in the case of linear regression models where the independence assumption is required to make statistical inferences using the model. In the case of other machine learning models such as K-Nearest Neighbours and Decision trees this violation of independence may have more of an effect on the model accuracy estimated using model validation techniques such as *random cross validation*. When using random k-fold cross validation to test the accuracy of a model, the assumption is that the data are independent and identically distributed (Arlot, 2009) which therefore makes the cross-validated

prediction error a good approximation to the true expected prediction error (Burman et al., 1994). However, in the case of dependent samples, this may reduce the accuracy of the estimate and in some cases present over optimistic estimations of model accuracy. Validation methods that do observe these dependencies include methods such as 'blocked hold-out' validation, whereby the training data always precedes the test data in a temporal sense, and the training and test observation dates to not overlap. This method is utilised in Section 7.4 when we compare the data-driven methodology with the existing physics based approach used at Network Rail.

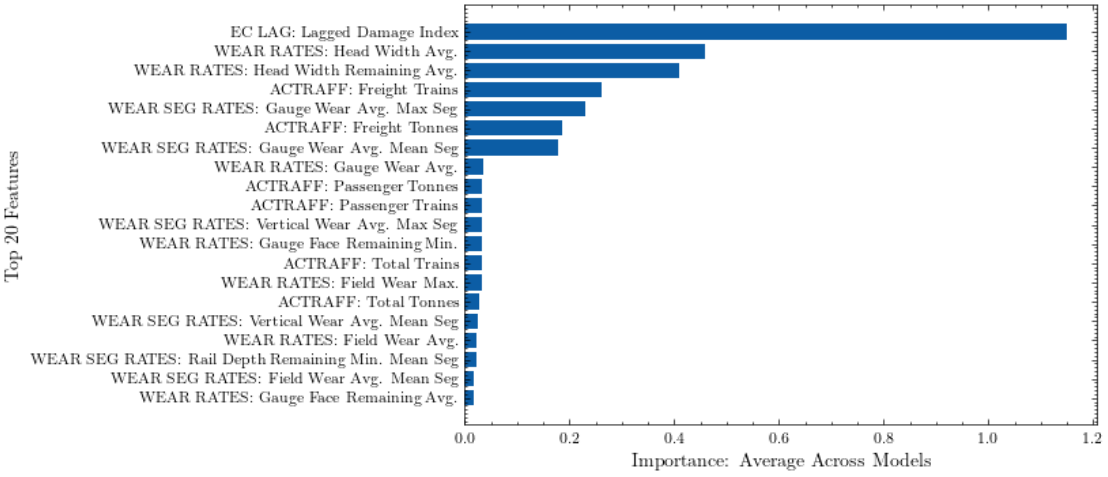


FIGURE 7.13: Permutation Feature Importance: Average Importance Across all Models

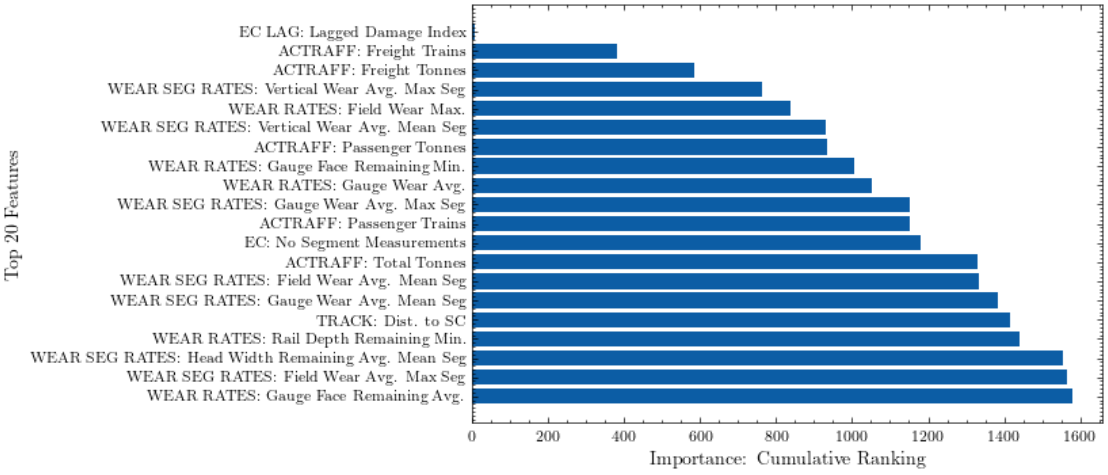


FIGURE 7.14: Permutation Feature Importance: Cumulative Rank Across all Models

7.4 Modelling with Track-Ex

As described in Section 3, Track-Ex is the NR implementation of the WLRM. The intention of this section is to provide a basic comparison of the approach for

estimating damage using Track-Ex with the proposed regression models and recorded EC measurements.

In order to generate a Track-Ex damage simulation which can be compared with the data-driven models presented in the previous sections there are some elements of both the data-driven and WLRM approaches that must be considered:

- **Forecast Horizon:** A Track-Ex RFA simulation uses a set of track geometry data to define the initial track characteristics over a chosen site at a specific point in time, an estimate of damage is determined based on these initial conditions, pre-calculated contact forces and the traffic over a given forecast horizon, whereas the data-driven approach includes damage measurements at the given recording dates. In this case, an estimation is determined over a 1 year (13 periods) horizon, and thus Track-Geometry files must be available at the beginning of the time-window, and Eddy Current data close to the end of the window in order to make the comparison. Track Geometry Data for TL1 is therefore collected from August 2018, to enable a prediction of damage to be made for August 2019 and compared with Eddy Current Data collected in that month. The Regression Model estimates must also be modified to reflect this set up. Thus far, regression models are evaluated using K-Fold Cross Validation, where 10 randomly generated 60-40 splits of the data are utilised to generate 10 different models and the mean average over all splits is presented. Instead when making future predictions similar to Track-Ex, it is important that the temporal ordering of observations is preserved when devising training and test splits. Therefore the data is split such that only data before August 2019 is used for training, and data in August 2019 for the test data.
- **Spatial Frequency:** The Track-Ex and Data-driven model approaches generate outputs on different spatial scales. For instance, the spatial frequency of Track-Ex is driven by the sampling frequency of the Track Geometry (TRGM) data (approximately 0.2m), whereas the ML models are presented at spatial frequencies of 11 yards.

Clearly the cases are not identical, and there will be some error, however it is the nearest approximation with the data available. The complete process for devising both sets of calculations is presented below and in Figure 7.15:

1. **Problem Specification:**
    - **Define Route:** TL1,
    - **Define Forecast Horizon:** 1 year (i.e. 13 periods) , prediction for August 2019,

2. **Calculate Damage along Route using Track-Ex (DI_{TE}):**

   (a) **Collect Data:**

   - **ACTRAFF:** Period 317 has been utilised for the estimation and the resulting damage is appropriately scaled to generate an annual result, Period 317 was selected since it is the last available period from the ACTRAFF data before the data would need estimating,
   - **TRGM:** Track Geometry data are collected from August 2018,
   - **Other Information:** The track summary data is utilised to define the stations, tunnels and line speeds,
   - **VDMs:** The Vehicle Damage Matrices (VDMs) are extracted from Visual Basic (VBA) implementation of Track-Ex (version 4.5),

   (b) **Make predictions:** RCF Damage predictions are estimated for a worst case scenario, i.e. assuming the rail is the high rail, note that the effects of braking and traction are not included in this implementation.

   - **Wear Condition:** Moderate,
   - **Wheel-Profile:** P8,
   - **Rail Type:** 113a,
   - **Steel Grade:** Standard 260,
   - **Lubrication:** 1,

   (c) Map Track-Ex damage to 11 yard segments: Track-Ex and Regression model are output at different spatial frequencies, the Track-ex outputs are at the frequency of the track geometry data whereas the regression model are output to 11 yards, in order to compare directly the Track-ex data would require mapping and aggregating to the 11 yard segments,

3. **Calculate Damage along Route using Regression Model ($DI2$):**

   (a) **Input Configuration:** Smoothing 5, Algorithms: [Linear Regression, Random Forest, SVR-Lin, SVR-Gaus, KNN, MLP] with optimised parameters as in Section 7.3.1,

   (b) **Data Preparation:**

   (c) **Model Training:** In this instance data are not randomly split into training and testing sets. Instead the test data is the damage data recorded on a selected test date and the training data is all data collected prior to this date. In this case the data is 'held -out', i.e. the model has not seen any of the data before. For this particular case the test date is selected as August 2019, The resulting training data size: 5770, test data size: 610.

   (d) **Model Testing:** Make predictions for test date (August 2019) using the trained model,

4. **Compare Model Outputs:**

    - **Scale Model Outputs:** The outputs of the models are of different orders of magnitude, the regression model is at the same scale as the input data, whereas the Track-Ex damage indicator is a non-dimensional quantity that represents relative damage across spatial locations. These may be normalised to enable closer comparison, DI_{TE} and $DI2$ are normalised using a standard scalar,
    - **Determine Comparison factors:** Methods for comparing the models are discussed in the next section.

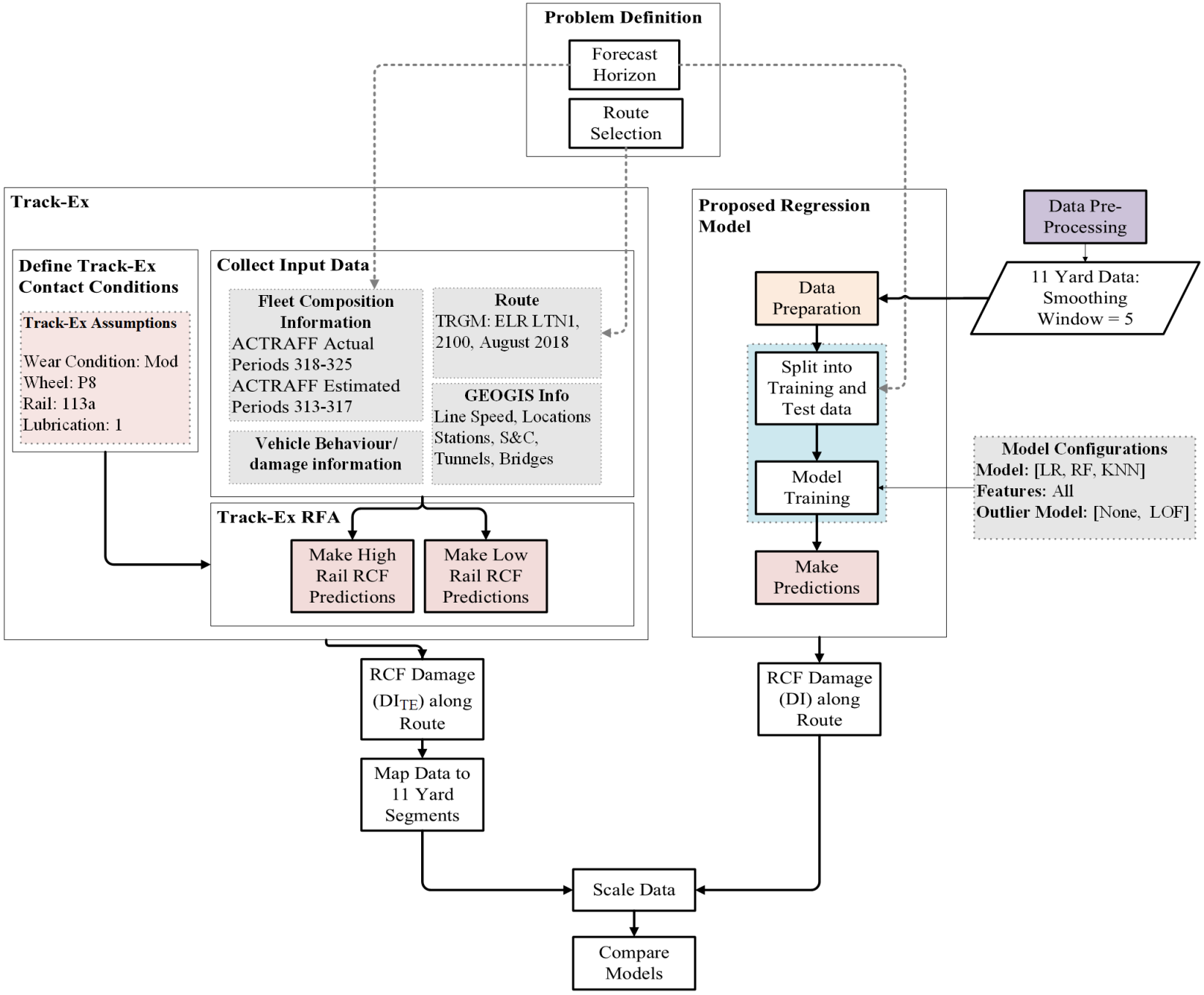


FIGURE 7.15: Comparison of Track-Ex Implementation and Proposed Models

7.4.1 Results

7.4.1.1 Results: WLRM Approach

Figure 7.16 illustrates the output of the Track-Ex simulations for the left and right rails and how these results compare with the EC measurements for August 2019. Note that the lower chart illustrates the curvature data utilised in both methods, clearly there are some regions where the TRGM data (Track-Ex input) is missing- these areas are highlighted in grey and all future comparisons will not include these regions.

Whilst on different scales, visually the measurement data and Track-Ex predictions appear to be in relatively good agreement, for instance the predictions for the left and right rail at approximately 30000 yards appear to be well aligned.

On the other hand there are instances of differences in locations of damage. For the left rail, there are 4 instances around 80000 yards where Track-Ex predicts damage to occur on the high rail (left rail) but this is not reflected in the EC data, a reason for this may be the presence of interventions such as rail replacements. Additionally at 60000 yards, Track-Ex does not predict the damage on the left or right rails, these rails are the lower rail of curves. Track-Ex estimates considerable forces at 45000 and 53000 yards on the right rail (this corresponds to the high rail at these locations) which are not indicated at the same levels in the EC data. At 45000 yards there is virtually no measured damage, and at 53000 yards there is damage detected but at a much lower levels than at 32000 yards (where Track-Ex predicts the opposite to occur). Track-Ex suggests that the highest levels of damage should be seen at approximately 53000 yards, whereas the measured data suggests that highest levels of damage are seen at 32000 yards.

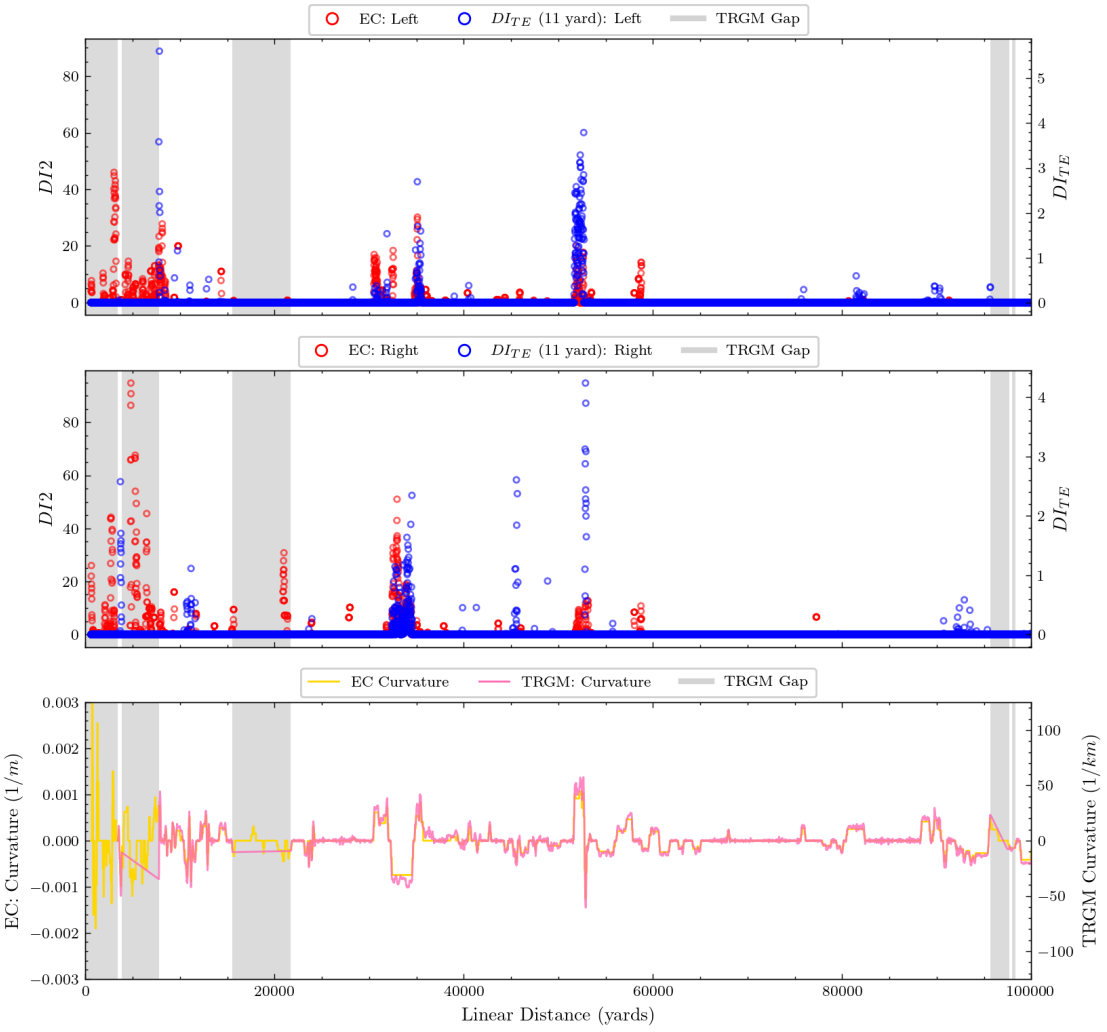


FIGURE 7.16: Track-Ex Predictions

7.4.1.2 Results: Data-Driven Approach

Table 7.8 presents the results for the optimised data-driven models, where the training data includes only EC observations prior to August 2019, and the predictions are calculated for August 2019.

The first observation from the results in Table 7.8 is the considerable reduction from the results presented in Section 7.3.1 and Table 7.6. The change is a result of the alteration in validation method, instead of randomly splitting the data into training and testing sets, the data are treated as sequential in order to provide a forecasting estimate. Therefore for training the model will only be trained on data recorded prior to August 2019 (since the test date is August 2019), all data from 2020, 2021, and 2022 is not utilised. The reason for choosing this date was the availability of TRGM data here and the ability to utilise ACTRAFF data prior to the pandemic where estimates vary considerably from pre-pandemic levels. Whilst in the case of random cross-validation the accuracy for KNN, MLP, SVR-Gaus and RF models were very

similar, in this case the Random Forest algorithm predicts RCF damage for August 2019 with the highest accuracy. The following section presents the analyses conducted for comparing the data-driven and WLRM based approaches with the observed results.

TABLE 7.8: Modelling Results: LOF 0, Smoothing 5, Input Features 7, Optimum Parameters, Hold Out Validation

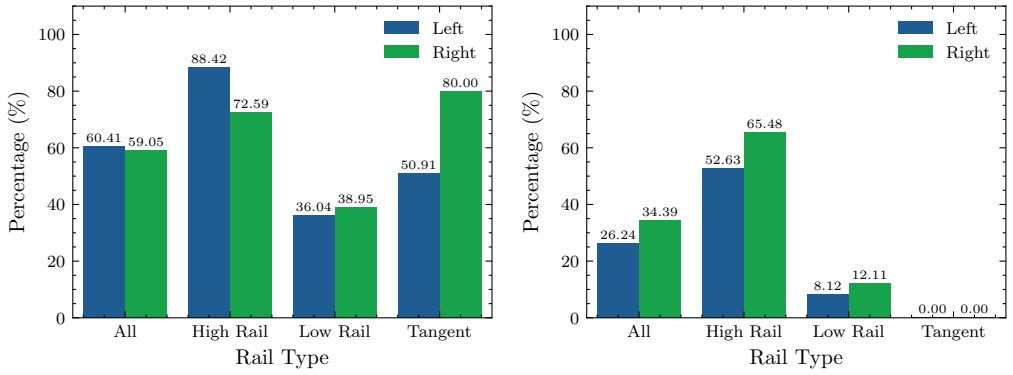
Index	Regression Model	Outlier Model	R^2	MAE	MSE	Run Time (s)
0	LR	None	0.597	0.54	0.749	10.8503647
1	RF	None	0.706	0.406	0.546	15.3321705
2	SVR-Lin	None	0.617	0.523	0.712	19.5808818
3	SVR-Gaussian	None	0.499	0.446	0.931	16.4730338
4	KNN 1	None	0.446	0.55	1.03	13.3388583
5	KNN 2	None	0.448	0.549	1.026	11.78977
6	MLP	None	0.544	0.504	0.848	20.2853095

7.4.2 Results: Comparison

Due to the nature of these two approaches, it is difficult to directly compare their outputs. Therefore below are some metrics that may help indicate their similarities, some similar metrics were used by Krishna et al. (2021). Here we denote Track-Ex estimations, regression model estimations and EC measurements as DI_{TE}, DI_{RM} and DI_{EC} respectively. The metrics are the percentage of damaging cases, false negative and false positives, and correlation of damaging cases.

7.4.2.1 Percentage Damaging Cases

Firstly we calculate the percentage of non-zero damage cases for each model to determine how well the models estimate where damage will occur. The results for Track-Ex and the Observed values are shown in Table 7.9, which contains the percentage of damaging cases alongside the absolute number of cases for Track-Ex and the observed damage. Note that this data includes only the track sections where Track-Ex has valid track geometry data (which is 442 11 yard segments). And further that the data-driven models are not included in the table since the damaging cases are identical to the Observed damaging cases, the data-driven models predict the location of damaging sections with 100% accuracy. Figure 7.17 illustrates the results shown in the Table 7.9. Notably Track-Ex appears to neglect half of the damaging cases compared with the measured data, from the area of track under study the observed data highlight 60% and 59% of sections to have some RCF damage for the left and right rail respectively, compared with 26% and 34.39% identified by Track-Ex. However, for the high rail the number of damaging cases are much closer between the



(A) EC Observed (B) Track-Ex

FIGURE 7.17: Percentage of Damaging Cases

observed data and Track-Ex. On the other hand, when looking at the low rail and tangent sections of track Track-Ex estimates so few cases of damage on the low rail and none at all on tangent track that the difference between the outcomes is significant.

TABLE 7.9: Comparison of Observed and Track-Ex Predicted Damaging Cases by Rail Type

	TE Damaging %	**Total No. Cases** $DI_{TE} > 0$	**Obs Damaging %**	**Total No. Cases** $DI_{EC} > 0$	**Total No. Sections**
Left Rail	26.24	116	60.40	267	442
Right Rail	34.38	152	59.04	261	442
High Rail L	52.63	100	88.42	168	190
High Rail R	65.48	129	72.6	143	197
Low Rail L	8.12	16	36.04	71	197
Low Rail R	12.1	23	38.94	74	190
Tangent L	0	0	50.9	29	55
Tangent R	0	0	80	44	55

7.4.2.2 False Positives and False Negatives

Whilst the percentage of non-zero cases are outlined in the previous section, this metric does not indicate how many of these cases were correctly predicted compared with the measured data. The quantities of False Positive (FP) and False Negative (FN) will determine where the observed and Track-Ex estimations overlap.

- **False Positives:** Where the observed value is zero but the model predicts non-zero damage,

$$FP = \frac{(D_{EC} = 0) \cap (D_{model} \neq 0)}{(D_{EC} = 0)} \tag{7.5}$$

- **False Negatives:** Where the observed value in non-zero but the model predicts zero damage,

$$FN = \frac{(D_{EC} \neq 0) \cap (D_{model} = 0)m}{(D_{EC} \neq 0)} \tag{7.6}$$

Table 7.10 indicates the percentage of FN, FP and the total number of damaging and non-damaging cases for reference (once more only the 442 sections where the TRGM data is valid are considered in the comparison). There are no cases of FN or FP for any of the regression models which is unsurprising since the data-driven approach relies on the values the algorithm has seen previously at each location. The data-driven approach would only generate a FN if the historic data did not present any damage at this location, and conversely would only result in a FP if there was no longer any damage at a location that has previously seen damage.

For the high rail, in the case of FP, where damage is observed to be 0, Track-Ex incorrectly predicts non-zero values for 37% and 50% of cases for the left and right rails respectively, however we note in this case that the total of cases where $DI_{EC} = 0$ is low (Left-22 cases, Right-54 cases). Further Figure 7.18 and Figure 7.19 illustrates the FPs and FNs along the track for the left and right rails, the plots also show where the rails have been replaced between 2014 and 2019. Whilst the rail replacements do not account for all of the false positives identified by Track-Ex, they do appear to overlap with some cases. It can be seen when observing more closely that there are a number of incidences on the right rail between 10000 and 12000 yards where a FP is indicated but this also coincides with an area which has undergone recent rail replacement which may explain the absence of RCF in the measured data. In the case of the low rail, as expected the FP are very low (Left- 1.5%, Right 12%) since Track-Ex predicts such a low number of damaging cases for the low rail, it reflects that out of all the cases where we observe damage to occur on the low rail, that Track-Ex indicates damaging cases for very few cases. Since Track-Ex does not predict damage for tangent track, there are no false positives.

False Negatives (FNs), on the other hand, are areas that could be further investigated to understand if there are elements driving RCF which Track-Ex does not consider since, FNs are the cases where, according to the Eddy Current data there are RCF cracks occurring in these segments, however Track-Ex does not predict damage here. For the high rail, the rate is 46.43% of cases for the left rail and 23.07% on the right rail. As expected, for the low rail, there is a very high percentage of false negatives due to the low incidence of RCF damage predicted by Track-Ex on the low rail. Similarly, for the tangent rail sections, once again since Track-Ex does not predict damage in these sections, none of the cases identified in the measured data are correctly predicted.

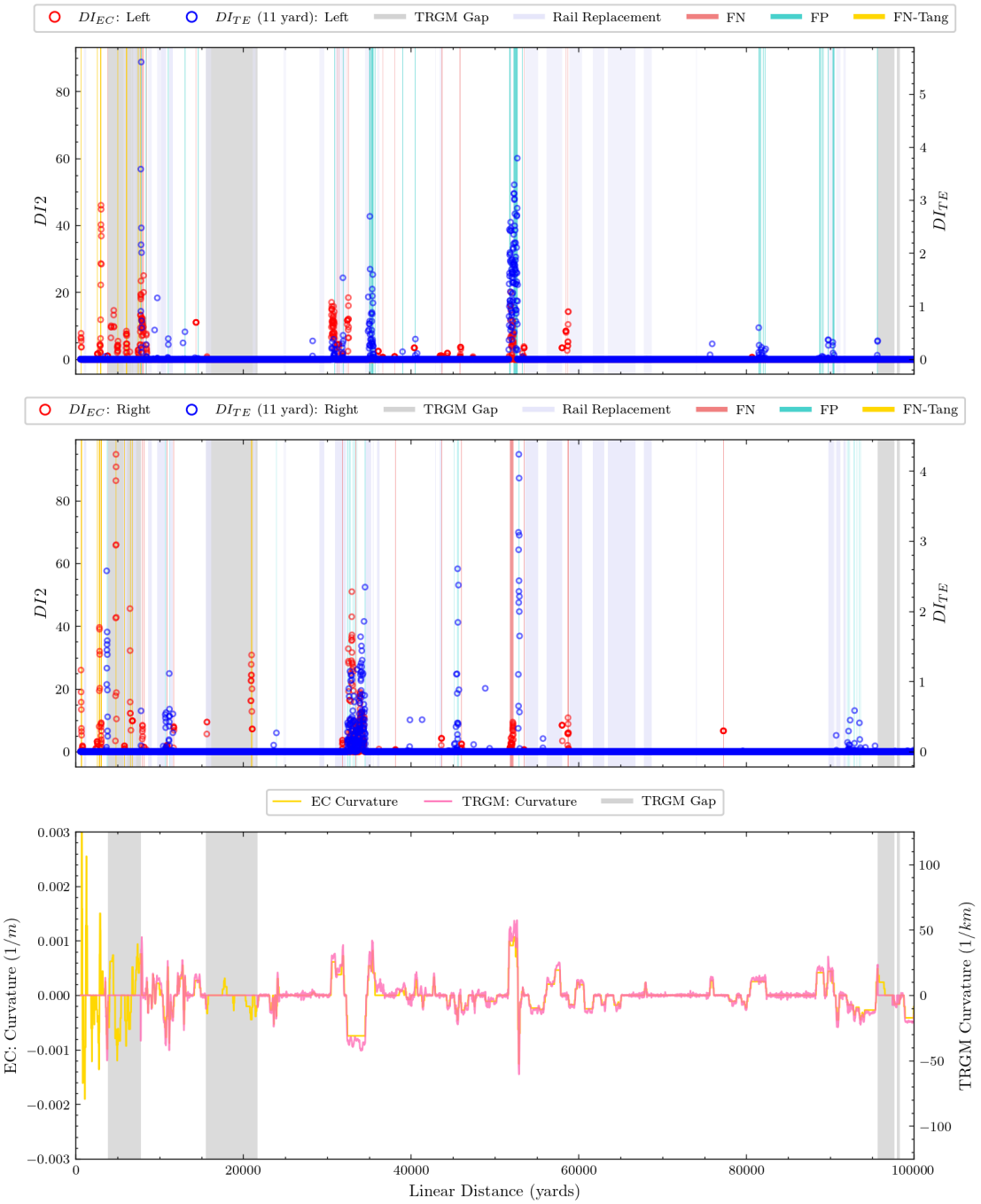


FIGURE 7.18: Track-Ex Predictions: False Negative and False Positives, 0- 100000 yards

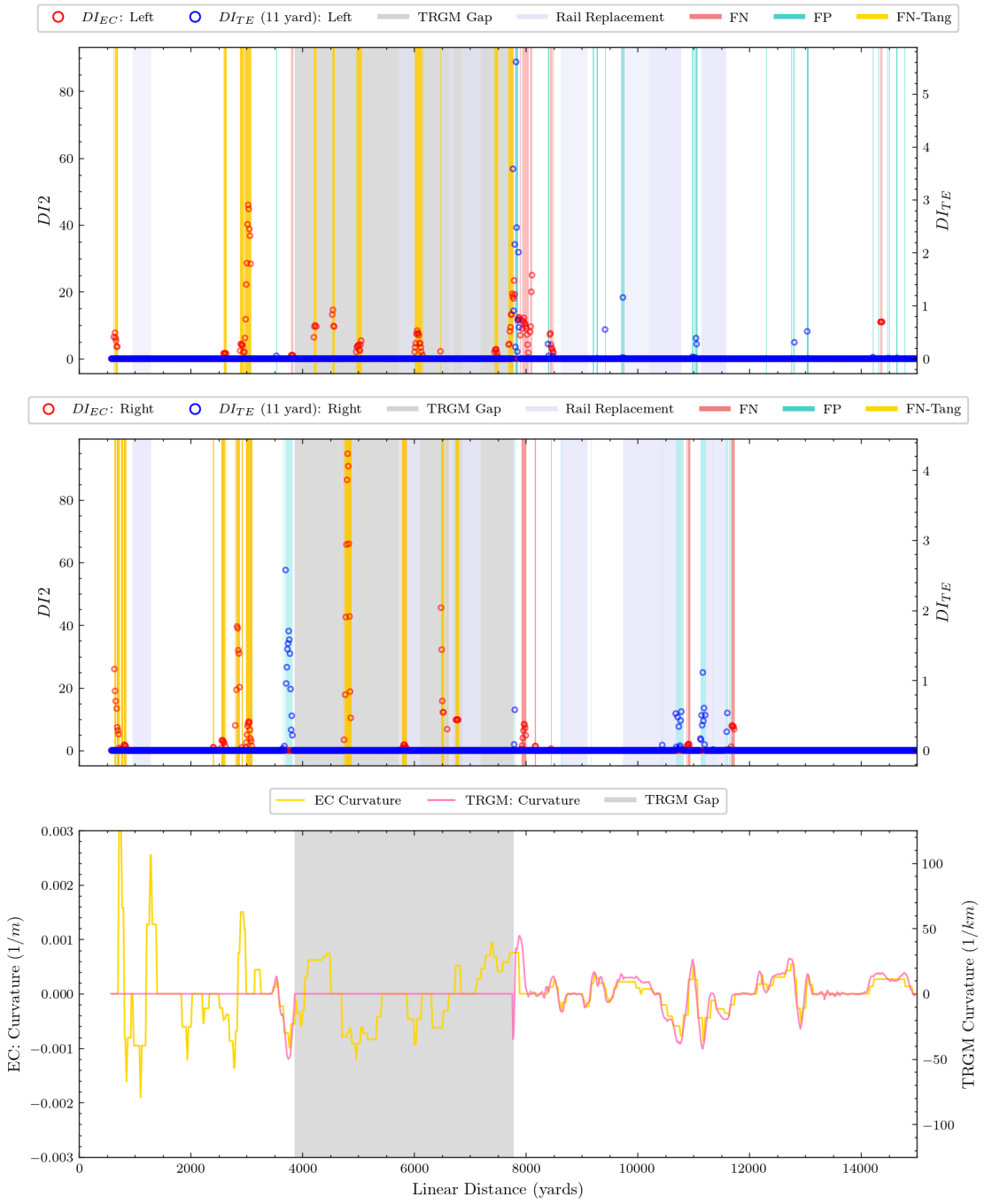


FIGURE 7.19: Track-Ex Predictions: False Negative and False Positives, 0- 15000 yards

TABLE 7.10: Quantity of False Negatives and False Positives Generated by Track-Ex Simulation Compared with Observed Data

	FN (%)	**FN:** $DI_{EC} \neq 0$	**FP (%)**	**FP:** $DI_{EC} = 0$
L All Rail	61.04	267	6.85	175
R All Rail	54.40	261	18.23	181
L High Rail	46.43	168	45.45	22
R High Rail	23.07	143	35.18	54
L Low Rail	80.28	71	1.59	126
R Low Rail	87.83	74	12.06	116
L Tangent Rail	100	28	0	27
R Tangent Rail	100	44	0	11

7.4.2.3 Correlation of Damaging Cases

A measure for joint variability of the estimated damage and actual damage is established using correlation analysis. Higher correlation denotes higher values from one model mainly corresponding with with high values of the observed.

$$\rho_{i-j} \longleftrightarrow \rho_{j-i} = \frac{cov((DI_i),(DI_j))}{\sigma_{DI_i}\sigma_{DI_j}} \tag{7.7}$$

where 'cov' denotes covariance function and σ denotes standard deviation. This value varies between -1 and 1, where a negative value implies an inverse relationship, 0 implies very low correlation (i.e. random behaviour) and a value of 1 suggests linear proportionality.

Figure 7.20 depicts the correlation between the ML models (LR, RF, SVR-Lin, SVR-G, kNN-1, kNN-2, MLP), Track-Ex Predictions (DI_{TE}) and the Observed data (DI_{EC}) for the Left and Right Hand Rails. Note that we compare only areas where Track-Ex has non-zero entries, since FN are addressed in the previous section, additionally we do not present Track-Ex correlation results for the low or tangent rail sections since Track-Ex generally predicts zero damage in these areas and thus the correlation is zero.

In general, the ML models present higher correlation for the RHR than the LHR such that all models are demonstrating a correlation > 0.93, whereas the LHR shows correlations > 0.71. Overall the Random Forest algorithm presents the highest overall correlation with the observed data (0.93 and 0.97 for the left and right rails). In contrast the Track-Ex predictions are very weakly positively correlated with the observed values for both rail sides.

However, it should be emphasised that the reported correlation coefficients are representative of the sample, denoted by r, the true population statistic is represented by ρ. Thus, we conduct a statistical hypothesis test to determine whether the

calculated relationships in the sample are statistically significant, or could happen by chance given the sample data. Our test the following hypotheses:

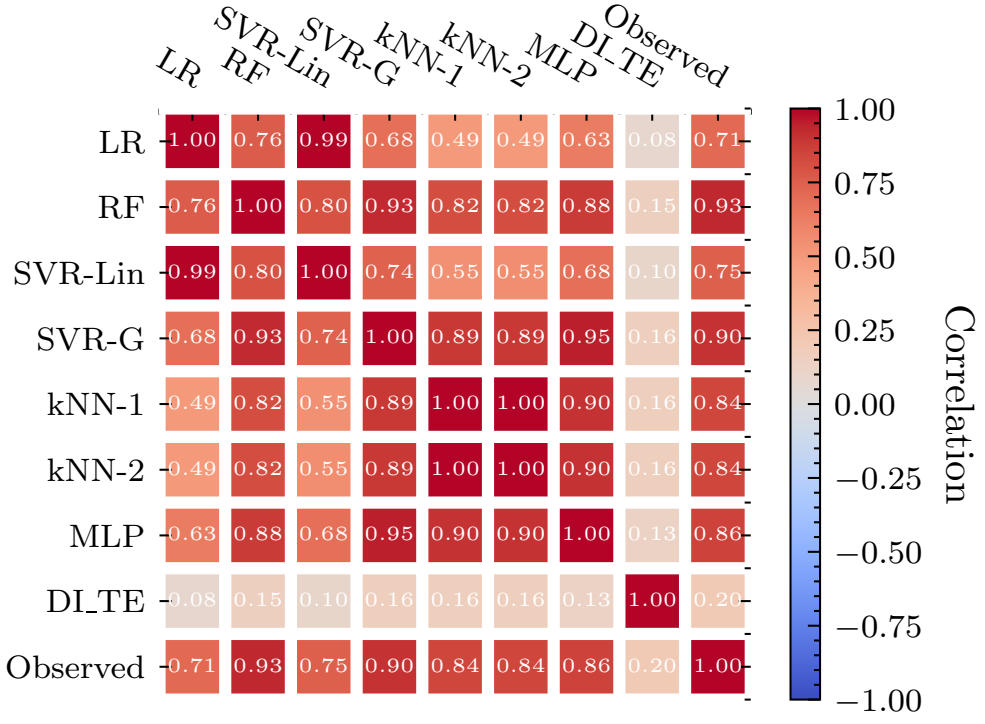
- Null Hypothesis: $H_0 : \rho = 0$ (i.e. the population correlation coefficient is not significantly different from zero, and therefore there is no linear relationship between the model predictions and the observed values),
- Alternate Hypothesis: $H_0 : \rho \neq 0$ (i.e. the population correlation coefficient is significantly different from zero, and therefore given the data we can conclude that there is sufficient evidence to support a linear relationship between the model predictions and observed values).

Table 7.11 summarises the calculated p-values, notably all p-values are significantly less than the typically utilised significance level of 0.05. Thus, there is sufficient evidence to conclude that there is a linear relationship between the ML model predictions and the observed data for the high rail because the correlation coefficient is significantly different from zero.

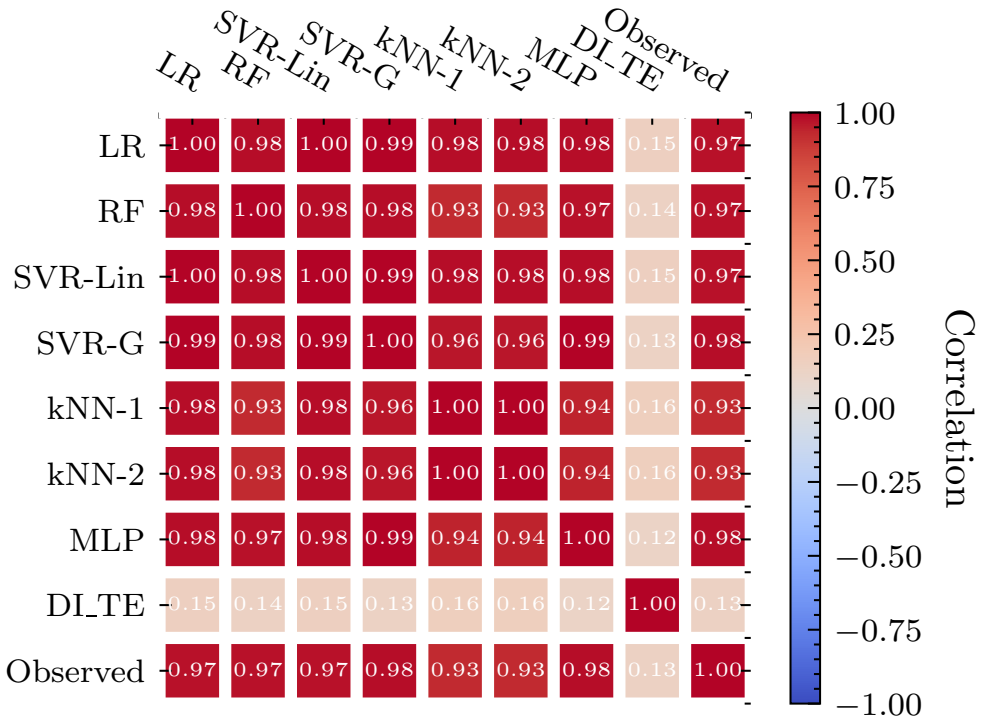
To summarise, the relationship between the Random Forest model *predicted* values (with optimal input parameters) and the *observed* damage values is much closer than for all other models, and the correlation coefficient is statistically significant. The random forest model tends to predict high and low values of damage well.

TABLE 7.11: Correlation P-Values

Model	Track-Ex: Left Rail	Observed: Left Rail	Track-Ex: Right Rail	Observed: Right Rail
LR	0	0	0.00000195	0
RF	0	0	0	0
SVR Lin	0	0	0.00000006	0
SVR Gaus	0	0	0	0
KNN1	0	0	0	0
KNN2	0	0	0	0
MLP	0	0	0	0
Track-Ex Prediction	NA	1.00E-08	NA	0



(A) Left Rail



(B) Right Rail

FIGURE 7.20: Correlation Coefficients (r) of Damaging Cases: High Rail

7.5 Concluding Remarks

This chapter is introduced by formally framing the regression problem which aims to devise a mapping which relates a set of features with a RCF damage parameter derived from Eddy Current measurements. We present a methodology employed to combine the pre-processed data generated in Chapter 5, and specifically prepared for this learning problem, with the use of machine learning algorithms to extract the complex relationships in the data.

Numerous model configurations are proposed and tested to determine optimal model settings such as the input feature set and the level of spatial smoothing applied to the input data. From this process the smoothing level selected is a window size of 5, and the Input Feature Set 7, which are subsequently utilised for further modelling to determine optimal algorithm hyper-parameters. The models were validated using k-fold cross validation such that the data were randomly divided into 10 different training-test subsets and the average performance over all of the 10 folds is presented, in order to reduce the affect of over-fitting. Over-fitting is a common issue presented when using highly flexible algorithms such as decision trees and neural networks, but by assessing the accuracy of the model on different subsets of the data, the variability of the final model accuracy is reduced, which has been illustrated in Figure 7.5. Using this approach RF, SVR-Gaus, KNN and MLP algorithms produced models with highest accuracy (> 0.85). However, it should be highlighted, due to the size of the data, when optimising parameters such as the smoothing window and algorithm hyper-parameters, the same set of training and test data were utilised as when building and determining the final accuracy of the ML models. Ideally, a third, validation set should be used once parameters have been optimised to determine the true generalisation performance. In this case therefore, there is the possibility that the models are over-fitted to the data, and the accuracy does not truly represent their ability to generalise well.

Analysis of model residuals has also highlighted some potential violations of commonly used statistical assumptions such as multicollinearity of the input features and heteroscedasticity in the model errors. These characteristics raise some questions into the statistical validity of using models which are based on linear regression, particularly for making inferences and determining model characteristics such as error bounds.

Further, despite the remarkable function-fitting capabilities demonstrated using machine learning techniques, their opacity in explaining the basis of their predictions still remains an issue. Feature importances are therefore generated across all models to understand the driving features for predicting RCF damage. The highest ranking features highlighted in this procedure are the lagged damage indicator, the traffic features and the wear features, which correlates well with what we understand about

RCF generation and propagation. Instinctively, previously measured RCF damage will clearly influence the current RCF damage in a particular location, and its propagation will rely heavily on the accumulated traffic and the wear characteristics of the track. However, the significant dependence of the models on lagged damage strongly indicates the presence of autocorrelation in the data. Autocorrelation implies that instances are not independent, and therefore presents issues when using methods such as ordinary least squares. Additionally, whilst many machine learning methods do not rely on the independence assumption during training, random k-fold cross validation does make this assumption. We must therefore bare this in mind when considering the resulting model accuracies generated using random cross validation. Alternatively, validation methods that observe these dependencies such a 'blocked hold-out' validation can account for autocorrelation, and are discussed in the second part of the chapter when a comparison with the physics based methodology is presented.

The second part of this chapter demonstrates a different approach in the modelling methodology. Rather than randomly splitting the data into training and test splits, the temporal structure of the data was observed such that the timestamps for data contained in the training set and test sets do not overlap. For instance, observations collected in 2015, 2016, 2017 and 2018 are selected for training, and observations collected in 2019 are retained (or 'held-out') for testing. This enables a true future estimate of RCF damage to be obtained and accounts for autocorrelation, and additionally enables a comparison of the data-driven technique with the physics based approach implemented in Network Rail software-tool Track-Ex. Input parameters and model hyper-parameters are retained from the first part of this chapter and the results are presented for each of the ML algorithms, these results are then subsequently compared with a Track-Ex simulation which has been devised to match the observational circumstances as closely as possible. Since it is difficult to directly compare these methods a number of metrics are analysed to provide an insight into how these methods compare.

Firstly, it is established that the data-driven approaches correctly identify all the locations where RCF is detected in the observed data-set, i.e. the number of damaging cases is identical, and there are no instances of false negatives (where the observed data indicates a non-zero case but the model estimates zero damage) or false positives (where the observed data indicates a non-damaging case but the model predicts damage at this location). When we compare Track-Ex with the observed data, Track-Ex only predicts approximately half the number of total damaging cases as compared with the measured data for both rail sides. However it is estimated that many of these cases that are not identified correspond to low rail and tangent rail sections, where Track-Ex notably predicts extremely low instances of damage or none at all for tangent sections. Additionally there are many instances of False Negatives

and False Positives, i.e. the damaging and non-damaging cases do not correctly overlap with the measured data. Due to the much lower percentage of damaging cases identified by Track-Ex, the high False Negative values are plausible, in particular with respect to the low rail and tangent rail sections. Track-Ex incorrectly predicts no damage for 80% and 87% of cases for the left and right lower rails compared with the observed data, and does not predict any cases of damage for tangent track. In the case of False Positives, some cases may be explained by the presence of rail replacements which result in the observed damage falling to zero where a replacement has occurred. Further the FN and FP may also be influenced by spatial misalignments between the Track-Ex simulations and the EC data. It should be noted that it is well recognised that Track-Ex tends to under-predict low rail RCF damage (Burstow, 2004; Boyacioglu et al., 2018; Bevan, 2020). High rail damage is driven by the direction of creep forces in the traction direction, which enables fluid entrapment in minor cracks and drives these deeper into the rail, resulting in higher growth rates and risk of transverse defects (Burstow, 2004). On the other hand, it has been observed that low rail RCF cracks are driven predominantly by lateral forces, and the longitudinal (traction) forces are small. Since Track-Ex requires a positive longitudinal shear force in order to trigger the damage calculation (such as in the case of high rail damage), these cases are likely to indicate no RCF damage.

Finally, the correlation of damaging cases is analysed for each of the data-driven and Track-Ex models compared with the observed data. The predictions generated by the random forest algorithm present strong positive correlations with the observed damage, with correlation coefficients of 0.93 and 0.97 for the left and right rails, whilst other data-driven models correlate well on the whole (i.e. all correlation coefficients $>$ 0.7). Comparatively the Track-Ex model predictions demonstrate weak positive correlation with the observed values (0.2 and 0.13). However, again it should be noted that there are some discrepancies in the comparison. Track-Ex is not intended to predict the length/ depth of cracks, rather where they are likely to initiate due to high contact forces, whereas the regression models base their predictions on the historic locations and magnitudes of damage.

Thus to conclude, whilst we understand that there are a number of practical reasons that prevent engineers from using Track-Ex to plan where to correctively grind or mill we have also presented some quantitative evidence to suggest why we might choose not to use Track-Ex in this instance. Data-driven models on the other hand have presented a robust method of indicating with 100% accuracy exactly *where* the damaging and non-damaging cases are located, and additionally they indicate a good correlation with the *intensity* of damage that should be expected. Therefore, combining these techniques with the approaches currently in use; particularly for a corrective strategy, engineers can highlight the areas forecast to have high damage using the proposed model.

Chapter 8

Summary

8.1 Conclusions

In summary, this thesis has provided a body of research which seeks to formulate and solve the problem of predicting rail damage due to Rolling Contact Fatigue using a data-driven methodology. The approach aims to use the numerous data sources at Network Rail more effectively and develop a strategy which could be integrated with planning of maintenance tasks such as grinding and milling, with the objective of a better targetting of resources.

The major difficulties which have arisen when embarking on such a task are the wealth of data-driven techniques available and the utilisation of real data sources compared with simulation data. Real Eddy Current data are plagued by noise and complexities in their structure which have driven the direction of modelling in this study. A model driven by data is only as good as its inputs, thus the data processing task has been central to this study, without which regression modelling or analysis of the data directly were unlikely to produce useful or reliable results. Additionally the data demonstrates some complexities such as the presence of space and time, sampling irregularity and sparsity which restrict the ability to apply certain techniques that rely on evenly gridded, high frequency data.

This study has presented an approach that considers these difficulties, and utilises optimisation and machine learning algorithms to explore, process, integrate and formulate the data for the learning task. We then provide two primary outputs; 1) a number of simple approaches that could be applied directly to assist the preventive and corrective strategies already in use at Network Rail (Chapter 6), and 2) a regression model based on machine learning algorithms to forecast damage, which could be used in a variety of planning tasks. This method has additionally been compared with a well-known physics based engineering model currently used by Network Rail- the WLRM (Chapter 7).

In order to achieve these objectives the initial sub research questions outlined in Chapter 1 were considered, and these are summarised below:

1. *What are the critical drivers for RCF initiation and degradation?* To understand the key parameters necessary to build a RCF prediction model, the underlying physics driving crack initiation and propagation were considered. It is well understood in the domain that cracks initiate on rails due to fatigue, whereby the wheel/rail contact experiences numerous wheel passes which surpass the fracture point of the rail material eventually resulting in the formation of minor cracks. Additionally wear plays a crucial role in managing RCF, since low levels of wear act to remove minor cracking and prevent cracks from reaching a critical size needed for branching and propagation. Thus the crucial parameters that affect RCF are determined to be track curvature, super-elevation, line speed, accumulated traffic, rail mettalurgy, rail profile (i.e. head, side wear), and presence of defects,

2. *How are RCF cracks detected and monitored?* Traditionally ultrasonic testing methods have been used to detect internal defects in the rail, however these techniques perform poorly at detecting small surface cracks $<$ 4mm. Eddy Current technology on the other hand has the ability to detect very light cracking at the surface of the rail such as RCF damage. The method has been pilotted since 2015 on the UK network but have not as yet been analysed in any great detail. The Eddy Current data for plain line track were thus selected to drive data-driven prediction of RCF damage.

3. *What are the current grinding practices in use by Network Rail?* The two leading strategies used by Network Rail are in-traffic preventive grinding and corrective grinding. Preventive grinding is performed cyclically based on the track curvature and tonnage, whereas corrective grinding is a risk-based activity which uses the recent Eddy Current data, and a number of additional sources that have been culminated in a risk framework based on expert judgement. Notably neither of these strategies capitalise on the availability of the WLRM model to inform their strategies, nor do they use information of historical rates of cracking. Thus, a methodology based on real data may be beneficial.

4. *What existing models are available for RCF evolution modelling?* There are many studies which cover RCF modelling, and in Chapter 3 we review the engineering models that couple multi body simulations to generate different contact conditions, and empirical methods to estimate a rail damage indicator. The WLRM is a well-known technique used by Network Rail and is implemented within a software tool known as Track-Ex where it is possible to determine RCF risk for large sections of track based on the anticipated future traffic. As compared with more complex and higher fidelity methods, its accuracy may be

lower however it has the ability to predict damage on large sections of the rail, which is a necessity if it is to be used alongside strategies and everyday maintenance use. However, it is noted that within Network Rail this model is in fact not generally used by maintenance teams to impact their decisions on grinding and milling, this is likely to be due to a number of reasons. For instance, the outputs of the models do not present information such as current conditions of the rail, the model assumes perfect initial conditions, and assumes a linear damage accumulation rather than using measured rates. Additionally the software can be difficult to use and requires the gathering of numerous data sources for its inputs, and can only generate results for a portion of the track at a time. To do this numerous times for different track lengths, and additionally gather supporting data regarding the current condition of the rail is a highly manual and time-consuming process from a practical perspective. In the literature review, as far as we are aware there are no studies that investigate the use Eddy Current data, and machine learning models to drive a RCF prediction model. Whilst there are some related studies which look in particular at squats, and many papers on applications of machine learning for video and track geometry data, this is not the case for modelling of head checks, gauge corner cracking which are detected using EC and Ultrasonics,

5. *How can the available RCF condition data be formulated for regression analysis?* Preliminary Analyses of the Eddy Current data reveals some of the complex characteristics of the data, such as the presence of space and time, temporal and spatial sampling irregularity, data sparsity and misalignments. Additionally the temporal resolution and quality of the sources of interventions data are not suitable for deriving useful information. It is therefore determined that the data must be cleaned, and filtered to generate a data set which represents the degradation of the rail without the presence of interventions. Moreover additional features must be generated to capture the spatial and temporal features, and different data sources must be combined to leverage some of the key RCF drivers stemming from the literature review.

6. *What insights can be gained from the RCF condition data?* Once the data has been pre-processed, aligned and integrated with other data sources, Chapter 6 generates a number of KPIs which provide an indication of the historic damage over each 11 yard segment. These KPIs are used in two simple approaches, firstly the KPIs are used to divide track segments into distinct groups representing track curvature and low, moderate and high levels of damage. These categories may be used to indicate where interventions strategies may need revising. Secondly an approach for indicating where RCF hotspots are located on the track based on these KPIs, and ranking these is presented. The two methods indicate how simple approaches can be useful once the data is in a cleaned and structured format.

7. *Can regression analysis be used to build an effective RCF regression model using machine learning algorithms?* Several machine learning algorithms are trained using an array of input features, input data sets (with different levels of spatial smoothing applied), and algorithm parameters. The models are tested across 10 different folds on the input data to determine their sensitivity to different training data and generalisation capability. Low levels of smoothing (i.e. a window size of 5- i.e. 55 yard segments), and a feature set which includes lagged damage indicators, wear, traffic, spatial characteristics (such as curvature, cant, speed), distance to track features (such as stations and S&C) and defect parameters result in a model with a high (> 0.8) average r^2 score.

8. *How does a data-driven approach compare with existing physic-based models?* A Track-Ex simulation is performed to generate estimations of rail damage over a year from August 2018 to August 2019. Additionally the proposed data-driven approach is modified such that the validation method observes the temporal ordering of the data such that a true and comparable forecast can be generated. The observed data, track-ex predictions and ML predictions are presented, and a number of metrics are presented to indicate their similarities. In general, the ML methods generate much closer predictions to reality with respect to correctly identifying the location of damaging cases, and additionally the general correlation in magnitude of these estimated damaging cases. Track-Ex in particular suffers with low prediction accuracy for low rail and tangent track. However it is noted that the comparison is not exact, and may suffer from some bias, for instance the Track-Ex predictions are for a specific simulation case which assumes a specific wheel profile, level of lubrication, etc, and there is no simulation of the braking and traction experienced at stations. There is also the issue of mis-alignment, Track-Ex predictions rely on TRGM data for position, whereas the ML models and observed data are aligned precisely with each other, but may be misaligned from the track geometry data. Further, it is noted that Track-Ex has never been intended to predict crack depth or length, rather an indication of the magnitude of the contact forces needed to drive RCF damage.

In an overall conclusion, the work sought to answer the question: *Can data-driven methods be used to build an effective data-driven RCF prediction model to help inform grinding strategies?*

This thesis has shown principally that the data processing task is critical in uncovering patterns in this type of noisy, spatio-temporal data such that it can be used for subsequent extraction of useful information. In particular, a routine for consistently aligning Eddy Current data in space is presented using partitioning, cross-correlation and optimisation techniques in order to address the low frequency and irregular nature of the EC data. Once the data is processed, simple analytic techniques

accounting for degradation patterns can be used to directly augment some of the existing preventive and corrective strategies. Two such suggestions are presented, firstly tangent and curved sections of track are analysed and categorised based on their historic Eddy Current measurements. These new categories are aimed to indicate where interventions strategies, such as preventive grinding, application of lubrication and friction modifiers, and identification of areas for high performance rail, may be adapted in future. Additionally, the processed data may be incorporated into a methodology to identify and rank current RCF hotspots to assist in managing the risk of these sites.

Secondly, this thesis has shown that a regression model can be built based on machine learning algorithms to determine damage due to RCF with relatively high accuracy. However, the performance of the model relies on extensive pre-processing and feature engineering to remove noise and capture the underlying phenomena. The ability of machine learning algorithms to map highly complex relationships between inputs and outputs is demonstrated, but it is also important to understand their limitations. Although these methods can indicate the relative importance of the input features, unlike physics-based methods, most cannot be used as inferential models to understand *why* a phenomena is occurring. An exception of course is Linear Regression, a statistically rooted distributional based method, where a function is estimated to represent the phenomena, and coefficient values indicate the contribution of each feature. Nevertheless, such distributional methods make rather stringent assumptions of the data under study, which, if violated can undermine their validity. For instance, further analysis has highlighted issues such as autocorrelation (non-independent instances), multicollinearity (feature dependencies) and heteroscedasticity (non-constant errors). Whilst most machine learning methods do not impose these same assumptions, and the presence of these features may not necessarily impact their ability to fit a suitable function to the data, there can be other issues arising due to these characteristics. For instance, if autocorrelation exists in the data, using random cross validation to estimate generalisation error may not be accurate. Instead, validation methods that observed the dependencies in the data (i.e. the temporal structure) can be used to combat this and are demonstrated when comparing the data-driven methodology with the physics-based approach using Track-Ex. Additionally, other methods in future could be used to account for some of these characteristics. For instance, rather than prediction of the absolute damage, prediction of the rate of change of damage can remove the effect of autocorrelation, a technique known as differencing, and feature reduction techniques could also be applied to reduce the effects of multicollinearity.

Additionally understanding the bounds of the model is critical, the purely data-driven models are limited by the data they are trained on, for example, if the model has not seen an instance with damage index > 100 before (i.e. the variability in the data has

changed, the data is non-stationary), then it will not be able to predict this either, and therefore may tend to under-predict in this case. Yet, these models can capture high levels of non-linear complexities, can generate predictions on the scales at which the data is measured, and be continually updated using real -time information. The generation of the modelling data sets further integrates numerous sources to a common reference frame which is highly useful to a track engineer. To better understand risk and degradation, predict specific outputs and highlight patterns within the data (rather than providing an explanatory model), these techniques and models can be highly advantageous to analysts and railway engineers.

We next present our contributions to the field of research, followed by the recommendations for future work and direction.

8.2 Contributions

The thesis contributions to this field (in the order they are introduced) include:

- A robust approach for consistently aligning Eddy Current data is proposed that addresses the characteristics of low frequency and sparse data. The approach may be used to enable reliable analyses of degradation rates and perform regression modelling (Chapter 5),
- The development of a methodology for cleaning and integrating numerous real data sources related to the generation of RCF defects on the UK rail network. The approach proposed a RCF damage parameters based on EC data, and combats undesirable qualities such as noise and different data scales in order to formulate a data set that can be learned from using data-driven and specifically machine learning techniques (Chapter 5),
- Demonstration of an approach for repeatedly analysing the processed modelling data set to better understand RCF degradation rates and augment the existing Network Rail preventive and corrective grinding strategies. New KPIs are proposed to determine the estimated amount of RCF damage in a specific track segment. These KPIs can be used for maintenance decision making by infrastructure managers (Chapter 6),
- This thesis presents a thorough comparison of the development and training of regression models using different machine learning algorithms. Further we propose the most suitable model for the application of predicting damage over a length of track on the UK rail network that may be integrated with existing maintenance strategies (Chapter 7),

- The proposed data-driven methodology is formulated as a forecasting problem and compared with outputs generated by the Whole Life Rail Model (WLRM). Track-Ex a software tool which implements a simplified version of the WLRM, using look-up tables generated by multi body simulations. An approach is described which enables a basic comparison between the observed data, Track-Ex predictions and the ML predictions. Overall, the analysis indicates that for forecasting damage, the proposed approach based on numerous shallow decision trees presents improved correlation with the observed data as compared with the WLRM approach (Chapter 7).

In summary, probably the most important contribution of this thesis is that, given suitably pre-processed data, modern data-driven modelling can predict RCF at least as well as, and often better than, the current physics-based approach used by Network Rail and can effectively augment the practices of rail maintenance teams.

8.3 Recommendations and Further Work

Here we present recommendations in the areas of data collection, practical application of the methods applied in this study and some areas of further modelling and analysis that could be conducted by industry or researchers.

- **Data Collection:** In future, if operators such as Network Rail intend to use their data more effectively, it is critical that data collection and in particular data quality is considered more carefully. Some specific observations include:
  - Improve the timing of data collection: in order to assess the effectiveness of interventions, there needs to be Eddy Current measurements taken before and after an intervention such as grinding or milling.
  - Improve the quality of collected data: the grinding data would ideally contain more information than simply where and when the grinding has taken place, it would be desirable to have information on the machine number and type, the grinding profile and pattern, the speed, the number of passes etc. Further the grinding and milling records are generally collected by the contractor, e.g. Speno or Loram, and are passed to Network Rail in the form of individual pdf files, this data contains useful information which should be contained in a relational database for any future work regarding analysis of defects and maintenance data,
  - It would considerably speed up analysis if wear data were contained within a relational database, for the purposed of this study it was necessary to locate within numerous csv and Excel files the relevant ELR and TID,

and then extract these manually and input into a personal relational database in order to integrate with EC data

- Accurate Traffic Data is critical for this application, a replacement must be be found for ACTRAFF for future work,
- Resolution of the rail replacement data: currently the year of replacement is available, this type of analysis requires a date or at least month of replacement,
- More accurate track characteristic data, such as track type, track curvature and location of S&C and stations,

- **Practical Applications:** It is recommended that some of the methods proposed in this thesis are applied on some selected lengths of track in collaboration with the routes, for example in the Anglian region:
  - **Data Pre-processing:** Data for the selected routes should be extracted and the pre-processing methods demonstrated in Chapter 5 should be applied to each of these. Subsequently an analysis should be conducted to determine whether parameters of the process require adjusting, in particular with regard to the signal alignment process, this can be verified through visual analysis of the resulting data and some basic statistical analysis indicating the performance of the alignment (statistics such as the Pearsons cross correlation coefficient)),
  - **Preventive Application:** Following the applied pre-processing of the data the KPIs proposed in Section 6.2 should be calculated to indicate which curved and tangent track section may require further or less interventions. These results can be verified with track experts, and then further proposals for intervention strategy modification could be suggested. For example, it could assist in targetting candidate areas for high performance rail or improved lubrication,
  - **Corrective Application:** Additionally, the procedure proposed in Section 6.3 should also be trialled for these track lengths. Initially, the parameters detailed in this study should be used, however once the results are presented these should be analysed with route experts. The parameters should be subsequently fine tuned, and other data incorporated such as wear and traffic data. The results of this process should also be directly compared with the proposals that would otherwise be generated using the existing Anglia Risk Assessment procedure for Severe and Very Severe RCF sites,

- **Regression Modelling:** In future, the techniques presented in Chapter 7 could be used to develop a RCF prediction model using the machine learning algorithms proposed here. Further areas of investigation include:

  - **Validation:** The resulting models could be further validated using the historic data especially as the data set grows and enables for a more conclusive validation. The model should also be tested on additional track lengths. Unfortunately due to time constraints, and difficultly in obtaining all of the relevant data sources, the analysis was only performed for a single track length. Ideally in future the analysis would be repeated for a secondary route of similar criticality and sufficient quantity of Eddy Current data. The results could then be compared and the approach validated for other sections of track,
  - **Modification of target variable:** As explained in the conclusions, additionally methods such as differencing can be applied to the data, so as to predict the *rate of degradation* rather than the absolute damage,
  - **Reformulate as a Classification Problem for Rate Prediction:** To extend the regression model and provide an output that may be in a more useful format for track engineers, the learning task could be reformulated to a classification problem. Instead of predicting the RCF damage index, the target variable could be reconstructed such that a rate category is predicted (as stated in the previous point), all the RCF damage indices may be converted to a rate of change and then categories such as high, moderate and low for rate of degradation may be derived.

    The engineer could then see, given a forecasting window, which segments of track are likely to degrade faster, and should therefore be targetted first. This approach can be used in conjunction with the techniques shown in Chapter 6 to highlight and prioritise hot spots,
  - **Feature Reduction:** We propose also that further work is done to look at feature reduction techniques to reduce the effects of multicollinearity.

- **Other Analyses and Modelling:** Here we present the possibilities for further development of the work outlined in this thesis:

  - **Segment Based Model:** Based on the pre-processed modelling data set, can we use a decision tree/ classification algorithm to predict segment summary data (such as the KPIs presented in Chapter 6). For instance, then we could determine if the average rate of degradation of a track segment can be determined from its input features and if this model could be applied to a completely different section of track. Could an approach be devised such that we know which parameters drive the average segment rate of track deterioration?
  - **Data-driven Wear Model:** A similar data driven regression model could be developed for the wear data, particularly since the data is much higher frequency in spatial and temporal dimensions, and therefore extracting

degradation rates may be more straightforward. However, the issues of poor intervention data still remain.

– **Analysis of EC Probe data:** A distributional analysis could be performed on the individual probe data to understand the relationship between the location of the damage on the rail head, and the other track features.

Appendix A

RCF Physics

A.1 Rolling Contact Fatigue

To comprehend deterioration phenomena such as RCF and wear, an understanding of the forces generated at the wheel-rail interface is required. Contact forces at the WRI are driven by the static load of the vehicle sitting on the track, alongside the dynamic interaction between the track and vehicle as the train accelerates, brakes and steers during operation. These forces are divided into longitudinal forces, and lateral forces which are transmitted through the small area of contact known as the contact patch. This section provides an introduction into the physics of rolling contact, and the frictional forces present in a real system and how they are generated.

A.1.1 Principle of Rolling Contact/ Creep Forces

As a rail vehicle rolls over the rail, the motion may be described as pure rolling provided there is no slippage, or sliding at the point of contact between the two bodies. The resistance force due to friction is attributed to the sliding motion and it is absent in pure rolling motion, however in reality the motion is a combination of both rolling and sliding. The cases of pure rolling and rolling-sliding motion are illustrated in Figure A.1:

- **Pure Rolling Motion:** Here, the angular velocity ω, should equal v_0/r_o, where v_0 is the sliding velocity and r_0 is the radius of the wheel. The rotational velocity component counterbalances the one from sliding at the point of contact, the contact point is thus at rest.
- **Rolling-Sliding Motion:** In the event of some over or under-speed, ω', where $\omega = v_0/r_0 + \omega'$, the point of contact is no longer at rest. A relative motion now exists between the contacting surfaces, known as a *rolling-sliding* motion.

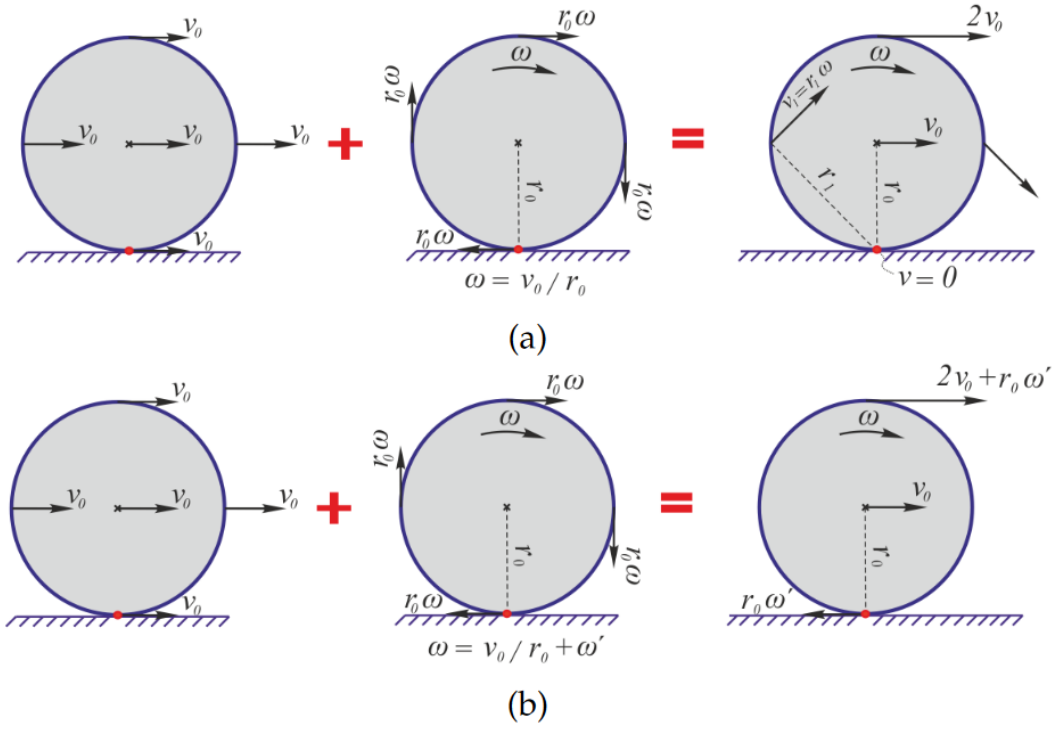


FIGURE A.1: a) Pure Rolling b) rolling-sliding motion of a rigid wheel (Shahzamanian Sichani, 2016)

In the second case, of rolling-sliding, there is a relative motion between the wheel and rail which results in friction which doesn't occur in the case of pure rolling. The resulting frictional forces generated in the contact patch due to micro-slippage, are known as *creep forces*, and act in longitudinal and lateral directions, and are in the opposite direction to the relative motion between the wheel and rail (Thompson, 2009). Formally, creepage is defined as the relative (sliding) velocity v normalised by the rolling velocity V (vehicle speed), and can be divided into three components: a longitudinal creep (ν_x), a lateral creep (ν_y) and an angular sliding velocity around an axis system normal to the contact patch, which is called spin (ω) or spin creep (φ) when divided by the vehicle speed:

$$\begin{aligned} \text{Longitudinal creep} : \nu_x &= \frac{v_x}{V} \\ \text{Lateral creep} : \nu_y &= \frac{v_y}{V} \\ \text{Spin creep} : \varphi &= \frac{\omega}{V} \end{aligned} \tag{A.1}$$

Longitudinal creepage is largely a result of over/ under speed due to braking or acceleration, and the rolling radius difference between the nominal and actual rolling radii during curving. Lateral creepage is attributed to the yaw angle (or angle of attack) of the wheelset in curves, whilst the spin is generated due to the conical shape

of the wheels and inclined rails (Shahzamanian Sichani, 2016). The principles of vehicle steering are addressed in the following section, which helps to explain the generation of such forces.

A.1.2 Steering Forces

In a curve a typical vehicle wheelset shifts laterally across the railhead as a result of the centrifugal forces and the conical wheel profile. Figure A.2 illustrates the wheelset sitting at the nominal mid-position when running along a tangent track, where the rolling radius is R_{nom}, namely $R_L = R_R = R_{nom}$. As the wheelset negotiates a curve the lateral motion allows the outer wheels to use their larger radius of their inner edge R_L, whilst the effective diameter of the inner wheelset (R_R) decreases, therefore a difference in rolling radius difference occurs. The circumference of the outer wheel is larger, and will try to roll further than the inner wheel for a given rotational speed. If the wheelset moves far enough laterally, the resulting rolling radius difference will be enough to compensate for the difference in rail lengths. In this case, the single wheelset achieves ”pure rolling” without any frictional forces being generated. If we consider a free wheelset, in which pure rolling is achieved, the position where the wheelset can roll freely round the curve is known as the *equilibrium rolling line*. The following equation links the lateral displacement, y, and the curve radius R:

$$\frac{R_{nom} - \gamma_y}{R_{nom} + \gamma_y} = \frac{R - l_0}{R + l_0} \tag{A.2}$$

The (equilibrium) rolling line offset is therefore:

$$y = \frac{R_{nom} l_0}{R\gamma} \tag{A.3}$$

where R_{nom} is the radius at the contact point where the wheelset is central, l_0 represents half the gauge, R the radius of the curve and γ is the effective conicity.

In reality, however the wheelset is not able to position itself perfectly radially in a curve in most cases, and additionally wheelsets are not ”free” and instead are constrained in order to provide lateral stability against an unstable mode referred to as ”hunting”.

Hunting phenomenon is a kinematic oscillation of a wheelset on lateral and yawing directions, and appears as a swaying motion of the railway vehicle. When a wheelset is unconstrained, hunting is generated even at very low speeds. Modern vehicle wheelsets are therefore mounted most commonly in a rigid bogie, restrained by a large yaw stiffness. Where the wheels are constrained to remain parallel with one

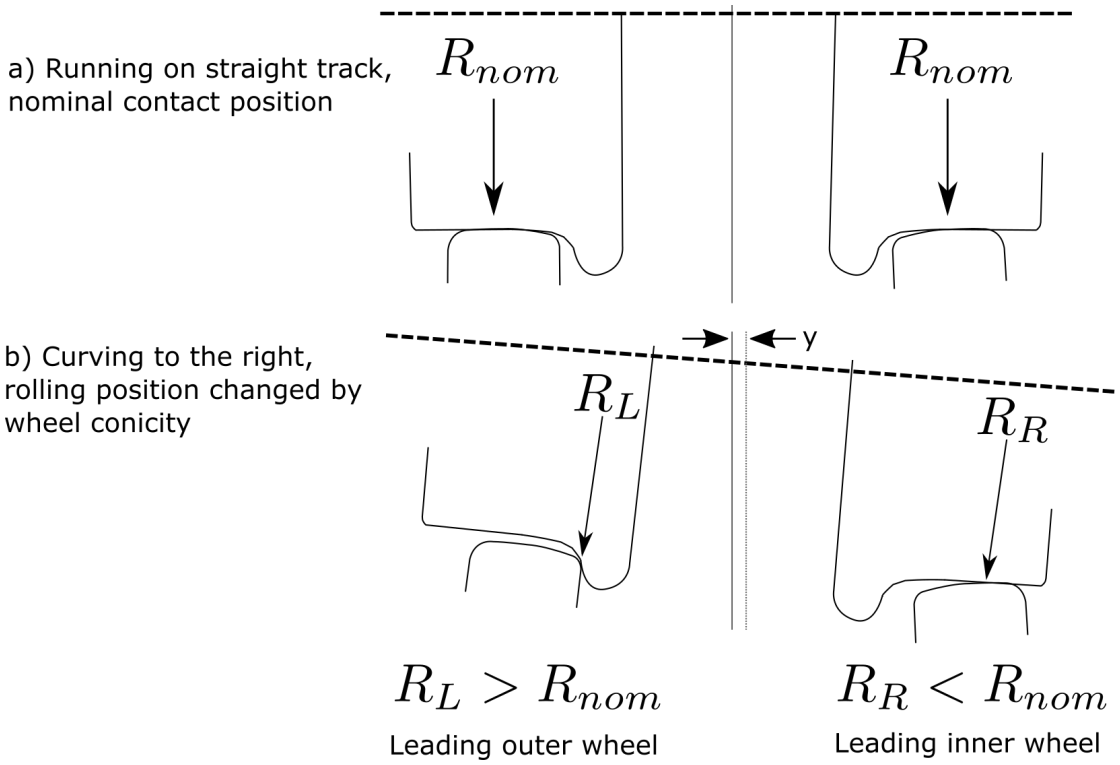


FIGURE A.2: Illustration of railway vehicle self-steering

another, curving performance departs from nominal behaviour as the wheels cannot both align with the curve radius of the track. Creep forces are generated between the wheel and rail attempting to steer the wheelset back into radial alignment, the size of which is dependent on the angle Ψ to be overcome and the yaw stiffness of the primary suspension (see Figure A.3).

Figure A.3 and Figure A.4 show a typical bogie navigating a large radius curve and a small radius curve respectively, both indicate the lateral and yaw positions of the wheelsets relative to the flangeway clearance; the distance between the wheel flange and the railhead. The arrows in the figures illustrate the direction and relative magnitudes of the forces on the wheels, the forces on the rails are in the opposite direction.

In the first curving case, wheelsets are free to curve free of flange contact, the leading wheelset rolls forwards and out from the equilibrium rolling line which generates an RRD excess giving rise to creepage (or micro-slip) between the wheel and rail. Crrepage is defined as the relative velocity normalised by the rolling velocity, and consequently gives rise to longitudinal creep forces which steer the wheelset relative to the rail. In this case, only small lateral forces are generated. Similarly, the required longitudinal steering forces at the trailing wheelset are generated by moving inwards from the equilibrium rolling line (Evans, 2002).

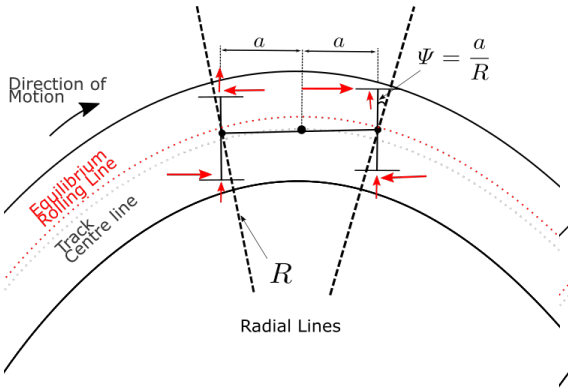


FIGURE A.3: Large Radius Curve No Flange Contact, Small Lateral Forces

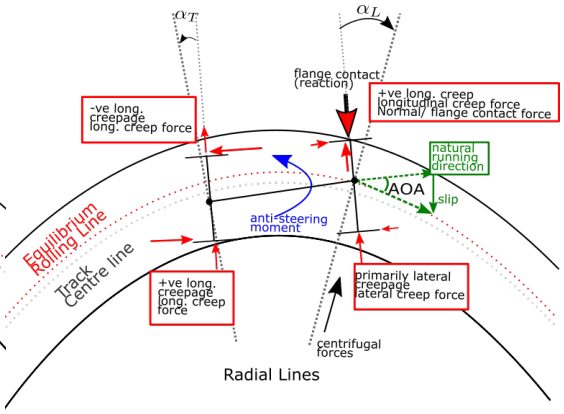


FIGURE A.4: Small Radius Curve: High AOA, Flange Contact, High Lateral Forces

As the radius of the curve tightens, the steering forces required to maintain radial alignment increase. This can be achieved by the wheelset moving farther from the equilibrium rolling line, increasing the RRD and thus the longitudinal creep forces. Eventually as the curve radius reduces, it is no longer possible to develop sufficient creep forces to steer the wheelset since either the limit of available adhesion has been reached, or because the flange prevents the wheelset moving out to generate a large enough RRD. Figure A.3 illustrates a case in which the flange prevents the leading wheelset moving out further to generate a sufficient RRD to steer the wheelset in pure rolling, giving rise to creepage (or micro-slip). The leading axle generates a negative angle of attack α_L, and gives rise to a large lateral creep force at the flange contact point trying to pull the vehicle in the direction that it is pointing. These lateral creep forces will be resisted by opposing flange contact, or normal forces, keeping the vehicle on the track. The friction at the contact becomes saturated, reducing the longitudinal creep forces and hindering steering in the leading wheelset further. On the trailing wheelset, α_T is positive, and the creep force is directed to the inside of the curve and counteracts the effect of the centrifugal force produced by cant deficiency. In general steering breaks down at the leading wheelset first before the trailing wheelset. Firstly, the flangeway clearance enables the bogie to rotate, thus increasing the AOA at the leading wheelset and reducing it at the trailing wheelset. Secondly, the equilibrium rolling line is located towards the outside of the curve, and thus the leading wheelset has a shorter distance to travel before making flange contact than the rear. Hence, it takes a tighter curve to result in flange contact in the rear wheelset, this ultimately results in lower forces in the trailing wheelset.

The primary factors affecting the curving performance include:

- curve radius;
- wheel and rail profiles;

- cant deficiency
- wheelset yaw stiffness;
- bogie wheel base
- axleload;
- coefficient of friction between the wheel and rail.

Ultimately, for a wheelset to steer radially in a curve, the yaw moment, caused by longitudinal creep forces, has to be higher than the resisting forces of the primary suspension. Thus, a more flexible primary suspension (lower resisting forces), or a greater wheel conicity (larger RRD and hence steering force), the smaller the curve radius for which perfect curving will be possible given a particular flangeway clearance. However the compromise is vehicle stability, the greater the conicity, and softer the suspension, the lower the rolling speed at which the wheelset becomes unstable.

A.1.3 Traction and Braking

As mentioned previously, the tangential force, also known as tractive force, and longitudinal creep exist because of the micro-slip that occurs in the rear region of the contact region as traction is transferred to rail due to sliding friction in relation to micro-slip. To brake or accelerate for example, an extra torque is applied to the wheel to generate an under-speed or an over-speed. This results in a relative motion at the point of contact and generates the friction force in the desired direction to adapt the acceleration or braking of the wheel (Shahzamanian Sichani, 2016).

Typical braking and acceleration rates for passenger vehicles are 0.8 m/s^2 (corresponding to 9%g). Assuming a four axle vehicle has a mass of 45 000 kg gives a wheel/rail longitudinal force of 4.5 kN during braking and acceleration, which is significantly less than the tangential forces that are typically generated when a vehicle travels through a curve.

Appendix B

Machine Learning Algorithms

B.1 Supervised Machine Learning Algorithms for Regression

The supervised machine learning algorithms discussed and utilised in this study are as follows:

- Linear Regression,
- Support Vector Regression,
- Nearest Neighbour Algorithms: K-Nearest Neighbours or Radial Neighbours,
- Tree Based Methods,
- Neural Networks, the Multi-layer Perceptron,
- Ensemble Methods: Random Forests.

B.1.1 Linear Regression

Linear regression was first conceptualised in the early 19th century (Stanton, 2001) and is often considered to be within the world of statistics rather than machine learning. The concept of regression is to determine how different variables are related to each other by means of a function or mapping, this is sometimes referred to within machine learning as our hypothesis function $h(x)$. In the non-linear case a curve is used to define the function, this is known as *polynomial regression*, linear regression is a subset in which lines can be used to fit the data. Figure B.1 demonstrates the simplest case of linear regression- *Simple Linear Regression*, with only a single feature.

In the linear setting, we propose that y can be approximated as a linear function of x, the *hypothesis function* is denoted as:

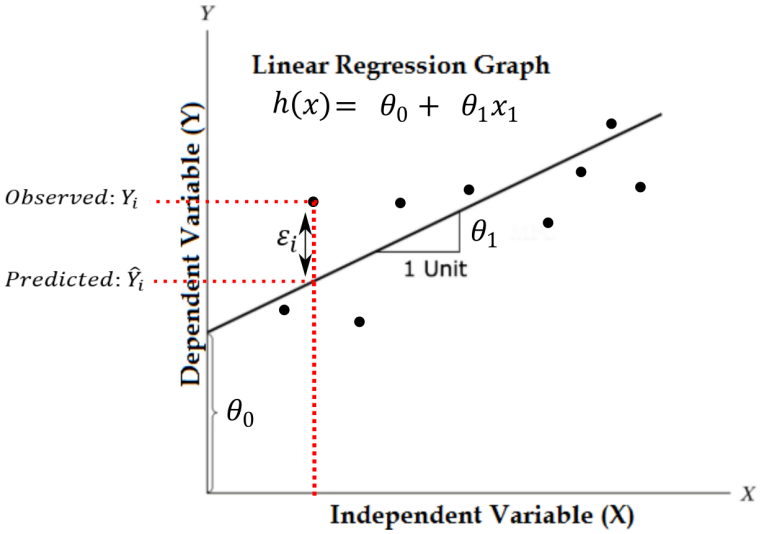


FIGURE B.1: Illustration of Simple Linear Regression

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_n x_n \tag{B.1}$$

Where θ_i' are the *parameters* (or weights) parametrising the space of linear functions mapping from X to Y, where n is the total number of features to be included in the model. It is our objective the learn the optimal weights θ_i for our model to generate the best possible mapping between the input variables X and the target variable Y. These optimal linear predictors are determined from the data by deriving a suitable *cost function*, and solving for the parameters which minimize this cost function. The *cost function* is an indicator of how well the model $h_\theta(x)$ agrees with the data, and for linear regression can be derived given some probabilistic assumptions about the data. Given that we assume the target variables and inputs are related via:

$$y^{(i)} = \hat{\theta}^T x^{(i)} + e^{(i)} \tag{B.2}$$

If we assume that the errors e_i are independently and identically distributed (i.i.d) according to a Gaussian distribution), with mean 0 and variance σ^2 i.e. $e_i \sim N(0, \sigma^2)$, a cost function is determined which corresponds to finding the maximum likelihood estimate of θ: The least-squares cost function is defined:

$$J(\theta) = \frac{1}{2} \sum_{i=1}^{m} (h_\theta(x^{(i)}) \cdot y^{(i)})^2 \tag{B.3}$$

The solution for minimising $J(\theta)$ may be found explicitly, through the use of the 'normal equations' or using an iterative solver such as the method of gradient descent (Ng, 2007). A comprehensive description can be found in (Kutner et al., 2004).

B.1.2 Support Vector Machines

First we present the supervised learning problem for the SVMs. We seek to determine some unknown and non-linear dependency (function) $y = f(x)$ between some high-dimensional input vector x and scalar output y (or vector output in the case of multi-class SVMs). In this case we have no information about the underlying distributions, the only information is the training dataset:
$D = (x_i, y_x) \in X \times Y, i = 1, l$, where l is the number of training data pairs and is therefore equal to the size of the training data set D.

However, unlike the assumptions which are utilised in approaches such as Linear Regression and Neural networks, the assumptions underpinning SVMs are very different. Classic Statistical inference assumes that 1) the data can be modelled by a set of linear in parameter functions, 2) the underlying joint probability distribution is a gaussian distribution, and this assumption enables the maximum likelihood estimator to be used for parameter estimation (which reduces to the minimization of the sum-of-errors-squares cost function as explained in B.1.1). However, in reality many of these assumptions turn out to be problematic and thus the maximum likelihood estimator is no longer appropriate.

Instead the formulation of the SVM is based on the structural risk minimisation principle(SRM). This induction method seeks to minimise an upper bound of the generalisation error consisting of the sum of the training errors and a confidence interval, rather than the commonly used empirical risk minimisation (ERM) principle which minimises the training error(Cao, 2003). The result is to improve the generalisation capability of the algorithm.

In its simplest form, we demonstrate the use of the SVM principle through using a linear separating hyperplane to create a 'maximal margin classifier'. For visualisation purposes Figure B.2 represents a simple, linearly separable binary classification task. Here, a support vector machine aims to pass a hyperplane through the points to classify the data into two groups. The hyperplane is a linear separator which can be extended into any dimension; for example as illustrated a line (2D), a plane (3D), or a hyperplane (4D+). In other words, the decision boundary, i.e. the separation line in input space is defined by the equation $w_1x_1 + w_2x_2 + b = 0$, where w_1, w_2 represent weights. The difficulty is, there are many possible ways of separating the data, and thus a method for obtaining the optimal hyperplane is required such that it generalises as well as possible to new data.

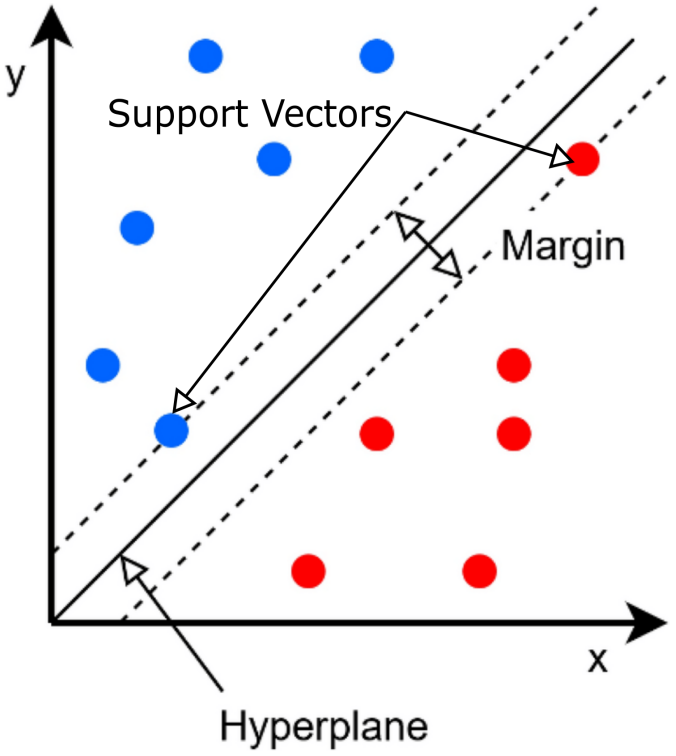


FIGURE B.2: Illustration of Linearly Separable Problem

The proposed solution is to choose the separator that minimise the training error (i.e. empirical risk) with the largest *margin* from the surrounding data points, known as the Maximal Margin Hyperplane (MMH). As seen in Figure B.2, the MMH is the mid line of the widest block that we can place between the two classes such that they are perfectly separated. The location of this hyperplane depends only on the *Support Vectors*, which are the training observations which lie directly on the margin boundary. The maximal margin M is defined (Kecman, 2001):

$$M = \frac{2}{|w|} \tag{B.4}$$

It can further be shown that the minimisation or norm $||w||$ equals a minimisation of $w^T w = \sum_{i=1}^{n} w_i^2$ and this leads to a maximisation of margin M. Hence the learning problem is deduced (note that the multiplication by $1/2$ is for numerical convenience only) (Kecman, 2005):

$$\begin{aligned} \text{Minimise :} \quad & \frac{1}{2} w^T w \\ \text{subject to} \quad & \\ y_i[w^T x_i + b] \geq 1, \; & i = 1, ..., l \end{aligned} \tag{B.5}$$

This is a classic quadratic optimisation problem with inequality constraints and can be solved using Lagrange multipliers.

However, these constraints simply state that each observation must be on the correct side of the hyperplane and at least a distance M from it (i.e. the data are linearly separable). In reality, however a routine is required that is robust to observations which do not enable a perfect separation of classes. This is known as the Support Vector Classifier (SVC) or Soft Margin Classifier (SMC). A SVC enables some observations to be on the incorrect side of the hyperplane. We therefore introduce some 'slack' parameters, namely ξ_i - the slack values and C- a penalty parameter. C, also known as the regularisation constant, controls te trade-off between the model complexity and the empirical risk, which helps to avoid over-fitting. If C is too large, the optimisation will only minimise the empirical risk, regardless of the model complexity.

Thus, instead the hyperplane must satisfy:

$$\begin{aligned} \text{Minimise}: \quad & \frac{1}{2}w^T w + C\sum_{i=1}^{l}\xi_i \\ & \text{subject to} \\ y_i[w^T x_i + b] \geq 1 - \xi_i,\ & i = 1, ..., l,\ \xi_i \geq 0, \end{aligned} \tag{B.6}$$

In summary we see that the resulting formulation simultaneously balances minimising the empirical risk ($C\sum_{i=1}^{l}\xi_i$) via optimisation constraints and $||w||^2$.

In the case of regression, we measure the error of approximation instead of the margin, M used in classification. In this case a novel loss function is introduced here, Vapnik's *linear loss function* with ϵ-insensitivity zone. In principle, the loss function defines a 'tube' such that where a predicted value is inside the tube, then the cost is zero, otherwise the cost is equal to the difference between the predicted value and the radius ϵ of the tube. It follows for regression that we form a linear regression hyperplane $f(x, w) = w^T x + b$ by minimising:

$$\begin{aligned} \text{Minimise}: R_{w,\xi,\xi*} = \frac{1}{2}||w||^2 + C(\sum_{i=1}^{l}\xi_i + \sum_{i=1}^{l}\xi*_i) \\ \text{subject to} \\ y_i - w^T x_i - b \leq \epsilon + \xi_i, \quad i = 1, l, \\ w^T x_i + b - y_i \leq \epsilon + \xi_i^*, \quad i = 1, l, \\ xi*_i \geq 0,\ \xi_i^* \geq 0, \quad i = 1, l, \end{aligned} \tag{B.7}$$

where ξ_i and $\xi^* i$ for measurements above and below the ϵ-tube respectively. Similar to procedures applied in the SV classifiers' design, we solve the constrained optimization problem above by forming a primal variables Lagrangian.

In reality, many problems are not linearly separable even with the allowable errors introduced by soft margins, and therefore a powerful extension is available to this method to allow non-linear decision boundaries. The principle is to use transformation functions to convert the input data into a form which also enables the use of linear solvers. These functions are known as kernel functions, K. Examples of typically used kernel functions are polynomial, gaussian, sigmoid.

B.1.3 Random Forests

B.1.3.1 Decision Trees

A decision tree is considered a map of the reasoning process that helps solve the task of classifying cases into individual categories. The tree structure, as illustrated in FigureB.3 is composed of *nodes* and *branches*, and terminal nodes are called *leaves*. Intermediary nodes, known as decision or internal nodes must be assigned an appropriate splitting attribute/ feature. The choice of the attribute is typically based on some *impurity measure*, which is determined from the subset of the training dataset. The impurity measure is used to calculate the *split measure function* for each feature, and the node is split into child nodes according to the chosen attribute. When a node data cannot be split into additional child nodes, it is called a terminal node. There exist two types of decision trees: binary and non-binary. In the case of non-binary tree, the node is split into as many children as the number of elements of set . Each branch is labelled by a single value of attribute . If the tree is binary, the node is split into two child nodes (Rutkowski et al., 2014).

The problem of constructing optimal binary decision trees is an NP-complete problem and thus theoreticians have searched for efficient heuristics for constructing near-optimal decision trees. The existing algorithms for decision tree construction differ mainly by tree type (binary or non-binary) and *impurity measure*. Some of the most commonly used algorithms are:

- **ID3 Algorithm:** Produces non-binary trees and uses information entropy as the *impurity measure*, the *split measure function* is called information gain. The ID3 algorithm has been extended to algorithms such as ID4 and ID5,
- **C4.5 Algorithm:** Is an updated version of the ID3 algorithm and includes an additional function, called the *split information* which takes high values for attributed with large domains. The *split measure* is then determined as the ratio of the *information gain* and *split information*,

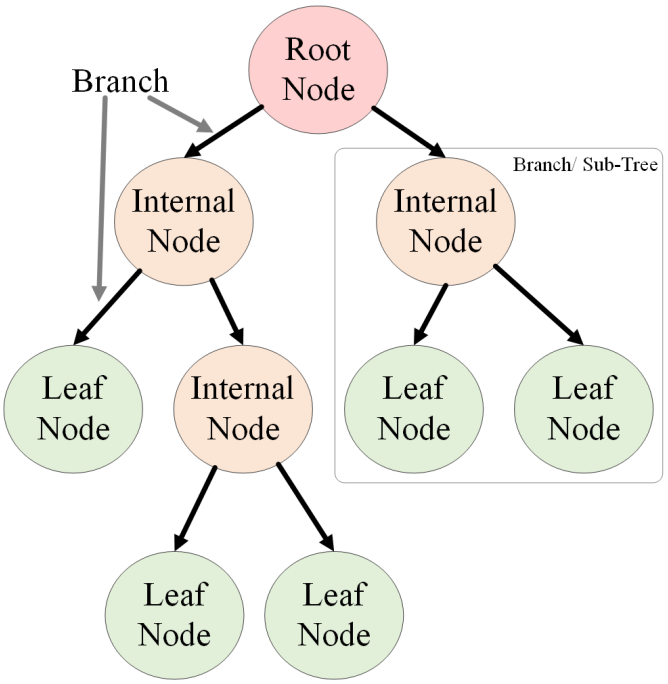


FIGURE B.3: Illustration of Linearly Separable Problem

- **CART Algorithm:** The CART procedure performs "binary recursive partitioning". The term "binary partitioning" as described above implies that the parent nodes are always split into two child nodes, and "recursive" means that the process is repeated by treating each child node as a parent node. This process is repeated until further partitioning is impossible or is limited by some criterion set by the user. Once the first terminal node has been created, the algorithm repeats the procedure for each set of data until all data are categorized as terminal nodes (Steinberg, 2009; Breiman, 1984). The CART algorithm includes seven single-variable splitting criteria, namely—Gini, Symmetric Gini, Twoing, Ordered Twoing, Class Probability for classification trees, Least Squares, and Least Absolute Deviation for regression trees, and also one multi-variable splitting criterion, the Linear Combinations method.

One of the drawbacks of using decision tree algorithms is that, in their construction they are repeatedly build to account for the variability in the data, which can result in overfitting. One way of reducing this problem is to *prune* the tree. The actions of the pruning phase are often referred to as post-pruning in contrast to the pre-pruning that occurs during the growth phase and which aims to prevent splits that do not meet certain specified threshold (e.g. minimum number of observations for a split search, minimum number of observations for a leaf). In addition, we can apply techniques such as bagging, or bootstrap aggregation is a technique for reducing the variance of an estimated prediction function (Hastie et al., 2009). For regression, the same regression tree is fitted many times to bootstrap sampled versions of the training data,

and average the result. The idea is to average many noisy but unbiased models, and hence reduce the variance.

B.1.3.2 Ensemble Learning

Random Forests are a signifiant modification of bagging that builds a large collection of de-correlated trees where each tree is trained independently on a random subset of the data.

Although any one tree may be somewhat over-trained, the randomness in the training process encourages the trees to give independent estimates, which can be combined to achieve accurate and robust results.

Random Forests are highly interpretable, fast to train and simple to use with very few hyper-parameters, making the technique a popular choice for many applications. The key algorithm hyper-parameters are:

- **N-estimators:** The number of trees in the forest. Increasing this parameter slows the learning process, a typical setting is 100 estimators,
- **Maximum tree depth:** The maximum depth of the tree. If this is parameter is not restricted then the tree continues to grow until all leaves are pure or the minimum samples in split criteria is reached, over fitting is thus more likely,
- **Minimum Samples in Split:** The minimum number of samples required to split an internal node. Setting this to a moderate level prevents the tree from becoming too complex and over fitting to the training data.
- **Maximum Number of Features:** The number of features to consider when looking for the best split, all features are considered for each split rather than a random subset.

B.1.4 K-Nearest Neighbours

The K-Nearest Neighbours (kNN) algorithm is one of the oldest and simplest methods for data classification tasks, but has nevertheless proved effective in many applications (Hart, 1968). In the case of classification, the kNN rule assigns each unlabelled example by the majority label among its k nearest neighbours in the *training set*. Or in the case of regression, the label assigned is the average over the k-nearest neighbours. The performance of the algorithm depends heavily on the choice of k, increasing k tends to smooth out decision boundaries, which is useful where the data set is noisy, whereas small values of k tend to result in over-fitting.

The steps of the estimation procedure are as follows, after specifying a value for k, for each observation x in the test data set:

1. **Find the k-nearest neighbours for observation x:**
   - Calculate the distance between x and each point in the training data-set using one of the distance functions such as:
     - Euclidean:,
     $$d(x, y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2} \tag{B.8}$$
     - Manhattan:,
     $$d(x, y) = \sum_{i=1}^{m} |x_i - y_i| \tag{B.9}$$
   - Sort the values in ascending order based on distances, i.e. closest observations first,
   - Find the top k values from the list, i.e. the k-nearest neighbours for point x,
2. **Estimate value/ label for x:** Assign the new data points based on the average of the k-nearest neighbours for that point, additionally weights can be introduced, such that nearer neighbours contribute more to the average than more distant points (Imandoust, 2013).

From the steps defined above it is clear that, unlike many other machine learning algorithms there is not defined training and testing process, i.e. all computation is deferred until the test examples require classifying (or estimating). This is known as memory based classification, or lazy learning, and comes with greater storage requirements and high computational cost (Aha, 1997). Improvements such as reducing the scale of the search for the k nearest neighbours (Sun et al., 2009), and incorporating techniques such as clustering to reduce the search space (Taneja et al., 2014).

Other drawbacks of the kNN algorithm include the choice of k, for many cases deciding the output of each test example using the same number of neighbours k as other test points is not suitable, and therefore adaptive versions (AdaNN) of the algorithm have since been developed (Sun, 2010).

A summary of the key hyper-parameters are as follows:

- **Number of Neighbours:** the number of neighbours to include during the value estimation or classification or an observation,
- **Weights:** with *Uniform* weights, all points in each neighbourhood are weighted equally, whereas *distance* based weighting weights points by the inverse of their distance, closer points will have more of an influence than points further away.

- **Metric:** The distance metric to use for the tree, typically *Minkowski* is used, if $P = 1$ then *Manhattan*, $P = 2$ then standard *Euclidean*. Other possible distance metrics are Cosine and Jaccard.
- **P:** power parameter used for the Minkowski metric, if p =1 this is the equivalent of using the Manhattan distance metric, if p = 2, then equivalent to using standard Euclidean metric.
- **Leaf Size:** affects the speed of the construction and query as well as memory.

B.1.5 Multi Layer Perceptron

Deep Learning is a specialisation of machine learning (sometimes referred to as modern machine learning), and consists of more complex methods which generally are based on deep neural networks. Neural Networks (NNs) utilise many layers of non-linear processing of information in order to perform supervised or unsupervised feature extraction, pattern analysis, classification and prediction. In particular, Deep Learning (DL) aids in learning more complex features and input-output relationships in a given set of data, and additionally can learn multiple levels of useful representation from such as images, sound, text and numeric data types.

The techniques based on Artificial Neural Network (ANN) are especially effective in the solution of high complexity problems for which a traditional mathematical model is difficult to build (where the nature of the input-output relationship is neither well defined nor easily computable) (Szkuta et al., 1999).

However, it is often difficult to ascertain why a neural net produces a particular result, and they are critically labelled as 'black boxes', in which the users do not understand the inner workings.

Neural networks, as their name suggests, are inspired by the biological brain and the nervous system. The resemblance is two fold, 1) knowledge is acquired by the network through a learning process, and 2) interconnection strengths known as synaptic weights are used to stored the knowledge (Haykin, 1998). In summary, the technique learns through a process by which the free parameters (i.e. synaptic weights and bias levels) of a network are adapted through a continuing process of stimulation by the environment in which the network is embedded (i.e. the information it is provided).

A neural network consists of a number of interconnected processing elements called *neurons*, which can be perceived as a processing unit. In a neural network neurons are connected to each other through a *'synaptic weight'* (or just *'weight'*). Each neuron in a network receives *'weighted'* information via these connections from the neurons that it is connected to and produces an output by passing a weighted sum of those input

signals (either external inputs from the environment or the outputs of other neurons) through an *'activation function'* (such as a step function). This principle is illustrated in Figure B.4, which shows a single layer neural network, known as a *Perceptron*. The figure shows the flow of information, the inputs are first multiplied by their weights, then a weighted sum is determined, and an activation function is used to map the input to the required values such as (0,1) or (-1, 1). The perceptron is the building block of a neural network which typically consists of multiple such elements to generate a non-linear mapping.

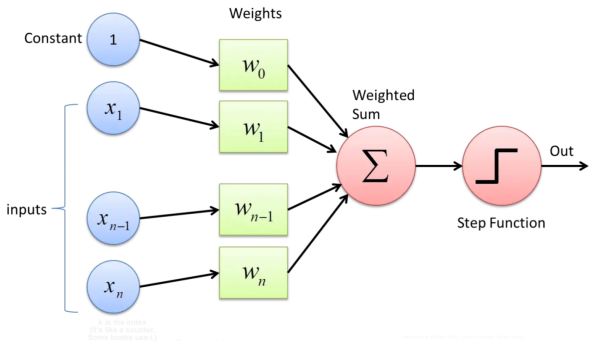


FIGURE B.4: An Illustration of a Perceptron

There are two-main categories of network architectures which depend on the type of connections between the neurons, these two architectures may be applied to different learning problems and include:

- **Feed Forward Networks:** A Feed-forward Neural Network (FNN) is an artificial neural network wherein connections between the nodes do not form a cycle, i.e. there is no feedback from the outputs of the neurons towards the inputs throughout the network. The feed-forward neural network was the first and simplest type of artificial neural network devised. In this network, the information moves in only one direction—forward—from the input nodes, through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network. A MLP is a supplement of the feed-forward neural network, consisting of three types of layers- input layers, hidden layer, output layer which are shown in Figure B.5. The learning procedure for determining network weights uses the *back-propagation algorithm* which is simply described as follows:

  1. Starting with the input layer, propagate data forward to the output layer. This step is the forward propagation.
  2. Based on the output, calculate the error (the difference between the predicted and known outcome). The error needs to be minimized.
  3. Back-propagate the error. Find its derivative with respect to each weight in the network, and update the model.

Repeat the three steps given above over multiple epochs to learn the ideal weights. An epoch is one cycle of the learning process for a neural network, a forward pass and a backward pass together are counted as one pass. Finally, the output is taken via a threshold function to obtain the predicted class labels.

A MLP can approximate any continuous value function and solve problems which are not linearly separable. The major use cases of MLP are pattern classification, recognition, prediction and approximation. Additionally Convolutional Neural Network (CNN) are further extensions of the FNN, which are widely used for image classification, image clustering and object detection in images. They are also employed for optical character recognition and natural language processing.

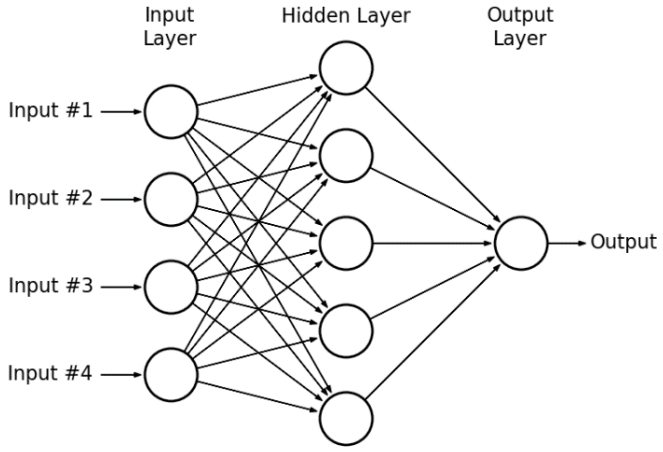


FIGURE B.5: A Hypothetical Example of a Multilayer Perceptron Network (Hassan et al., 2015)

- **Recurrent Neural Network (RNN):** On the other hand, where there exists such feedback, i.e. a connection from the outputs towards the inputs, then the network is called a 'recurrent neural network'. A Recurrent Neural Network (RNN) (unlike feed-forward architectures such as MLP)enable modelling of sequence data such as sound, time series sensor data or written natural language.

Other extensions of these method do exist such as Restricted Boltzmann machines, Deep Belief Networks, Generative adversarial networks and Deep auto-encoders.

Some of the key hyper-parameters for the MLP are as follows:

- **Hidden Layer Sizes:** The number of neurons in the *i*th hidden layer,
- **Activation:** Activation function for the hidden layer, typically the rectified linear unit function is used,

- **Solver:** The solver for weight optimisation.
- **Max. Iterations:** The maximum number of iterations for weight optimisation,
- **Alpha:** The strength of the L2 regularisation term. L2 is divided by the sample size when added to the loss. The term combats overfitting by constraining the size of the weights. Increasing alpha may fix high variance (a sign of overfitting) by encouraging smaller weights, resulting in a smoother decision boundary. Similarly, decreasing alpha may fix high bias (a sign of under-fitting) by encouraging larger weights, potentially resulting in a more complicated decision boundary,
- **Learning Rate:** The learning rate schedule for weight updates,
- **Tolerance:** Tolerance for the optimisation, determines when convergence is reached.

Appendix C

Regression Analysis: Results

C.1 Results 1: Model Comparisons

TABLE C.1: Results: Outlier Factor: 0

		Smoothing Window												
Input Features	**Regression Model**	**0**	**2**	**4**	**5**	**6**	**8**	**10**	**12**	**14**	**15**	**20**	**25**	**30**
Set 0	0	0.12	0.16	0.20	0.21	0.21	0.21	0.22	0.22	0.21	0.22	0.21	0.20	0.19
	1	0.29	0.23	0.31	0.26	0.33	0.46	0.53	0.41	0.46	0.49	0.46	0.00	0.27
	2	0.12	0.16	0.20	0.21	0.21	0.21	0.22	0.22	0.21	0.22	0.21	0.20	0.19
	3	0.09	0.15	0.23	0.25	0.27	0.30	0.33	0.32	0.33	0.32	0.29	0.26	0.22
	4	0.19	0.26	0.41	0.39	0.42	0.53	0.56	0.47	0.43	0.49	0.56	0.45	0.48
	5	0.24	0.30	0.44	0.44	0.44	0.54	0.59	0.53	0.49	0.53	0.59	0.50	0.54
	6	0	0	0	0	0	0	0	0	0	0	0	0	0
Set 1	0	0.19	0.19	0.23	0.24	0.25	0.26	0.27	0.28	0.28	0.29	0.28	0.28	0.28
	1	0.44	0.43	0.62	0.56	0.63	0.71	0.75	0.61	0.66	0.72	0.85	0.77	0.83
	2	0.19	0.19	0.23	0.24	0.25	0.26	0.27	0.28	0.28	0.29	0.28	0.28	0.28
	3	0.29	0.31	0.43	0.44	0.48	0.54	0.58	0.60	0.61	0.62	0.64	0.64	0.66
	4	0.36	0.51	0.69	0.71	0.74	0.81	0.81	0.83	0.84	0.84	0.86	0.88	0.89
	5	0.41	0.51	0.68	0.68	0.72	0.79	0.80	0.82	0.83	0.83	0.85	0.86	0.88
	6	0	0	0	0	0	0	0	0	0	0	0	0	0
Set 2	0	0.19	0.19	0.23	0.24	0.25	0.26	0.27	0.28	0.28	0.29	0.28	0.28	0.28
	1	0.44	0.43	0.62	0.56	0.63	0.71	0.75	0.61	0.66	0.72	0.85	0.76	0.83
	2	0.19	0.19	0.23	0.24	0.25	0.26	0.27	0.28	0.28	0.29	0.28	0.28	0.28
	3	0.29	0.32	0.43	0.46	0.49	0.55	0.59	0.60	0.62	0.63	0.64	0.64	0.67
	4	0.35	0.46	0.64	0.67	0.70	0.79	0.81	0.82	0.84	0.84	0.85	0.87	0.89
	5	0.39	0.46	0.63	0.65	0.69	0.77	0.78	0.81	0.82	0.82	0.83	0.85	0.87

Continued on next page

		Smoothing Window												
Input Features	**Regression Model**	**0**	**2**	**4**	**5**	**6**	**8**	**10**	**12**	**14**	**15**	**20**	**25**	**30**
	6	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0.21	0.21	0.26	0.27	0.29	0.30	0.31	0.31	0.32	0.33	0.33	0.34	0.35
	1	0.44	0.44	0.64	0.58	0.64	0.72	0.75	0.62	0.67	0.73	0.86	0.77	0.86
	2	0.21	0.21	0.26	0.27	0.29	0.30	0.31	0.31	0.32	0.33	0.33	0.34	0.35
Set 3	3	0.33	0.39	0.50	0.53	0.57	0.64	0.69	0.71	0.73	0.72	0.73	0.74	0.77
	4	0.32	0.44	0.64	0.66	0.71	0.79	0.82	0.82	0.84	0.84	0.84	0.88	0.90
	5	0.36	0.46	0.62	0.65	0.70	0.77	0.79	0.81	0.83	0.83	0.83	0.87	0.87
	6	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	-0.05	0.09	0.19	0.26	0.22	0.31	0.31	0.30	0.33	0.34	0.34	0.35	0.35
	1	0.47	0.46	0.65	0.65	0.65	0.75	0.73	0.64	0.69	0.71	0.84	0.74	0.84
	2	-0.05	0.09	0.19	0.26	0.22	0.31	0.31	0.30	0.33	0.34	0.34	0.35	0.35
Set 4	3	0.31	0.40	0.53	0.56	0.61	0.67	0.72	0.74	0.75	0.73	0.74	0.76	0.78
	4	0.31	0.42	0.62	0.64	0.68	0.77	0.81	0.82	0.83	0.84	0.86	0.88	0.89
	5	0.35	0.44	0.60	0.62	0.67	0.74	0.78	0.80	0.81	0.81	0.83	0.85	0.86
	6	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	-0.04	0.10	0.20	0.26	0.22	0.31	0.31	0.30	0.33	0.34	0.34	0.36	0.35
	1	0.48	0.46	0.66	0.65	0.65	0.75	0.74	0.64	0.69	0.71	0.84	0.75	0.85
	2	-0.04	0.10	0.20	0.26	0.22	0.31	0.31	0.30	0.33	0.34	0.34	0.36	0.35
Set 5	3	0.31	0.39	0.53	0.56	0.61	0.67	0.72	0.74	0.75	0.73	0.74	0.76	0.78
	4	0.29	0.41	0.60	0.63	0.68	0.76	0.81	0.83	0.83	0.84	0.86	0.88	0.90
	5	0.34	0.43	0.59	0.61	0.66	0.74	0.78	0.80	0.81	0.81	0.83	0.86	0.86
	6	0	0	0	0	0	0	0	0	0	0	0	0	0

Continued on next page

		Smoothing Window												
Input Features	**Regression Model**	**0**	**2**	**4**	**5**	**6**	**8**	**10**	**12**	**14**	**15**	**20**	**25**	**30**
Set 6	0	-0.04	0.10	0.20	0.26	0.22	0.31	0.31	0.30	0.33	0.34	0.34	0.36	0.35
	1	0.48	0.46	0.66	0.65	0.66	0.75	0.73	0.64	0.69	0.71	0.85	0.75	0.85
	2	-0.03	0.10	0.20	0.26	0.22	0.31	0.31	0.30	0.33	0.34	0.34	0.36	0.35
	3	0.31	0.39	0.53	0.56	0.61	0.67	0.72	0.74	0.75	0.73	0.74	0.76	0.78
	4	0.29	0.41	0.60	0.63	0.67	0.76	0.81	0.83	0.83	0.84	0.86	0.88	0.90
	5	0.34	0.43	0.59	0.61	0.66	0.74	0.78	0.80	0.81	0.81	0.83	0.86	0.86
	6	0	0	0	0	0	0	0	0	0	0	0	0	0
Set 7	0	0.62	0.73	0.77	0.80	0.80	0.83	0.83	0.84	0.84	0.84	0.85	0.86	0.85
	1	0.68	0.76	0.82	0.85	0.86	0.88	0.90	0.90	0.91	0.91	0.92	0.92	0.91
	2	0.62	0.73	0.77	0.80	0.80	0.83	0.83	0.84	0.84	0.84	0.85	0.86	0.85
	3	0.45	0.56	0.69	0.72	0.74	0.78	0.82	0.85	0.86	0.84	0.84	0.85	0.85
	4	0.55	0.67	0.80	0.83	0.84	0.89	0.91	0.91	0.92	0.93	0.93	0.95	0.95
	5	0.57	0.68	0.79	0.82	0.84	0.88	0.90	0.91	0.92	0.92	0.93	0.94	0.94
	6	0	0	0	0	0	0	0	0	0	0	0	0	0
Max	0.68	0.76	0.82	0.85	0.86	0.89	0.91	0.91	0.92	0.93	0.93	0.95	0.95	

C.2 Results 2: Selected Models

TABLE C.2: Results: Hyper-parameter Tuning: RF, Feature Set 7

Index	r^2: OF-None	r^2: OF1	r^2: OF2	**Range** r^2	**max depth**	**Max Features**	**Min Samples Split**	**No. Trees**
0	0.817	0.821	0.821	0.038	5	0.5	20	25
1	0.821	0.821	0.821	0.033	5	0.5	20	50
2	0.823	0.823	0.823	0.028	5	0.5	20	75
3	0.823	0.824	0.824	0.031	5	0.5	20	100
4	0.814	0.815	0.815	0.051	5	0.5	40	25
5	0.817	0.815	0.815	0.037	5	0.5	40	50
6	0.818	0.818	0.818	0.031	5	0.5	40	75
7	0.818	0.818	0.819	0.032	5	0.5	40	100
8	0.811	0.812	0.812	0.044	5	0.5	60	25
9	0.812	0.812	0.813	0.033	5	0.5	60	50
10	0.814	0.815	0.816	0.031	5	0.5	60	75
11	0.813	0.815	0.816	0.033	5	0.5	60	100
12	0.804	0.81	0.81	0.043	5	0.5	80	25
13	0.809	0.81	0.811	0.032	5	0.5	80	50
14	0.81	0.812	0.813	0.032	5	0.5	80	75
15	0.809	0.812	0.812	0.038	5	0.5	80	100
16	0.821	0.822	0.822	0.037	5	1	20	25
17	0.823	0.823	0.823	0.035	5	1	20	50
18	0.824	0.823	0.824	0.036	5	1	20	75
19	0.823	0.823	0.823	0.035	5	1	20	100
20	0.817	0.818	0.818	0.038	5	1	40	25
21	0.819	0.82	0.82	0.035	5	1	40	50
22	0.82	0.82	0.82	0.035	5	1	40	75
23	0.82	0.82	0.82	0.036	5	1	40	100
24	0.817	0.816	0.816	0.036	5	1	60	25
25	0.817	0.818	0.818	0.035	5	1	60	50
26	0.818	0.818	0.818	0.035	5	1	60	75
27	0.818	0.818	0.818	0.035	5	1	60	100
28	0.813	0.812	0.813	0.038	5	1	80	25
29	0.814	0.815	0.814	0.035	5	1	80	50
30	0.815	0.815	0.815	0.035	5	1	80	75
31	0.815	0.815	0.815	0.035	5	1	80	100
32	0.821	0.822	0.822	0.037	5	All	20	25
33	0.823	0.823	0.823	0.035	5	All	20	50
34	0.824	0.823	0.824	0.036	5	All	20	75
35	0.823	0.823	0.823	0.035	5	All	20	100

Continued on next page

Index	r^2: OF-None	r^2: OF1	r^2: OF2	**Range** r^2	**max depth**	**Max Features**	**Min Samples Split**	**No. Trees**
36	0.817	0.818	0.818	0.038	5	All	40	25
37	0.819	0.82	0.82	0.035	5	All	40	50
38	0.82	0.82	0.82	0.035	5	All	40	75
39	0.82	0.82	0.82	0.036	5	All	40	100
40	0.817	0.816	0.816	0.036	5	All	60	25
41	0.817	0.818	0.818	0.035	5	All	60	50
42	0.818	0.818	0.818	0.035	5	All	60	75
43	0.818	0.818	0.818	0.035	5	All	60	100
44	0.813	0.812	0.813	0.038	5	All	80	25
45	0.814	0.815	0.814	0.035	5	All	80	50
46	0.815	0.815	0.815	0.035	5	All	80	75
47	0.815	0.815	0.815	0.035	5	All	80	100
48	0.852	0.856	0.852	0.026	10	0.5	20	25
49	0.856	0.857	0.856	0.025	10	0.5	20	50
50	0.856	0.859	0.858	0.026	10	0.5	20	75
51	0.856	0.859	0.859	0.027	10	0.5	20	100
52	0.841	0.844	0.843	0.026	10	0.5	40	25
53	0.843	0.845	0.845	0.023	10	0.5	40	50
54	0.845	0.846	0.846	0.025	10	0.5	40	75
55	0.845	0.847	0.846	0.029	10	0.5	40	100
56	0.834	0.837	0.837	0.028	10	0.5	60	25
57	0.837	0.838	0.836	0.033	10	0.5	60	50
58	0.837	0.838	0.839	0.031	10	0.5	60	75
59	0.837	0.839	0.839	0.03	10	0.5	60	100
60	0.824	0.829	0.828	0.031	10	0.5	80	25
61	0.828	0.831	0.832	0.031	10	0.5	80	50
62	0.83	0.832	0.833	0.029	10	0.5	80	75
63	0.83	0.833	0.833	0.035	10	0.5	80	100
64	0.844	0.845	0.844	0.029	10	1	20	25
65	0.846	0.847	0.848	0.026	10	1	20	50
66	0.847	0.848	0.848	0.026	10	1	20	75
67	0.847	0.847	0.848	0.026	10	1	20	100
68	0.837	0.837	0.837	0.029	10	1	40	25
69	0.839	0.839	0.84	0.026	10	1	40	50
70	0.84	0.841	0.841	0.026	10	1	40	75
71	0.84	0.841	0.841	0.027	10	1	40	100
72	0.833	0.833	0.834	0.028	10	1	60	25
73	0.836	0.836	0.837	0.026	10	1	60	50
74	0.836	0.837	0.837	0.026	10	1	60	75
75	0.836	0.837	0.837	0.026	10	1	60	100
76	0.828	0.828	0.829	0.027	10	1	80	25

Continued on next page

Index	r^2: OF-None	r^2: OF1	r^2: OF2	**Range** r^2	**max depth**	**Max Features**	**Min Samples Split**	**No. Trees**
77	0.831	0.83	0.832	0.026	10	1	80	50
78	0.831	0.831	0.831	0.027	10	1	80	75
79	0.831	0.831	0.831	0.027	10	1	80	100
80	0.844	0.845	0.844	0.029	10	All	20	25
81	0.846	0.847	0.848	0.026	10	All	20	50
82	0.847	0.848	0.848	0.026	10	All	20	75
83	0.847	0.847	0.848	0.026	10	All	20	100
84	0.837	0.837	0.837	0.029	10	All	40	25
85	0.839	0.839	0.84	0.026	10	All	40	50
86	0.84	0.841	0.841	0.026	10	All	40	75
87	0.84	0.841	0.841	0.027	10	All	40	100
88	0.833	0.833	0.834	0.028	10	All	60	25
89	0.836	0.836	0.837	0.026	10	All	60	50
90	0.836	0.837	0.837	0.026	10	All	60	75
91	0.836	0.837	0.837	0.026	10	All	60	100
92	0.828	0.828	0.829	0.027	10	All	80	25
93	0.831	0.83	0.832	0.026	10	All	80	50
94	0.831	0.831	0.831	0.027	10	All	80	75
95	0.831	0.831	0.831	0.027	10	All	80	100

TABLE C.3: Results: Hyper-parameter Tuning: Linear SVR, Feature Set 7

Index	r^2: OF- None	r^2: OF1	r^2: OF2	C	epsilon
0	0.8031	0.8029	0.8027	0.1	0
1	0.8027	0.8027	0.8027	0.1	0.01
2	0.799	0.7989	0.799	0.1	0.1
3	0.8027	0.8025	0.8024	1	0
4	0.8025	0.8023	0.8023	1	0.01
5	0.7986	0.7985	0.7984	1	0.1
6	0.8026	0.8025	0.8026	10	0
7	0.8024	0.8023	0.8023	10	0.01
8	0.7985	0.7986	0.7985	10	0.1
9	0.8027	0.8026	0.8022	100	0
10	0.8023	0.8024	0.8023	100	0.01
11	0.7985	0.7986	0.7984	100	0.1

TABLE C.4: Results: Hyper-parameter Tuning: Gaussian SVR, Feature Set 7

Index	r^2: OF- None	r^2: OF1	r^2: OF2	C	epsilon	gamma
0	0.7316	0.7081	0.7153	0.1	0	0.001
1	0.7044	0.6577	0.6739	0.1	0	0.01
2	0.4513	0.3866	0.416	0.1	0	0.1
3	0.732	0.7082	0.7157	0.1	0.01	0.001
4	0.7042	0.658	0.6739	0.1	0.01	0.01
5	0.4517	0.3871	0.4166	0.1	0.01	0.1
6	0.7324	0.7098	0.7171	0.1	0.1	0.001
7	0.7067	0.6624	0.6785	0.1	0.1	0.01
8	0.4582	0.3951	0.4244	0.1	0.1	0.1
9	0.7936	0.7815	0.7851	1	0	0.001
10	0.8184	0.8055	0.8106	1	0	0.01
11	0.738	0.7021	0.7163	1	0	0.1
12	0.7938	0.7818	0.7853	1	0.01	0.001
13	0.8186	0.8057	0.811	1	0.01	0.01
14	0.7382	0.7022	0.7167	1	0.01	0.1
15	0.7968	0.7852	0.7884	1	0.1	0.001
16	0.8213	0.8085	0.813	1	0.1	0.01
17	0.7399	0.7037	0.7173	1	0.1	0.1
18	0.813	0.8052	0.8068	10	0	0.001
19	0.8552	0.8486	0.8546	10	0	0.01
20	0.839	0.8249	0.8299	10	0	0.1
21	0.8131	0.8055	0.8071	10	0.01	0.001
22	0.8553	0.8487	0.8549	10	0.01	0.01
23	0.8394	0.825	0.83	10	0.01	0.1
24	0.817	0.8091	0.8113	10	0.1	0.001
25	0.8579	0.8513	0.8564	10	0.1	0.01
26	0.8386	0.8224	0.8273	10	0.1	0.1
27	0.8274	0.8124	0.8213	100	0	0.001
28	0.8649	0.8571	0.8618	100	0	0.01
29	0.8186	0.8095	0.815	100	0	0.1
30	0.8277	0.8128	0.8218	100	0.01	0.001
31	0.8657	0.8578	0.8621	100	0.01	0.01
32	0.8197	0.8104	0.8157	100	0.01	0.1
33	0.8308	0.8179	0.826	100	0.1	0.001
34	0.8681	0.8601	0.8636	100	0.1	0.01
35	0.8235	0.8131	0.8188	100	0.1	0.1

TABLE C.5: Results: Hyper-parameter Tuning: kNN Uniform, Feature Set 7

Index	r^2: OF- None	r^2: OF1	r^2: OF2	k	weights
0	0.8396	0.8241	0.8309	5	uniform
1	0.8113	0.7905	0.7978	10	uniform
2	0.7765	0.7438	0.7605	20	uniform
3	0.7267	0.6888	0.7049	40	uniform
4	0.6695	0.6198	0.6435	80	uniform
5	0.6163	0.5635	0.5887	160	uniform

TABLE C.6: Results: Hyper-parameter Tuning: kNN Distance, Feature Set 7

Index	r^2: OF- None	r^2: OF1	r^2: OF2	k	Weights
0	0.8592	0.8488	0.8521	5	distance
1	0.8466	0.8342	0.8365	10	distance
2	0.8243	0.8061	0.8123	20	distance
3	0.788	0.7632	0.771	40	distance
4	0.7392	0.7031	0.7167	80	distance
5	0.6853	0.6418	0.6593	160	distance

TABLE C.7: Results: Hyper-parameter Tuning: MLP, Feature Set 7

Index	r^2: OF- None	r^2: OF1	r^2: OF2	Alpha	Hidden Layer Sizes	Max. Iterations
0	0.4293	-1.9888	-12.2598	0.0001	(10,)	500
1	0.555	-1.7061	-5.9686	0.0001	(10,)	1000
2	0.3714	-6.8149	-15.2392	0.0001	(50,)	500
3	0.3776	-414.0033	-261.6909	0.0001	(50,)	1000
4	0.066	-103.5081	-621.683	0.0001	(100,)	500
5	-0.1735	-564.3177	-216.8023	0.0001	(100,)	1000
6	0.5127	-605.6314	-1550.7774	0.0001	(150,)	500
7	0.3324	-579.5402	-139.7467	0.0001	(150,)	1000
8	0.6658	-10.9096	-3.4702	0.0001	(10, 10)	500
9	0.743	0.1233	-183.5286	0.0001	(10, 10)	1000
10	0.6282	-132.6529	-2702.4027	0.0001	(50, 50)	500
11	0.6383	-157.2431	-144.7607	0.0001	(50, 50)	1000
12	0.5695	-200.8833	-72.1203	0.0001	(100, 100)	500
13	0.5427	-668.3284	-1449.452	0.0001	(100, 100)	1000
14	0.5006	-134.6024	-124.0172	0.0001	(150, 150)	500
15	0.6077	-757.7507	-1211.1036	0.0001	(150, 150)	1000
16	0.8187	-7.2695	-17.0073	0.0001	(10, 10, 10)	500
17	0.8204	-0.7128	-18.9069	0.0001	(10, 10, 10)	1000
18	0.7699	-52.5498	-13.4315	0.0001	(50, 50, 50)	500
19	0.7827	-13.2516	-599.3989	0.0001	(50, 50, 50)	1000
20	0.7552	-21.021	-264.718	0.0001	(100, 100, 100)	500
21	0.8333	-0.7688	-51.0198	0.0001	(100, 100, 100)	1000
22	0.7807	-8.6755	-145.3774	0.0001	(150, 150, 150)	500
23	0.8128	-19.7202	-69.3327	0.0001	(150, 150, 150)	1000
24	0.3196	-7.8384	0.2327	0.001	(10,)	500
25	0.2437	-0.0301	-31.5949	0.001	(10,)	1000
26	0.0246	-78.7186	-2.3244	0.001	(50,)	500
27	0.3702	-6.6913	-658.8468	0.001	(50,)	1000
28	0.1315	-123.1272	-112.2057	0.001	(100,)	500
29	0.4038	-12.1274	-173.9801	0.001	(100,)	1000
30	0.5157	-363.3368	-2713.3598	0.001	(150,)	500
31	-0.6334	-157.8776	-651.4548	0.001	(150,)	1000

Continued on next page

Index	r^2: OF- None	r^2: OF1	r^2: OF2	Alpha	Hidden Layer Sizes	Max. Iterations
32	0.492	-0.1767	0.2109	0.001	(10, 10)	500
33	0.7319	-4.1582	-16.3545	0.001	(10, 10)	1000
34	0.6717	-584.3937	-278.7636	0.001	(50, 50)	500
35	0.6718	-14.9049	-421.868	0.001	(50, 50)	1000
36	0.609	-147.4449	-7435.6625	0.001	(100, 100)	500
37	0.694	-194.3486	-45.0211	0.001	(100, 100)	1000
38	0.5482	-440.0282	-500.3461	0.001	(150, 150)	500
39	0.6868	-80.122	-24.9888	0.001	(150, 150)	1000
40	0.8295	-2.8777	-0.3529	0.001	(10, 10, 10)	500
41	0.8351	-2.3924	-0.2669	0.001	(10, 10, 10)	1000
42	0.751	-459.0997	-13.6008	0.001	(50, 50, 50)	500
43	0.801	-110.6572	-83.7297	0.001	(50, 50, 50)	1000
44	0.7693	-22.5649	-13.1217	0.001	(100, 100, 100)	500
45	0.676	-289.804	-569.7783	0.001	(100, 100, 100)	1000
46	0.5806	-150.1574	-24.358	0.001	(150, 150, 150)	500
47	0.7598	-9.9558	-13.2478	0.001	(150, 150, 150)	1000
48	0.3287	-1.2037	-234.9798	0.01	(10,)	500
49	-0.0261	-1.2438	-89.0473	0.01	(10,)	1000
50	0.0359	-82.2769	-1559.8811	0.01	(50,)	500
51	-0.2023	-36.203	-42.4132	0.01	(50,)	1000
52	0.5107	-163.6201	-1562.5629	0.01	(100,)	500
53	0.1888	-34.4618	-220.0302	0.01	(100,)	1000
54	0.4771	-388.7191	-44.1708	0.01	(150,)	500
55	-0.7727	-302.1465	-47.918	0.01	(150,)	1000
56	0.806	-70.336	-3.4946	0.01	(10, 10)	500
57	0.8106	-3.0067	-0.297	0.01	(10, 10)	1000
58	0.689	-255.3956	-93.0901	0.01	(50, 50)	500
59	0.7168	-102.9523	-203.3915	0.01	(50, 50)	1000
60	0.5254	-727.9897	-883.1918	0.01	(100, 100)	500
61	0.5366	-1004.3353	-136.2118	0.01	(100, 100)	1000
62	0.6643	-1111.6144	-7633.8232	0.01	(150, 150)	500
63	0.6501	-27.6218	-237.8633	0.01	(150, 150)	1000
64	0.821	-5.1139	-15.0293	0.01	(10, 10, 10)	500
65	0.7991	-23.4557	-0.4033	0.01	(10, 10, 10)	1000
66	0.7926	-1.1685	-1111.1431	0.01	(50, 50, 50)	500
67	0.8086	-17.9715	-179.9365	0.01	(50, 50, 50)	1000
68	0.7888	-63.4335	-38.1141	0.01	(100, 100, 100)	500
69	0.8048	-7.4766	-9.924	0.01	(100, 100, 100)	1000
70	0.7151	-1.8986	-30.2876	0.01	(150, 150, 150)	500
71	0.8307	0.2082	-13.1303	0.01	(150, 150, 150)	1000
72	0.5795	-1.4562	-133.985	0.1	(10,)	500
73	0.3377	-15.0185	-150.2155	0.1	(10,)	1000
74	-0.0241	-9.6804	-49.3126	0.1	(50,)	500
75	-0.109	-29.3162	-39.596	0.1	(50,)	1000

Continued on next page

Index	r^2: OF- None	r^2: OF1	r^2: OF2	Alpha	Hidden Layer Sizes	Max. Iterations
76	-0.9485	-123.5031	-2179.1003	0.1	(100,)	500
77	-0.4502	-231.1833	-313.9783	0.1	(100,)	1000
78	-2.6537	-114.3072	-23.7173	0.1	(150,)	500
79	-0.5617	-202.3694	-452.4885	0.1	(150,)	1000
80	0.7261	-0.255	-59.1445	0.1	(10, 10)	500
81	0.7409	-31.8655	-0.4785	0.1	(10, 10)	1000
82	0.0973	-711.3337	-22.7863	0.1	(50, 50)	500
83	0.7077	-29.7202	-85.4751	0.1	(50, 50)	1000
84	0.6579	-252.4446	-4162.6416	0.1	(100, 100)	500
85	0.6842	-81.4526	-407.2311	0.1	(100, 100)	1000
86	0.6398	-16.4616	-2233.7242	0.1	(150, 150)	500
87	0.3272	-258.0383	-926.0689	0.1	(150, 150)	1000
88	0.8226	-0.9857	-0.4103	0.1	(10, 10, 10)	500
89	0.8198	-5.7326	-3.6511	0.1	(10, 10, 10)	1000
90	0.8076	-17.5496	-34.5109	0.1	(50, 50, 50)	500
91	0.8471	-40.2505	-16.1575	0.1	(50, 50, 50)	1000
92	0.7745	-21.2071	-67.7571	0.1	(100, 100, 100)	500
93	0.8538	-4.7274	-4.6426	0.1	(100, 100, 100)	1000
94	0.7714	-435.2678	-10.2135	0.1	(150, 150, 150)	500
95	0.852	-40.9399	-30.5603	0.1	(150, 150, 150)	1000
96	0.666	-5.2329	-1.2738	1	(10,)	500
97	-0.1259	-2.1599	-7.0442	1	(10,)	1000
98	0.3736	-40.007	-1.1572	1	(50,)	500
99	0.3539	-104.6447	-1.7128	1	(50,)	1000
100	-2.8161	-166.5673	-29.7467	1	(100,)	500
101	0.5703	-162.665	-248.2539	1	(100,)	1000
102	0.4807	-798.7156	-279.6352	1	(150,)	500
103	-0.4329	-411.6042	-1540.3234	1	(150,)	1000
104	0.7682	-4.3824	-456.8886	1	(10, 10)	500
105	0.7779	-10.2597	-1.3945	1	(10, 10)	1000
106	0.701	-51.028	-234.9006	1	(50, 50)	500
107	0.7659	-358.3651	-3253.9739	1	(50, 50)	1000
108	0.6968	-2.9608	-587.667	1	(100, 100)	500
109	0.7481	-629.9378	-65.9016	1	(100, 100)	1000
110	0.7468	-773.6747	-4921.2635	1	(150, 150)	500
111	0.7706	-332.196	-200.9633	1	(150, 150)	1000
112	0.845	-1.8165	-4.9732	1	(10, 10, 10)	500
113	0.8439	-3.9975	-17.2314	1	(10, 10, 10)	1000
114	0.8567	-706.5803	-30.7286	1	(50, 50, 50)	500
115	0.8616	-5.9103	-120.9103	1	(50, 50, 50)	1000
116	0.8609	-49.3807	-132.0867	1	(100, 100, 100)	500
117	0.8641	-67.9346	-858.9842	1	(100, 100, 100)	1000
118	0.8639	-23.88	-39.6984	1	(150, 150, 150)	500
119	0.8641	-31.6046	-14.6052	1	(150, 150, 150)	1000

TABLE C.8: Results: Optimum Parameter Permutation Feature Importance (Rank Sum)

Rank	Feature Label	Importance	Rank Perm
0	EC LAG: Lagged Damage Index	1.150	7
1	ACTRAFF: Freight Trains	0.262	382
2	ACTRAFF: Freight Tonnes	0.187	584
3	WEAR SEG RATES: Vertical Wear Avg. Max Seg	0.034	762
4	WEAR RATES: Field Wear Max.	0.033	838
5	WEAR SEG RATES: Vertical Wear Avg. Mean Seg	0.025	931
6	ACTRAFF: Passenger Tonnes	0.034	933
7	WEAR RATES: Gauge Face Remaining Min.	0.033	1004
8	WEAR RATES: Gauge Wear Avg.	0.035	1053
9	WEAR SEG RATES: Gauge Wear Avg. Max Seg	0.231	1151
10	ACTRAFF: Passenger Trains	0.034	1152
11	EC: No Segment Measurements	0.016	1179
12	ACTRAFF: Total Tonnes	0.028	1331
13	WEAR SEG RATES: Field Wear Avg. Mean Seg	0.018	1332
14	WEAR SEG RATES: Gauge Wear Avg. Mean Seg	0.179	1384
15	TRACK: Dist. to SC	0.015	1417
16	WEAR RATES: Rail Depth Remaining Min.	0.016	1439
17	WEAR SEG RATES: Head Width Remaining Avg. Mean Seg	0.015	1556
18	WEAR SEG RATES: Field Wear Avg. Max Seg	0.013	1566
19	WEAR RATES: Gauge Face Remaining Avg.	0.018	1580
20	WEAR SEG RATES: Gauge Face Remaining Avg. Mean Seg	0.012	1585
21	WEAR SEG RATES: Rail Depth Remaining Min. Mean Seg	0.022	1586
22	WEAR RATES: Field Wear Avg.	0.023	1606
23	WEAR RATES: Head Width Avg.	0.459	1618
24	EC: Time Since First EC Meas.	0.013	1623
25	ACTRAFF: Total Trains	0.033	1701
26	TRACK: Dist. to Station	0.008	1704
27	WEAR RATES: Head Width Remaining Avg.	0.411	1825
28	WEAR SEG RATES: Rail Depth Remaining Min. Max Seg	0.013	1881
29	WEAR: Gauge Face Remaining Avg.	0.012	2036
30	Time Since Last Measurement	0.008	2114
31	TRACK: Rail High/ Low	0.005	2138
32	WEAR: Gauge Wear Avg.	0.007	2146
33	WEAR: Gauge Face Remaining Min.	0.010	2258
34	WEAR RATES: Gauge Wear Max.	0.007	2267
35	WEAR: Vertical Wear Max.	0.005	2274
36	WEAR: Head Width Avg.	0.009	2300
37	WEAR RATES: Vertical Wear Avg.	0.005	2368
38	TRACK: EMGTPA	0.004	2536
39	TRACK: Max Line Speed	0.003	2576
40	WEAR: Field Wear Avg.	0.005	2609

Continued on next page

Rank	Feature Label	Importance	Rank Perm
41	WEAR: Gauge Wear Max.	0.004	2662
42	TRACK: Dist. to Tunnel	0.002	2689
43	EC: Linear Distance	0.004	2692
44	WEAR RATES: Vertical Wear Max.	0.003	2714
45	WEAR: Field Wear Max.	0.003	2764
46	WEAR: Head Width Remaining Avg.	0.002	2770
47	TRACK: Cant	0.003	2780
48	TRACK: Curvature	0.003	2838
49	WEAR: Vertical Wear Avg.	0.002	2968
50	TRACK: Track Category	0.002	3058
51	TRACK: Tunnel	0.001	3068
52	WEAR SEG RATES: Gauge Face Remaining Avg. Max Seg	0.002	3109
53	WEAR: Rail Depth Remaining Min.	0.002	3154
54	WEAR SEG RATES: Head Width Remaining Avg. Max Seg	0.003	3258
55	TRACK: Rail Side	0.002	3306
56	TRACK: Station	0.000	3558

TABLE C.9: Results: Optimum Parameter Permutation Feature Importance (Rank Sum)

Rank	Feature Label	Importance	Rank Perm
0	EC LAG: Lagged Damage Index	1.150286144	7
1	WEAR RATES: Head Width Avg.	0.459241017	1618
2	WEAR RATES: Head Width Remaining Avg.	0.41063942	1825
3	ACTRAFF: Freight Trains	0.262325291	382
4	WEAR SEG RATES: Gauge Wear Avg. Max Seg	0.230935483	1151
5	ACTRAFF: Freight Tonnes	0.18671328	584
6	WEAR SEG RATES: Gauge Wear Avg. Mean Seg	0.178563982	1384
7	WEAR RATES: Gauge Wear Avg.	0.035241281	1053
8	ACTRAFF: Passenger Tonnes	0.034041771	933
9	ACTRAFF: Passenger Trains	0.033865358	1152
10	WEAR SEG RATES: Vertical Wear Avg. Max Seg	0.033776004	762
11	WEAR RATES: Gauge Face Remaining Min.	0.033175556	1004
12	ACTRAFF: Total Trains	0.032964674	1701
13	WEAR RATES: Field Wear Max.	0.032604228	838
14	ACTRAFF: Total Tonnes	0.028270902	1331
15	WEAR SEG RATES: Vertical Wear Avg. Mean Seg	0.025332224	931
16	WEAR RATES: Field Wear Avg.	0.022567379	1606
17	WEAR SEG RATES: Rail Depth Remaining Min. Mean Seg	0.021675632	1586
18	WEAR SEG RATES: Field Wear Avg. Mean Seg	0.017611201	1332
19	WEAR RATES: Gauge Face Remaining Avg.	0.017500694	1580
20	EC: No Segment Measurements	0.016002492	1179
21	WEAR RATES: Rail Depth Remaining Min.	0.015930605	1439

Continued on next page

Rank	Feature Label	Importance	Rank Perm
22	WEAR SEG RATES: Head Width Remaining Avg. Mean Seg	0.015183835	1556
23	TRACK: Dist. to SC	0.014960523	1417
24	WEAR SEG RATES: Rail Depth Remaining Min. Max Seg	0.013095835	1881
25	EC: Time Since First EC Meas.	0.013086222	1623
26	WEAR SEG RATES: Field Wear Avg. Max Seg	0.013024735	1566
27	WEAR SEG RATES: Gauge Face Remaining Avg. Mean Seg	0.011862748	1585
28	WEAR: Gauge Face Remaining Avg.	0.011645132	2036
29	WEAR: Gauge Face Remaining Min.	0.010038739	2258
30	WEAR: Head Width Avg.	0.00861301	2300
31	Time Since Last Measurement	0.00834261	2114
32	TRACK: Dist. to Station	0.0081018	1704
33	WEAR RATES: Gauge Wear Max.	0.007496382	2267
34	WEAR: Gauge Wear Avg.	0.006849703	2146
35	TRACK: Rail High/ Low	0.005498174	2138
36	WEAR: Vertical Wear Max.	0.004819856	2274
37	WEAR: Field Wear Avg.	0.004818404	2609
38	WEAR RATES: Vertical Wear Avg.	0.0048003	2368
39	TRACK: EMGTPA	0.004046726	2536
40	WEAR: Gauge Wear Max.	0.003663432	2662
41	EC: Linear Distance	0.003621293	2692
42	WEAR RATES: Vertical Wear Max.	0.00331394	2714
43	WEAR: Field Wear Max.	0.003212358	2764
44	TRACK: Max Line Speed	0.003168275	2576
45	TRACK: Cant	0.003156201	2780
46	TRACK: Curvature	0.002952025	2838
47	WEAR SEG RATES: Head Width Remaining Avg. Max Seg	0.002595615	3258
48	WEAR: Head Width Remaining Avg.	0.002441929	2770
49	TRACK: Dist. to Tunnel	0.002355197	2689
50	TRACK: Track Category	0.002255606	3058
51	WEAR: Vertical Wear Avg.	0.002211824	2968
52	WEAR SEG RATES: Gauge Face Remaining Avg. Max Seg	0.001810954	3109
53	WEAR: Rail Depth Remaining Min.	0.001709049	3154
54	TRACK: Rail Side	0.001515992	3306
55	TRACK: Tunnel	0.001011634	3068
56	TRACK: Station	0.000311488	3558

C.3 Final Model Comparison with Track-Ex

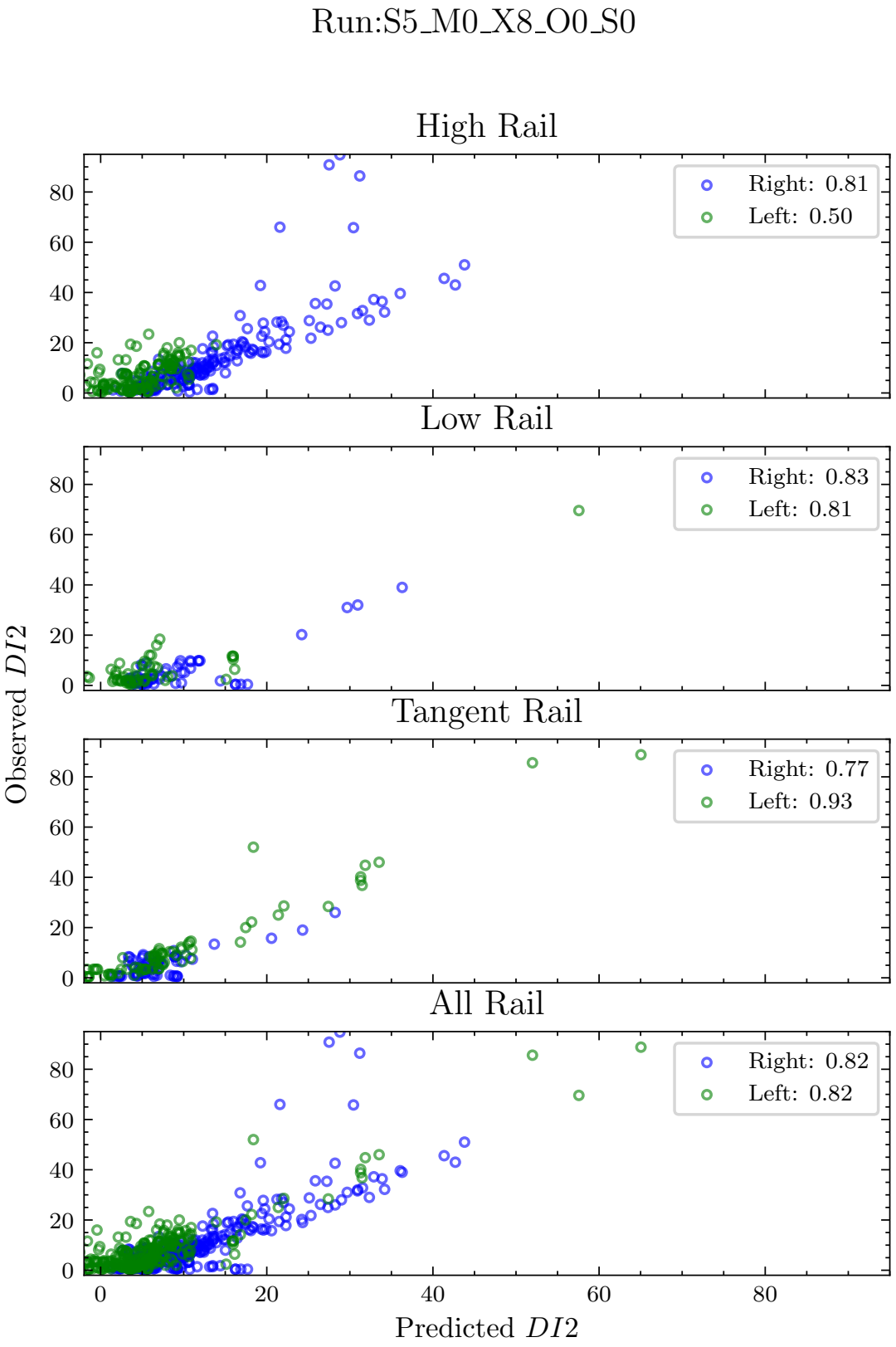


FIGURE C.1: Track-Ex Comparison: Linear Regression

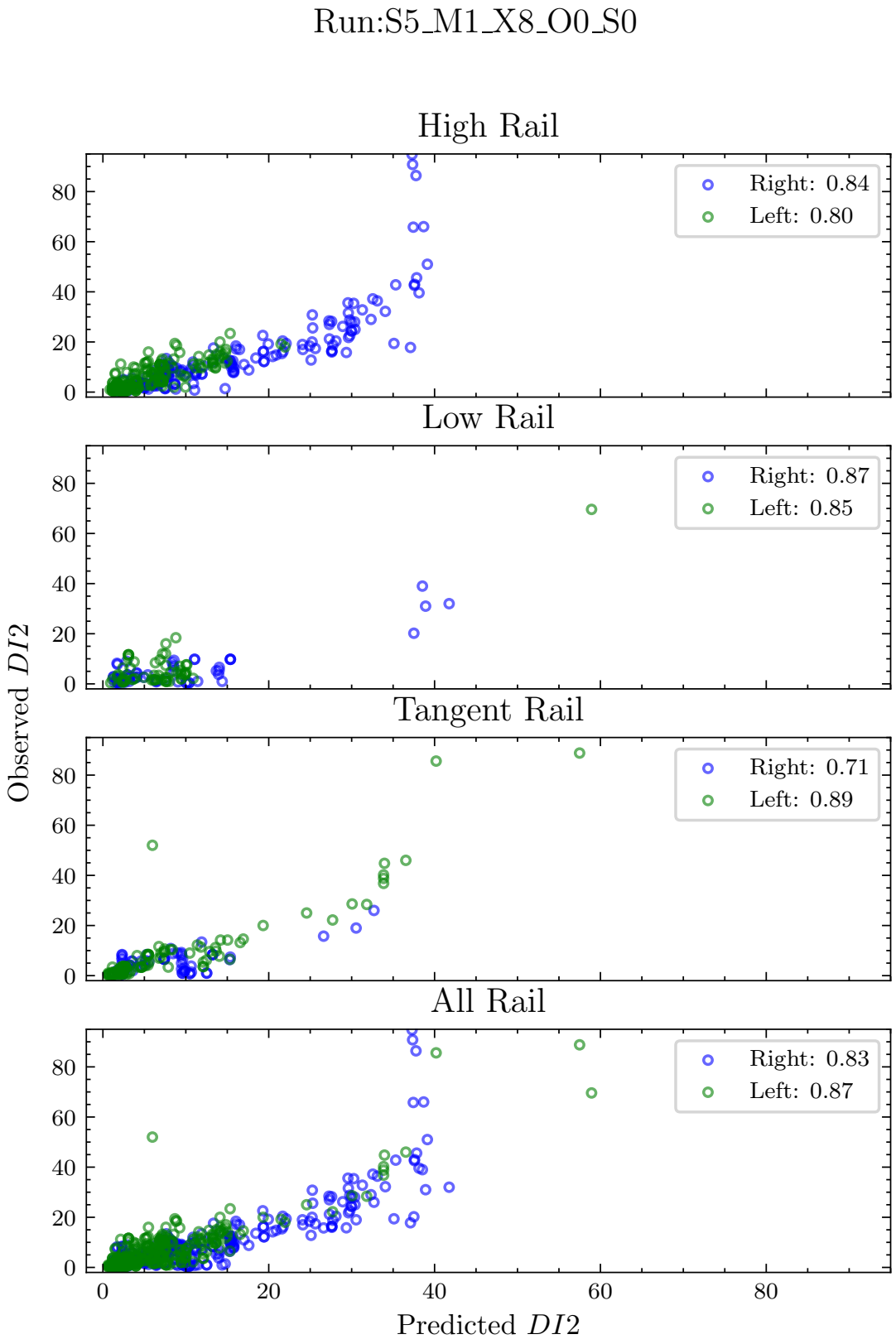


FIGURE C.2: Track-Ex Comparison: RF

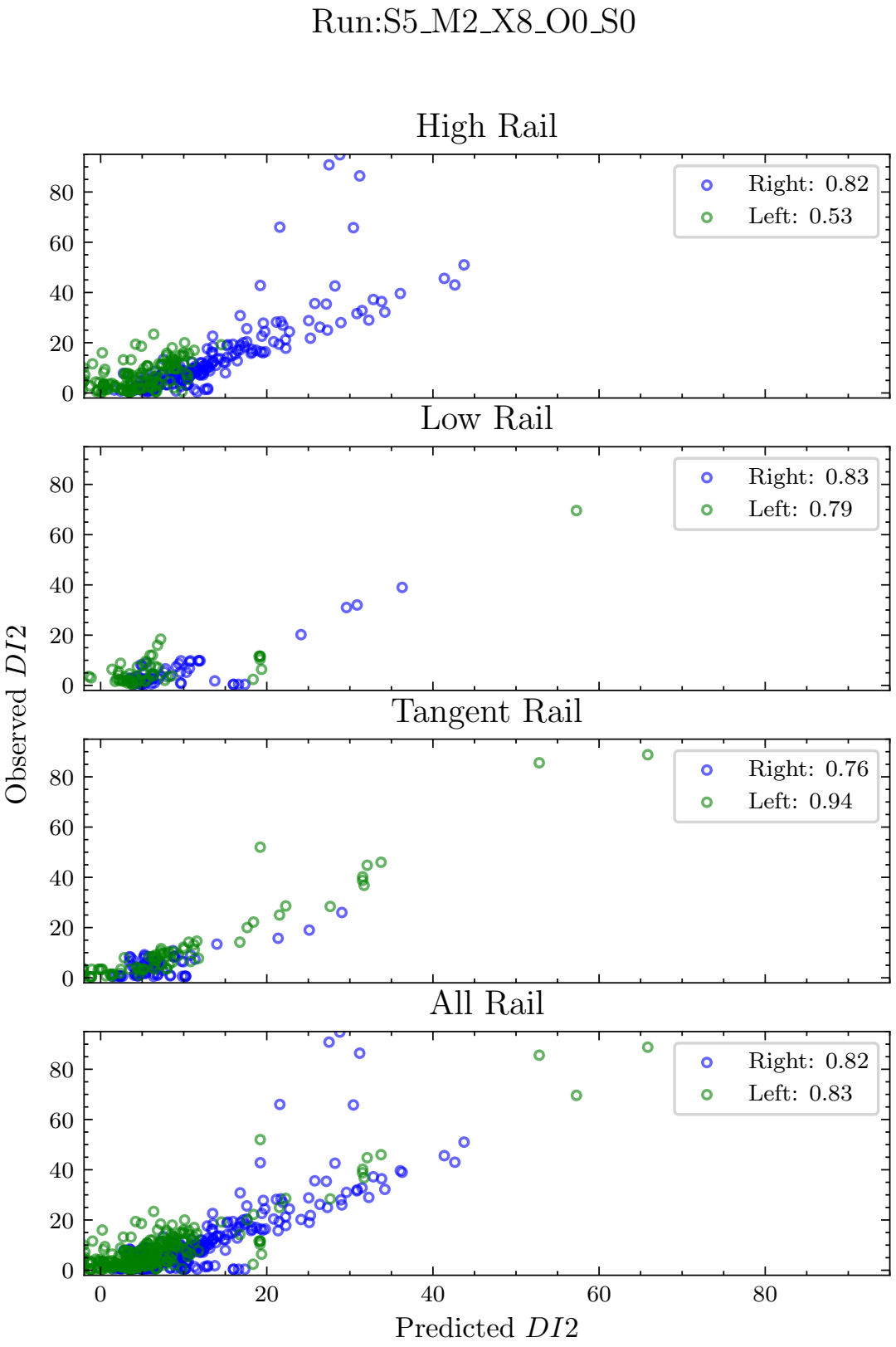


FIGURE C.3: Track-Ex Comparison: SVR Lin

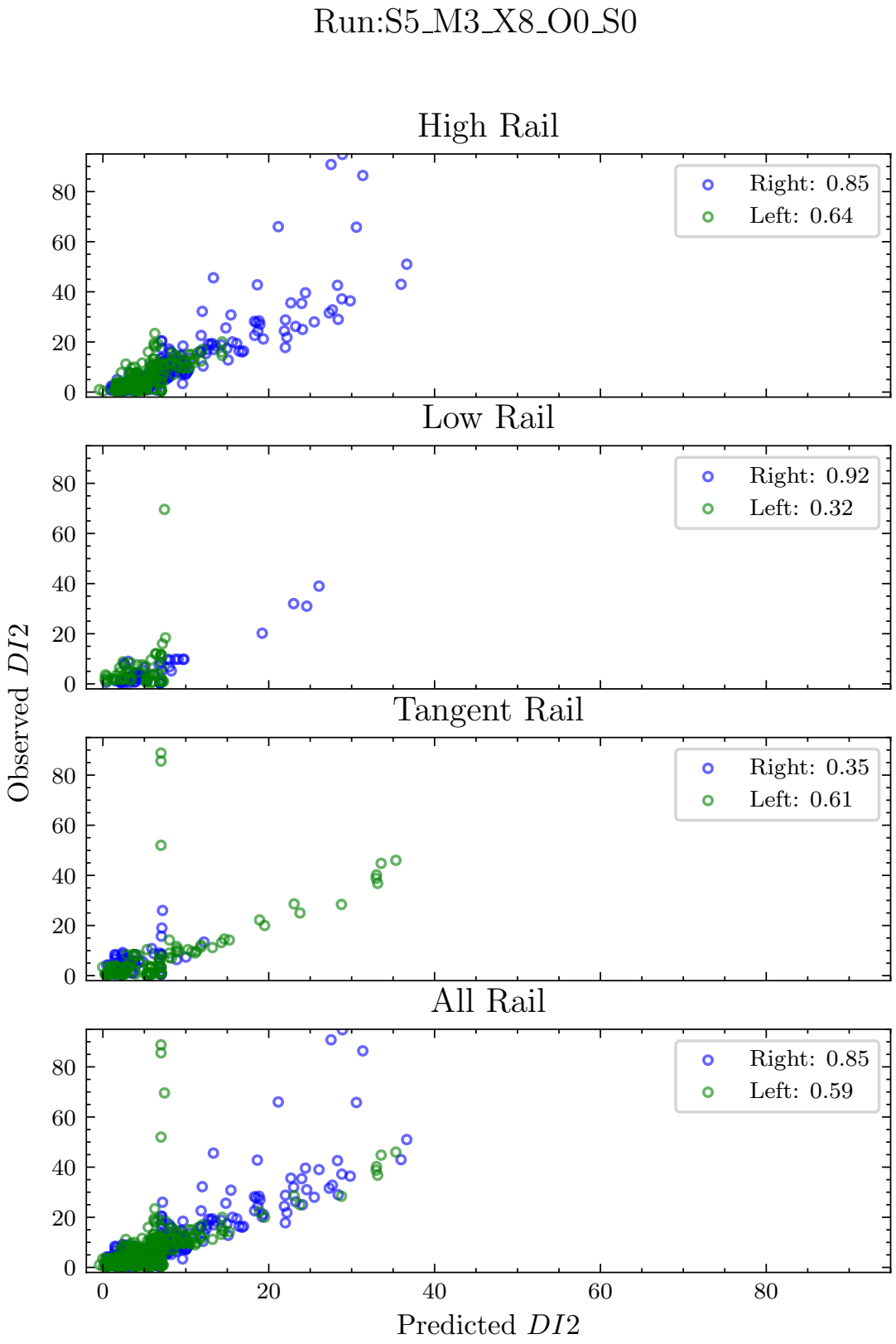


FIGURE C.4: Track-Ex Comparison: SVR Gaussian

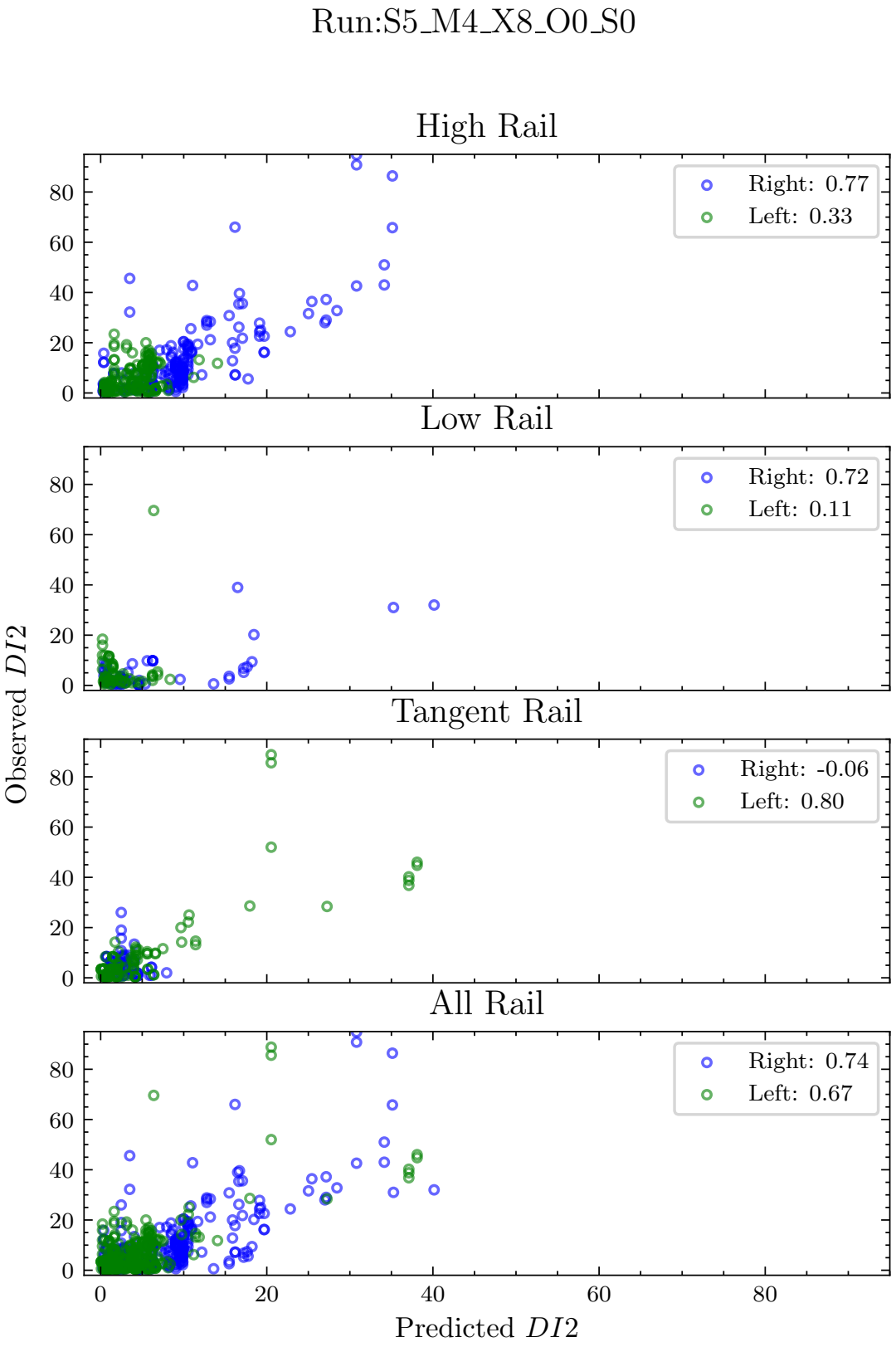


FIGURE C.5: Track-Ex Comparison: KNN 1

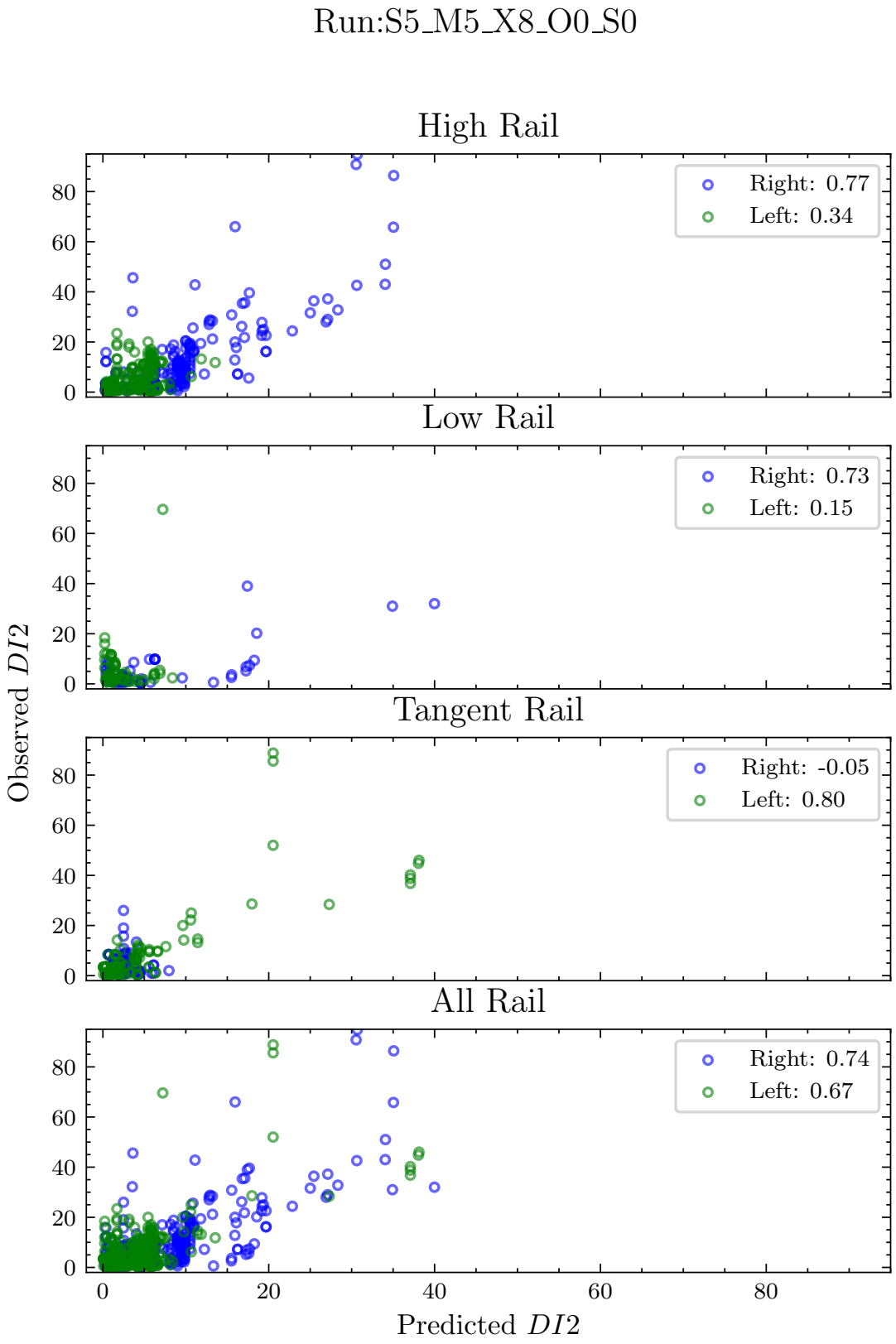


FIGURE C.6: Track-Ex Comparison: KNN 2

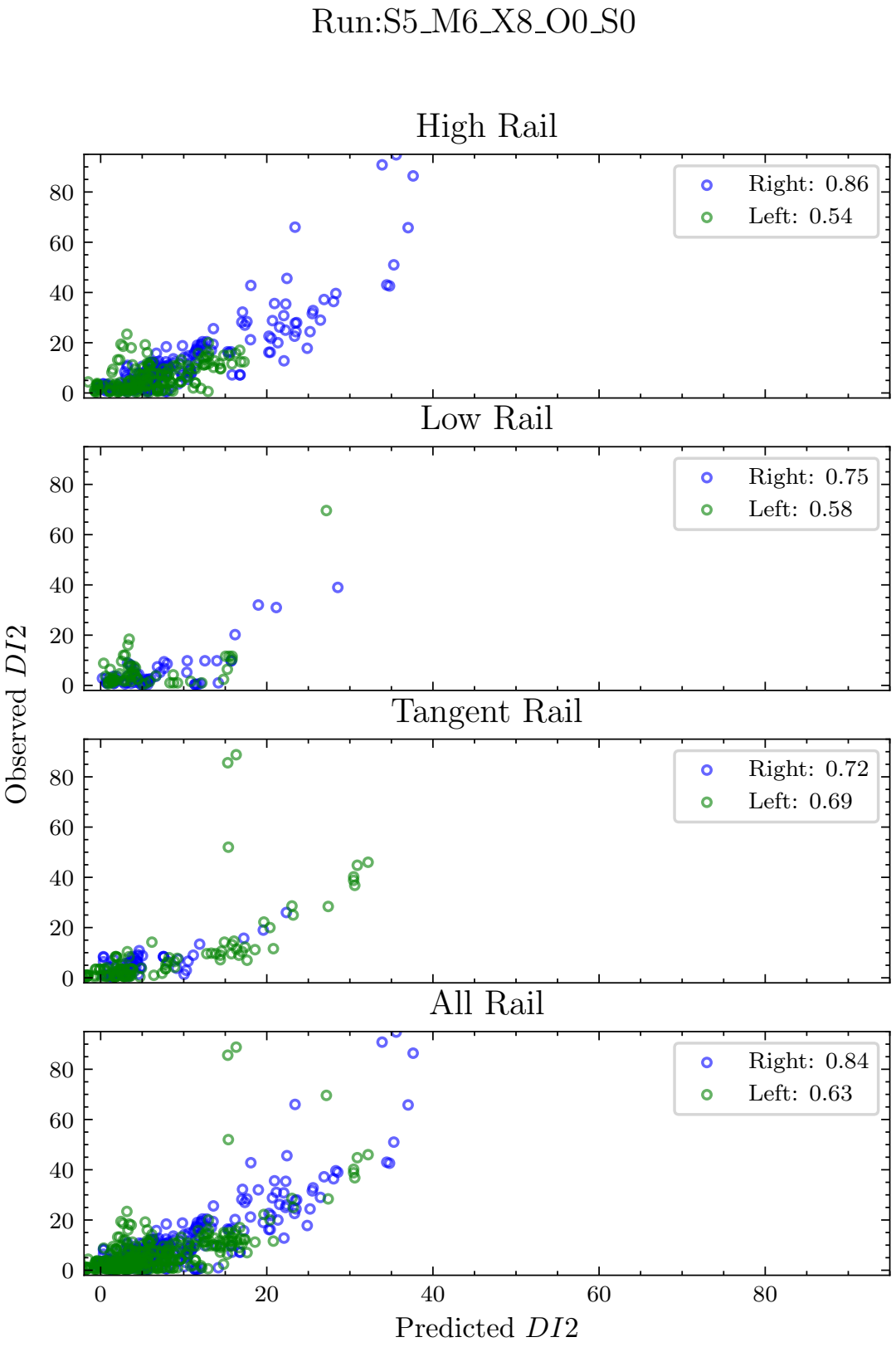


FIGURE C.7: Track-Ex Comparison: MLP

References

Aha, D. W. (1997). Editorial, *in* D. W. Aha (ed.), *Lazy Learning*, Springer Netherlands, Dordrecht, pp. 7–10.

Al-Douri, Y. K., Tretten, P. and Karim, R. (2016). Improvement of railway performance: a study of Swedish railway infrastructure, *Journal of Modern Transportation* **24**(1): 22–37.

Alasadi, S. (2017). Review of Data Preprocessing Techniques in Data Mining, *Journal of Engineering and Applied Sciences* **12**: 4102–4107.

Alloghani, M., Al-Jumeily, D., Mustafina, J., Hussain, A. and Aljaaf, A. J. (2020). A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science, *in* M. W. Berry, A. Mohamed and B. W. Yap (eds), *Supervised and Unsupervised Learning for Data Science*, Unsupervised and Semi-Supervised Learning, Springer International Publishing, Cham, pp. 3–21.

Alzubi, J., Nayyar, A. and Kumar, A. (2018). Machine Learning from Theory to Algorithms: An Overview, *Journal of Physics: Conference Series* **1142**(1): 012012. Publisher: IOP Publishing. **URL:** *https://dx.doi.org/10.1088/1742-6596/1142/1/012012*

Amari, S., McLaughlin, L. and Hoang Pham (2006). Cost-effective condition-based maintenance using markov decision processes, *RAMS '06. Annual Reliability and Maintainability Symposium, 2006.*, IEEE, Newport Beach, CA, USA, pp. 464–469.

Amato, F., Lombardo, L., Tonini, M., Marvuglia, A., Castro-Camilo, D. and Guignard, F. (2022). Spatiotemporal data science: theoretical advances and applications, *Stochastic Environmental Research and Risk Assessment* **36**(8): 2027–2029.

Arlot, Sylvain & Celisse, A. (2009). A survey of cross validation procedures for model selection, *Statistics Surveys* **4**.

Atluri, G., Karpatne, A. and Kumar, V. (2017). Spatio-Temporal Data Mining: A Survey of Problems and Methods. arXiv: 1711.04710.

Attoh-Okine, D. N. (2016). How Track Geometry Defects Affect the Development of Rail Defects, p. 16.

Ben-Gera, T., He, J. and Liu, X. (2016). A Preliminary Methodology for Broken Rail Caused Freight Train Derailment Risk Analysis, *2016 Joint Rail Conference*, American Society of Mechanical Engineers, Columbia, South Carolina, USA, p. V001T06A006.

Berman, J. J. (2016). Chapter 4 - Understanding Your Data, *in* J. J. Berman (ed.), *Data Simplification*, Morgan Kaufmann, Boston, pp. 135–187.

Bevan, A., Molyneux-Berry, P., Eickhoff, B. and Burstow, M. (2013). Development and validation of a wheel wear and rolling contact fatigue damage model, *Wear* **307**(1-2): 100–111.

Bevan, P. B. . A. (2020). Prediction of rail damage using a combination of shakedown map and wheel-rail contact energy, *Wear* **460-461**: 203457.
URL: *https://www.sciencedirect.com/science/article/pii/S0043164820309169*

Bhagi, P. C. (2011). Eddy Current Testing: Basics, *Journal of Non-Destructive Testing & Evaluation* **10**: 7–16.

Bogdański, S., Stupnicki, J., Brown, M. W. and Cannon, D. F. (1999). A Two Dimensional Analysis of Mixed-Mode Rolling Contact Fatigue Crack Growth in Rails, *European Structural Integrity Society*, Vol. 25 of *Multiaxial Fatigue and Fracture*, Elsevier, pp. 235–248.

Bombarda, D., Vitetta, G. M. and Ferrante, G. (2021). Rail Diagnostics Based on Ultrasonic Guided Waves: An Overview, *Applied Sciences* **11**(3): 1071.

Boyacioglu, P., Bevan, A. and Vickerstaff, A. (2018). Use of NDT Inspection Data to Improve Rail Damage Prediction Models, *IET Conference Proceedings* pp. –(1). Publisher: Institution of Engineering and Technology.
URL: *https://digital-library.theiet.org/content/conferences/10.1049/cp.2018.0048*

Boyer, H. E. (1986). Atlas of Fatigue Curves.
URL: *https://www.osti.gov/biblio/5500725*

Breiman, L. (1984). *Classification and Regression Trees*, Routledge.

Breiman, L. (2001). Random Forests, *Machine Learning* **45**(1): 5–32.

Breunig, M. (2000). LOF: Identifying Density-Based Local Outliers.

British Standards Institution (2018). BS EN 13231-5:2018 Railway applications - Track - Acceptance of works - Part 5: Procedures for rail reprofiling in plain line, switches, crossings and expansion devices.

Brunton, Steven L. & Kutz, J. N. (2019). *Regression and Model Selection*, Cambridge University Press, p. 117–153.

Burman, P., Chow, E. and Nolan, D. (1994). A Cross-Validatory Method for Dependent Data, *Biometrika* **81**: 351–358.

Burstow, M. (2003). Whole Life Rail Model: Development of a Rolling Contact Fatigue Damage Parameter, *Technical report*.

Burstow, M. (2013). A comprehensive track damage management tool.

Burstow, M. C. (2004). Whole Life Rail Model Application and Development for RSSB – Continued Development of an RCF Damage Parameter, *Technical Report AEATR-ES-2004-880 Issue 2*.

Burstow, M. C., Watson, A. S., Beagles, M. and Park, P. (2009). Simulation of Rail Wear and Rolling Contact Fatigue Using the Whole Life Rail Model.

Campesato, O. (2020). *Artificial intelligence, machine learning, and deep learning*, Mercury Learning and Information, Dulles, Virginia.

Cannon, D. F., Edel, K., Grassie, S. L. and Sawley, K. (2003). Rail defects: an overview, *Fatigue & Fracture of Engineering Materials & Structures* **26**(10): 865–886.

Cao Feng, Yong Ge and Jinfeng Wang (2014). Spatial data discretization methods for geocomputation.

Cao, L.J. & Tay, F. (2003). Support vector machine with adaptive parameters in financial time series forecasting, *IEEE Transactions on Neural Networks* **14**(6).

Caruana, Rich & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms, *Proceedings of the 23rd international conference on Machine learning*, ICML '06, Association for Computing Machinery, New York, NY, USA, pp. 161–168.

Chandola, Varun & Kumar, V. (2009). Outlier Detection : A Survey, *ACM Computing Surveys* **41**.

Chattopadhyay, G & Kumar, S. (2009). Parameter Estimation for Rail Degradation Model, *International Journal of Performability Engineering* **5**(2): 119–130.

Cios, K. J., Pedrycz, W. and Swiniarski, R. W. (1998). Data Mining Methods For Knowledge Discovery.

Claesen, Marc & De Moor, B. (2015). Hyperparameter Search in Machine Learning, *arXiv:1502.02127 [cs, stat]* . arXiv: 1502.02127.

Constantin (2015). Cant and Cant Deficiency. Where is the 11.82 coming from?
URL: *https://pwayblog.com/2015/10/29/11-82_cant-deficiency-un-compensated-acceleration-pway/*

Cope, G.H. & Institution, P. W. (1993). *British Railway Track: Design, Construction and Maintenance*, Permanent Way Institution.
URL: *https://books.google.co.uk/books?id=JttBPgAACAAJ*

Dembowsky, M. A. & Burstow, M. C. (2013). Track-Ex: A Comprehensive Track-Damage Management Tool.

Department for Transport (2021). Network Rail Factsheet 2020, *Technical report*.

Dey, A. (2016). Machine Learning Algorithms: A Review, **7**.

Dick, C. T., Barkan, C. P. L., Chapman, E. R. and Stehly, M. P. (2003). Multivariate Statistical Model for Predicting Occurrence and Location of Broken Rails, *Transportation Research Record: Journal of the Transportation Research Board* **1825**(1): 48–55.

Dirks, B., Enblom, R., Ekberg, A. and Berg, M. (2015). The development of a crack propagation model for railway wheels and rails, *Fatigue & Fracture of Engineering Materials & Structures* **38**(12): 1478–1491.

Dollevoet, R. (2010). Design of an anti head check profile based on stress relief.

Domingos, P. (2012). A few useful things to know about machine learning, *Communications of the ACM* **55**(10): 78–87.

Efron, B & Tibshirani, R. (1994). *An Introduction to the Bootstrap*, Chapman & Hall/CRC.

Ekberg, A., Kabo, E. and Andersson, H. (2002). An engineering model for prediction of rolling contact fatigue of railway wheels, *Fatigue Fracture of Engineering Materials and Structures* **25**(10): 899–909.

Ekberg, A., Åkesson, B. and Kabo, E. (2014). Wheel/rail rolling contact fatigue – Probe, predict, prevent, *Wear* **314**(1): 2–12.

Evans, J. & Iwnicki, S. (2002). Vehicle dynamics and the wheel/rail interface. **URL:** *https://www.semanticscholar.org/paper/Vehicle-dynamics-and-the-wheel-rail-interface-Evans-Iwnicki/cf53399efd043bbfefa9756dbfd95961910dd0ae*

Fletcher, D., Franklin, F. and Kapoor, A. (2009). Rail surface fatigue and wear, *Wheel–Rail Interface Handbook*, Elsevier, pp. 280–310.

Fumeo, E., Anguita, D. and Oneto, L. (2015). Condition Based Maintenance in Railway Transportation Systems Based on Big Data Streaming Analysis, *Procedia Computer Science* .

Gao, S., Szugs, T. and Ahlbrink, R. (2018). Use of Combined Railway Inspection Data Sources for Characterization of Rolling Contact Fatigue, p. 7.

Grassie, S. L. (2005). Rolling Contact Fatigue on the British Railway System: Treatment, *Wear* **258**(7): 1310–1318.

Gu, K. K., Lin, Q., Wang, W. J., Wang, H. Y., Guo, J., Liu, Q. Y. and Zhu, M. H. (2015). Analysis on the effects of rotational speed of grinding stone on removal behavior of rail material, *Wear* **342-343**: 52–59.

Guler, H. (2014). Prediction of railway track geometry deterioration using artificial neural networks: a case study for Turkish state railways.

Hamdi, A., Shaban, K., Erradi, A., Mohamed, A., Rumi, S. K. and Salim, F. D. (2021). Spatiotemporal data mining: a survey on challenges and open problems, *Artificial Intelligence Review* pp. 1–48.

Harris, R., Kalousek, J., Sroba, P., Caldwell, R. and Nuorala, M. (2011). Predictive Preventive Grinding Model to Control Rolling Contact Fatigue in Rails is Tested on BNSF Railway, *Proceedings IHHA Conference*, Calgary, Canada.

Hart, P. (1968). The condensed nearest neighbor rule (Corresp.), *IEEE Transactions on Information Theory* **14**(3): 515–516. Conference Name: IEEE Transactions on Information Theory.

Hassan, H., Negm, A., Zahran, M. and Saavedra, O. (2015). Assessment of artifical neural network for bathymetry estimation using high resolution satellite imagery in shallow lakes: Case study el burullus lake., *International Water Technology Journal* **5**.

Hastie, T., Tibshirani, R. and Friedman, J. (2009). Random Forests, *The Elements of Statistical Learning*, Springer New York, New York, NY, pp. 587–604.

Haykin, S. (1998). *Neural Networks: A Comprehensive Foundation*, 2nd edn, Prentice Hall PTR, USA.

Haykin, S. (2000). CHAPTER 4 - Neural Networks: A Guided Tour, *in* N. K. Sinha and M. M. Gupta (eds), *Soft Computing and Intelligent Systems*, Academic Press Series in Engineering, Academic Press, San Diego, pp. 71–80.

Heyder, René & Brehmer, M. (2014). Empirical studies of head check propagation on the DB network, *Wear* **314**(1): 36–43.

Hokstad, P., Langseth, H., Lindqvist, B. H. and Vatn, J. (2005). Failure modeling and maintenance optimization for a railway line, p. 14.

Huang, J.-C., Ko, K.-M., Shu, M.-H. and Hsu, B.-M. (2020). Application and comparison of several machine learning algorithms and their integration models in regression problems, *Neural Computing and Applications* **32**: 5461–5469.

Hwang, B., Hillmann, S., Schulze, M., Klein, M. and Heuer, H. (2015). Eddy Current Imaging for Electrical Characterization of Silicon Solar Cells and TCO layers, p. 10.

Imandoust, S.B. & Bolandraftar, M. (2013). Application of K-nearest neighbor (KNN) approach for predicting economic events theoretical background, *Int J Eng Res Appl* **3**: 605–610.

InnoTrack (2008). D4.4.1 – Rail Inspection Technologies.

Jamshidi, A., Faghih-Roohi, S., Hajizadeh, S., Núñez, A., Babuska, R., Dollevoet, R., Li, Z. and De Schutter, B. (2017). A Big Data Analysis Approach for Rail Failure Risk Assessment: Big Data Analysis Approach For Rail Failure Risk Assessment, *Risk Analysis* **37**(8): 1495–1507.

Jamshidi, A., Faghih Roohi, S., Núñez, A., Babuska, R., De Schutter, B., Dollevoet, R. and Li, Z. (2016). Probabilistic Defect-Based Risk Assessment Approach for Rail Failures in Railway Infrastructure, *IFAC-PapersOnLine* **49**(3): 73–77.

Jamshidi, A., Hajizadeh, S., Su, Z., Naeimi, M., Núñez, A., Dollevoet, R., De Schutter, B. and Li, Z. (2018). A decision support approach for condition-based maintenance of rails based on big data analysis, *Transportation Research Part C: Emerging Technologies* **95**: 185–206.

Jiang, T., Gradus, J. L. and Rosellini, A. J. (2020). Supervised Machine Learning: A Brief Primer, *Behavior Therapy* **51**(5): 675–687.

Jianxi, W., Yude, X., Songliang, L. and Liying, W. (2011). Probabilistic prediction model for initiation of RCF cracks in heavy-haul railway, *International Journal of Fatigue* **33**(2): 212–216.

Johnson, K. L. (1985). Contact Mechanics. ISBN: 9780521255769 9780521347969 9781139171731 Publisher: Cambridge University Press.

Kalousek, J. & Magel, E. (1997). Achieving a balance: The "magic" wear rate, **93**: 50–X4.

Kalousek, J., Sroba, P. and Hegelund, C. (1989). Analysis of Rail Grinding Tests and Implications for Corrective and Preventative Grinding, *Fourth International Heavy Haul Railway Conference 1989: Railways in Action; Preprints of Papers, The* p. 193. Publisher: Institution of Engineers, Australia.

Kapoor, A., Fletcher, D., Franklin, F. and Alwahdi, F. (2002). University of Sheffield and AEAT Whole Life Rail Model Interim Report, *Technical Report MEC/AK/AEAT/September02/*.

Kecman, V. (2001). *Support Vector Machines*, pp. 121–191.

Kecman, V. (2005). Support Vector Machines – An Introduction, *Support Vector Machines: Theory and Applications*, Vol. 177, pp. 605–605.

Korifi, R., Le Dréau, Y. and Dupuy, N. (2014). Comparative study of the alignment method on experimental and simulated chromatographic data, *Journal of Separation Science* **37**: 3276–3291. Publisher: Wiley-VCH Verlag.

Kowalski, Paweł & Smyk, R. (2018). Review and comparison of smoothing algorithms for one-dimensional data noise reduction, *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, pp. 277–281.

Krishna, V. V., Hossein-Nia, S., Casanueva, C., Stichel, S., Trummer, G. and Six, K. (2021). Rail RCF damage quantification and comparison for different damage models, *Railway Engineering Science* .

Kubin, W., Daves, W. and Stock, R. (2019). Analysis of rail milling as a rail maintenance process: Simulations and experiments, *Wear* **438-439**: 203029.

Kulkarni, S. S., Sun, L., Moran, B., Krishnaswamy, S. and Achenbach, J. D. (2006). A Probabilistic Method to Predict Fatigue Crack Initiation, *International Journal of Fracture* **137**(1-4): 9–17.

Kumar, S. (2006). *A Study of the Rail Degradation Process to Predict Rail Breaks*, PhD thesis, Lulea University of Technology.

Kutner, M., Nachtsheim, C. and Neter, J. (2004). *Applied Linear Regression Models*, Irwin/McGraw-Hill series in operations and decision sciences, McGraw-Hill/Irwin.

Lazarevic, Aleksandar & Kumar, V. (2005). Feature bagging for outlier detection, *Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining - KDD '05*, ACM Press, Chicago, Illinois, USA, p. 157.

Lechevalier, D., Narayanan, A. and Rachuri, S. (2014). Towards a domain-specific framework for predictive analytics in manufacturing, *2014 IEEE International Conference on Big Data (Big Data)*, pp. 987–995.

Lewis, R & Olofsson, U. (ed.) (2009). *Wheel-Rail Interface Handbook*, Woodhead Publishing.

Li, C. (2019). Preprocessing Methods and Pipelines of Data Mining: An Overview, *arXiv:1906.08510 [cs, stat]* . arXiv: 1906.08510.

Lin, Chang-Ching & Tseng, H.-Y. (2005). A neural network application for reliability modelling and condition-based predictive maintenance, *The International Journal of Advanced Manufacturing Technology* **25**(1-2): 174–179.

Liu, H., Hussain, F., Tan, C. and Dash, M. (2002). Discretization: An enabling technique, *Data Mining and Knowledge Discovery* **6**(4): 393–423.

Lo, J. T.-H., Gui, Y. and Peng, Y. (2012). Overcoming the Local-Minimum Problem in Training Multilayer Perceptrons with the NRAE Training Method, *in* D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, J. Wang, G. G. Yen and M. M. Polycarpou (eds), *Advances in Neural Networks – ISNN 2012*, Vol. 7367, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 440–447. Series Title: Lecture Notes in Computer Science.

Lopes Gerum, P. C., Altay, A. and Baykal-Gürsoy, M. (2019). Data-driven predictive maintenance scheduling policies for railways, *Transportation Research Part C: Emerging Technologies* **107**: 137–154.

Magel, E. E. (2011). Rolling contact fatigue: a comprehensive review, *Technical report*, U.S. Department of Transportation. Federal Railroad Administration.

Magel, E. E., Sawley, K. J. and Sroba, P. S. (2005). A Practical Approach to Controlling Rolling Contact Fatigue in Railways.

Magel, E., Kalousek, J. and Sroba, P. (2014). Chasing the Magic Wear Rate, Ajaccio, Corsica, France, p. 116.

Magel, E., Roney, M., Kalousek, J. and Sroba, P. (2003). The blending of theory and practice in modern rail grinding, *Fatigue & Fracture of Engineering Materials & Structures* **26**(10): 921–929.

Magel, E., Sroba, P., Sawley, K. and Kalousek, J. (2004). Control of Rolling Contact Fatigue of Rails, p. 29.

Maleki, F., Muthukrishnan, N., Ovens, K., Reinhold, C. and Forghani, R. (2020). Machine Learning Algorithm Validation: From Essentials to Advanced Applications and Implications for Regulatory Certification and Deployment, *Neuroimaging Clinics* **30**(4): 433–445. **URL:** *https://www.neuroimaging.theclinics.com/article/S1052-5149(20)30059-9/fulltext*

Marshland, S. (2014). *Machine Learning: An Algorithmic Perspective*, CRC press.

Merkert, R. (2005). The restructuring and future of the British Rail system, *White Paper 586*, University of Leeds.

Miner, M. A. (1945). Cumulative Damage in Fatigue, *Journal of Applied Mechanics* **12**(3): A159–A164.

Mitchell, T. M. (1999). Machine learning and data mining, *Communications of the ACM* **42**(11): 30–36.

Moridpour, S. (2016). Application of artificial neural networks in predicting the degradation of tram tracks using maintenance data | Semantic Scholar.

Müller, M. (2007). Dynamic time warping, *Information Retrieval for Music and Motion* **2**: 69–84.

Nargesian, F., Samulowitz, H., Khurana, U., Khalil, E. B. and Turaga, D. (2017). Learning Feature Engineering for Classification, *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, International Joint Conferences on Artificial Intelligence Organization, Melbourne, Australia, pp. 2529–2535.

Network Rail (2018a). Management of rail defects, Module 07, *Standard NR/L2/TRK/001/mod07*.

Network Rail (2018b). Rail Profile Management, Module 10, *Standard NR/L2/TRK/001/mod10*.

Network Rail (2018c). Visual inspection, ultrasonic and eddy current testing of rails: Module 06, *Technical Report NR/L2/TRK/001/mod06*.

Network Rail (2021a). Introduction to RCF Prediction.

Network Rail (2021b). NRIL Regulatory Financial Statements: for the year ended 31 March 2021, *Technical report*.

Network Rail (2022). Network Rail Factsheet 2021, *Technical report*.

Ng, T. M. . A. (2007). CS229 lecture notes.

ORR (2006). Train Derailment at Hatfield: A Final Report by the Independent Investigation Board, *Technical report*.

Palese, Joseph W & Wright, T. W. (2000). Risk Based Ultrasonic Rail Test Scheduling on Burlington Northern Santa Fe, p. 35.

Papaelias, M., Roberts, C. and Davis, C. (2008). A review on non-destructive evaluation of rails: State-of-the-art and future development, *Proceedings of The Institution of Mechanical Engineers Part F-journal of Rail and Rapid Transit - PROC INST MECH ENG F-J RAIL R* **222**: 367–384.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python, *Journal of Machine Learning Research* **12**: 2825–2830.

Peterson, M. L. (2000). Assessment of rail flaw inspection data, *AIP Conference Proceedings*, Vol. 509, AIP, Montreal (Canada), pp. 789–796. ISSN: 0094243X.

Pletz, M., Daves, W., Yao, W., Kubin, W. and Scheriau, S. (2014). Multi-scale finite element modeling to describe rolling contact fatigue in a wheel–rail test rig, *Tribology International* **80**: 147–155.

Ponter, A. R. S., Hearle, A. D. and Johnson, K. L. (1985). Application of the kinematical shakedown theorem to rolling and sliding point contacts, *Journal of the Mechanics and Physics of Solids* **33**(4): 339–362.

Popović, Z., Mićić, M. and Lazarević, L. (2022). Guidelines for rail reprofiling, *Transportation Research Procedia* **63**: 2562–2570.

Potas, J. R., de Castro, N. G., Maddess, T. and de Souza, M. N. (2015). Waveform Similarity Analysis: A Simple Template Comparing Approach for Detecting and Quantifying Noisy Evoked Compound Action Potentials, *PLoS ONE* **10**(9).

Rajbahadur, G. K., Wang, S., Kamei, Y. and Hassan, A. E. (2021). Impact of Discretization Noise of the Dependent Variable on Machine Learning Classifiers in Software Engineering, *IEEE Transactions on Software Engineering* **47**(7): 1414–1430. Conference Name: IEEE Transactions on Software Engineering.

Rodríguez-Arana, B., San Emeterio, A., Alvarado, U., Martínez-Esnaola, J. M. and Nieto, J. (2021). Prediction of Rolling Contact Fatigue Behavior in Rails Using Crack Initiation and Growth Models along with Multibody Simulations, *Applied Sciences* **11**(3): 1026.

Rumelhart, David E. & McClelland, J. L. (1987). Learning Internal Representations by Error Propagation, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*, MIT Press, pp. 318–362. Conference Name: Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations.

Rutkowski, L., Jaworski, M., Pietruczuk, L. and Duda, P. (2014). The CART decision tree for mining data streams, *Information Sciences* **266**: 1–15.

Schafer, Darwin. H & Barkan, C. P. L. (2008). A Prediction Model for Broken Rails and an Analysis of their Economic Impact, p. 32.

Schaffer, C. (1993). Selecting a classification method by cross-validation, *Machine Learning* **13**(1): 135–143.

Schmid, F. (ed.) (2010). *Wheel-Rail Best Practice Handbook*, 1st edn, University of Birmingham Press.

Shahzamanian Sichani, M. (2016). *On Efficient Modelling of Wheel-Rail Contact in Vehicle Dynamics Simulation*, KTH Royal Institute of Technology, Stockholm. OCLC: 943045309.

Sharma, S., Cui, Y., He, Q., Mohammadi, R. and Li, Z. (2018). Data-driven optimization of railway maintenance for track geometry, *Transportation Research Part C: Emerging Technologies* **90**: 34–58.

Shaulk, R. (2016). *Data Analytics for RCF Damages on the Dutch HSL Track*, Master of Science, Delft University of Technology.

Shetty, S. H., Shetty, S., Singh, C. and Rao, A. (2022). Supervised Machine Learning: Algorithms and Applications, *Fundamentals and Methods of Machine and Deep Learning*, John Wiley & Sons, Ltd.

Simha, N. K., Fischer, F. D., Shan, G. X., Chen, C. R. and Kolednik, O. (2008). J-integral and crack driving force in elastic–plastic materials, *Journal of the Mechanics and Physics of Solids* **56**(9): 2876–2895.

Smith, R. A. (2002). Rolling Contact Fatigue of Rail: Review of Current Understanding, *Technical report*.

Solomatine, Dimitri P. & Ostfeld, A. (2008). Data-driven modelling: some past experiences and new approaches, *Journal of Hydroinformatics* **10**(1): 3–22.

Sourget, F & Riollet, A.-M. (2006). PROBABRAIL: A statistical tool to improve preventative maintenance on rails.

Stanford, J. (2000). Burlington Northern Santa Fe Preventive Gradual Grinding Initiative, p. 45.

Stanford, J., Magel, E. and Sroba, P. (2001). Transitioning from corrective to preventive rail grinding on the BNSF railroad, *Confronting the Barriers of Heavy Haul Rail Technology*, p. 11.

Stanton, J. M. (2001). Galton, Pearson, and the Peas: A Brief History of Linear Regression for Statistics Instructors, *Journal of Statistics Education* **9**(3): null. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/10691898.2001.11910537.
URL: *https://doi.org/10.1080/10691898.2001.11910537*

Steinberg, D. (2009). *CART: Classification and Regression Trees*, Chapman and Hall/CRC, pp. 193–216.

Sugiyama, M. (2016). Chapter 38 - Outlier Detection, *in* M. Sugiyama (ed.), *Introduction to Statistical Machine Learning*, Morgan Kaufmann, Boston, pp. 457–468.

Sun, B., Du, J. and Gao, T. (2009). Study on the Improvement of K-Nearest-Neighbor Algorithm, *2009 International Conference on Artificial Intelligence and Computational Intelligence*, Vol. 4, pp. 390–393.

Sun, Shiliang & Huang, R. (2010). An adaptive k-nearest neighbor algorithm, *2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery*, Vol. 1, pp. 91–94.

Syakur, M. A., Khotimah, B. K., Rochman, E. M. S. and Satoto, B. D. (2018). Integration K-Means Clustering Method and Elbow Method For Identification of The Best Customer Profile Cluster, *IOP Conference Series: Materials Science and Engineering* **336**(1): 012017. Publisher: IOP Publishing.
URL: *https://dx.doi.org/10.1088/1757-899X/336/1/012017*

Szkuta, B., Sanabria, L. and Dillon, T. (1999). Electricity price short-term forecasting using artificial neural networks, *IEEE Transactions on Power Systems* **14**(3): 851–857.

Taneja, S., Gupta, C., Goyal, K. and Gureja, D. (2014). An Enhanced K-Nearest Neighbor Algorithm Using Information Gain and Clustering, *2014 Fourth International Conference on Advanced Computing & Communication Technologies*, pp. 325–329. ISSN: 2327-0659.

Thompson, D. (2009). Chapter 9 - Curve Squeal Noise, *in* D. Thompson (ed.), *Railway Noise and Vibration*, Elsevier, Oxford, pp. 315–342.

Tobler, W. R. (1970). A Computer Movie Simulating Urban Growth in the Detroit Region, *Economic Geography* **46**(sup1): 234–240. Publisher: Routledge _eprint: https://www.tandfonline.com/doi/pdf/10.2307/143141.
URL: *https://www.tandfonline.com/doi/abs/10.2307/143141*

Trummer, G., Marte, C., Dietmaier, P., Sommitsch, C. and Six, K. (2016). Modeling surface rolling contact fatigue crack initiation taking severe plastic shear deformation into account, *Wear* **352-353**: 136–145.

Valente, E. F. a. F. D. M. a. G. (2008). Multivariate analysis of fMRI time series: classification and regression of brain responses using machine learning | EndNote Click, *Magnetic Resonance Imaging* **26**.

Vapnik, V. (2006). *Estimation of Dependences Based on Empirical Data*, Springer Science & Business Media.

Venkateswara Rao, K. (2012). Spatiotemporal Data Mining: Issues, Tasks And Applications, *International Journal of Computer Science & Engineering Survey* **3**(1): 39–52.
URL: *http://www.airccse.org/journal/ijcses/papers/0212ijcses04.pdf*

Whitney, B. (2020). Rail Management: Hot Topics and Developments.

Wilson, N. (2018). NR/L2/TRK/001/Mod06 & Mod07 Changes: ANG-LTI-003.

Wu, J., Chen, X.-Y., Zhang, H., Xiong, L.-D., Lei, H. and Deng, S.-H. (2019). Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimizationb, *Journal of Electronic Science and Technology* **17**(1): 26–40.

Xu, P., Zhu, C., Zeng, H. and Wang, P. (2020). Rail crack detection and evaluation at high speed based on differential ECT system, *Measurement* **166**: 108152.

Yang, L. (2007). Control of spatial discretisation in coastal oil spill modelling.

Yuan, Y., Zhang, W., Zhang, P., Fan, X. and Zhu, M. (2021). Porous grinding wheels toward alleviating the pre-fatigue and increasing the material removal efficiency for rail grinding, *Tribology International* **154**.

Zakharov, S., Kharris, Landgren, Turne and Eberson (2001). *Guidelines to best practices for heavy haul railway operations: wheel and rail interface issues.*

Zarembski, A. M., Attoh-Okine, N. and Einbinder, D. (2016). On the Relationship between Track Geometry Defects and Development of Internal Rail Defects, p. 7.

Zhang, S. Y., Spiryagin, M., Ding, H. H., Wu, Q., Guo, J., Liu, Q. Y. and Wang, W. J. (2022). Rail rolling contact fatigue formation and evolution with surface defects, *International Journal of Fatigue* **158**: 106762.
URL: *https://www.sciencedirect.com/science/article/pii/S0142112322000433*

Zhang, W., Zhang, P., Zhang, J., Fan, X. and Zhu, M. (2020). Probing the effect of abrasive grit size on rail grinding behaviors, *Journal of Manufacturing Processes* **53**: 388–395.

Zhou, K., Ding, H. H., Zhang, S. Y., Guo, J., Liu, Q. Y. and Wang, W. J. (2019). Modelling and simulation of the grinding force in rail grinding that considers the swing angle of the grinding stone, *Tribology International* **137**: 274–288.

Zhu, W.-L., Yang, Y., Li, H. N., Axinte, D. and Beaucamp, A. (2019). Theoretical and experimental investigation of material removal mechanism in compliant shape adaptive grinding process, *International Journal of Machine Tools and Manufacture* **142**: 76–97.

Zoeteman, A., Dollevoet, R. and Li, Z. (2014). Dutch research results on wheel/rail interface management: 2001–2013 and beyond, *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit* **228**(6): 642–651.