

Copyright © and Moral Rights for this thesis and, where applicable, any accompanying data are retained by the author and/or other copyright owners. A copy can be downloaded for personal non-commercial research or study, without prior permission or charge. This thesis and the accompanying data cannot be reproduced or quoted extensively from without first obtaining permission in writing from the copyright holder/s. The content of the thesis and accompanying research data (where applicable) must not be changed in any way or sold commercially in any format or medium without the formal permission of the copyright holder/s.

When referring to this thesis and any accompanying data, full bibliographic details must be given, e.g.

A. I. Parkes (2021) “The Importance of Error Measures for Machine Learning Regression to Approximate the Ground Truth”, University of Southampton, Maritime Engineering, PhD Thesis, pagination.

UNIVERSITY OF SOUTHAMPTON

The Importance of Error Measures for Machine Learning Regression to Approximate the Ground Truth

by

Amy Isabel Parkes

A thesis submitted for the degree of
Doctor of Philosophy

in the
Faculty of Engineering and Physical Sciences
Maritime Engineering

Thursday 5th August, 2021

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES
MARITIME ENGINEERING

Doctor of Philosophy

**THE IMPORTANCE OF ERROR MEASURES FOR MACHINE LEARNING
REGRESSION TO APPROXIMATE THE GROUND TRUTH**

by Amy Isabel Parkes

As machine learning technology improves, it is increasingly relied upon when making significant decisions which require a high level of trust. Accuracy and interpretability is paramount for trust in regression methods, which comprise a large portion of the field. To apply these methods with confidence there needs to be a certainty that they have modelled the ground truth of a dataset—the correct input-output relationships. Conventional regression error measures, however, do not ensure that the correct relationships are modelled, as they only require accurate point predictions to assign low error to a method. A case study of power prediction for merchant vessels is used to illustrate the problem, where accurate prediction and correct input-output relationship modelling is required, although there is limited understanding of these input-output relationships. For this problem neural networks can produce predictions with a 2% Mean Absolute Relative Error, which is low enough for use in fuel saving devices on-board vessels in operation. The methods developed in this thesis have been deployed on over a dozen merchant vessels operated by Shell Shipping and Maritime, saving over 1/4 million tonnes of CO₂ emissions in 2020. However, the predictions are not interpretable, as the input-output relationships modelled are not consistent or correct. A new error measure, the Mean Fit to Median Error, is investigated which ensures networks approximate the conditional averages and is applicable to any dataset. This is verified on 36 artificial datasets, where the ground truth is known, and is shown to correlate to the ground truth on average 60% higher than traditional error measures correlate to the ground truth. The Mean Fit to Median Error is then applied to the ship powering example and shows a shift in the approximated relationships for the same Mean Absolute Relative Error values, showing an improvement in determining the ground truth. Networks reporting low Mean Fit to Median errors model more consistent and correct input-output relationships and are robust to areas of sparse data.

Contents

List of Figures	vii
List of Tables	xiii
Nomenclature	xv
Declaration of Authorship	xvii
Acknowledgements	xix
1 Introduction	1
1.1 Machine Learning Error Measures to Improve Inductive Bias	1
1.2 Aim and Objectives	6
1.3 Research Novelty	7
1.4 Scope of the Work	7
1.5 Outline of the Study	7
1.6 Publications	9
1.6.1 Journal Publications	9
1.6.2 Conferences and Technical Reports	10
1.7 Research Impact	10
2 Literature Review	13
2.1 Overview	13
2.2 Loss Functions	15
2.3 Traditional Regression Methods	17
2.4 Artificial Neural Networks	19
2.5 Kernel Methods	20
2.6 Bayesian Methods	22
2.7 Physics-Based Methods	25
2.8 Ship Power Prediction Example	27
2.8.1 Traditional Regression	28
2.8.2 Support Vector Machines	30
2.8.3 Bayesian Methods	31
2.8.4 Artificial Neural Networks: Data Treatment	32
2.8.5 Artificial Neural Networks: Performance Measures	35
2.9 Summary	37
3 Neural Networks for Ship Power Prediction	39
3.1 Artificial Neural Networks	39
3.1.1 Activation Functions and Initialisers	41
3.1.2 Optimisers	42
3.1.3 Regularisation	44
3.1.4 Performance Assessment	46

3.2	Merchant Vessel Data	47
3.2.1	Variable Selection	47
3.2.2	Filtering	49
3.2.3	Data Analysis	50
3.3	Model Selection	55
3.4	Benchmarking	64
3.4.1	Agreement With Traditional Methods	64
3.4.2	Sensitivity to Data Quantity	65
3.5	Vessels Without Data	66
3.5.1	Fleet data	67
3.5.2	Validate Neural Network Parameters	72
3.5.3	Prediction for Ship Without Data	74
3.6	Discussion	75
3.7	Summary	77
4	New ‘Fit to Median’ Measure	79
4.1	Regression Metrics	79
4.1.1	Minkowski-r Derivatives	80
4.1.2	Mean-Based Measures	83
4.2	Derivation of the New Error Measure	84
4.3	Artificial Datasets	86
4.4	Assess New Metric	89
4.4.1	Effectiveness on a Range of Datasets	89
4.4.2	Investigation into the Divergence	94
4.5	Ship Power Prediction	98
4.5.1	Similarity to Artificial Data	98
4.5.2	Potential for Reduced Compute Requirements	99
4.5.3	Improved Consistency of Input-Output Curves	100
4.6	Unseen Vessel Power Prediction	103
4.7	Summary	109
5	Discussion	111
5.1	Limitations	114
5.2	Future Work	115
6	Conclusion	117
	References	119
A	Supplementary Figures for Section 3.5	129
B	Supplementary Figures for Section 4.5	133
C	Supplementary Figures for Section 4.6	137

List of Figures

1.1	Example of two curves tracking similar patterns in the x range from 0-10 but diverging above x values of 10.	2
1.2	A schematic illustration of the conditional average of y with respect to x , represented by the function ϕ . At any given value x_0 of the input variable, the conditional average $\phi(x_0)$ is given by the average of y with respect to the distribution $p(y x_0)$ of the target variable, for that value of x (Bishop 1995).	2
1.3	Size and operating environment makes predicting merchant vessel behaviour difficult (NOAA 2017).	4
1.4	Comparing the ship data distribution to the distribution of data from a wind turbine (Parri 2017). Input variables of wind speed for (A) the wind turbine, and (B) ship speed for the ship, as they are the inputs most highly correlated to the output power.	5
1.5	Dataset usage in relation to thesis chapters.	8
2.1	Frequentist regression methods contextualised in relation to modelling flexibility and required domain knowledge. Bayesian approaches to all methods can be taken but do not provide any improved modelling of the ground truth.	14
2.2	Neural Network with four layers (two hidden layers) and three neurons in each hidden layer	19
2.3	Illustrations of an inverse problem, where $f(x) = x + 0.3\sin(2\pi x) + \epsilon$ with $\epsilon \sim N(0, 0.1)$. A) $f(x)$ against x , where f is a function also showing the ground truth without noise and B) the inverse, $f^{-1}(x)$ which is not a function, where the conditional average of $f^{-1}(x)$ does not approximate the ground truth relationship f^{-1}	27
2.4	Frequentist regression methods suitable for ship power prediction example contextualised in relation to modelling flexibility required and domain knowledge available.	28
2.5	Example of sinusoidal function not identified by a lower sampling frequency. . . .	34
3.1	Illustration of conjugate gradient approach compared to standard steepest descent, where contours represent an error surface with the global minimum in the middle circle (Tasneem 2019).	43
3.2	Scatter graph of speed through the water and shaft power of the dataset (A) before any trimming, (B) after removing all datapoints where shaft power is 0. . .	50
3.3	Box plots for every recorded value of shaft power at each recorded speed after trimming	50
3.4	Box plots for every recorded value of shaft power for each recorded wind direction	51
3.5	Wind direction histogram and average wind speed in each histogram bin	52
3.6	Box plots for every recorded value of shaft power at each recorded wind speed . .	52
3.7	Box plots for every recorded value of shaft power at each recorded wave height . .	53
3.8	Box plots for every recorded value of shaft power at each recorded draught	53
3.9	Box plots for every recorded value of shaft power at each recorded trim	54
3.10	Mean Absolute Relative Error in power prediction for a range of network sizes, each point is the average of 5 individual network runs.	55

3.11	Mean absolute relative training and testing error in power prediction for networks with 300 neurons for a range of layers.	56
3.12	Power prediction from a neural network with 1 hidden layer and 10 neurons A) one network prediction of isolated speed alongside visualisation of the mean and median shaft power conditioned on the speed, to illustrate the accuracy of the predicted relationship and B) multiple network predictions for isolated wind direction, without conditional averages, to illustrate the consistency of predicted relationships.	57
3.13	Power prediction from a neural network with 2 hidden layers and 50 neurons illustrating isolated speed alongside data visualisation	59
3.14	Power prediction from a neural network with 3 hidden layers and 300 neurons A) isolated speed-power curve and B) isolated wave height-power curve alongside data visualisation	60
3.15	Power prediction from a neural network with 4 hidden layers and 500 neurons illustrating isolated speed alongside data visualisation.	61
3.16	Predicted isolated speed power curves for 5 separate runs for differently sized networks (A) networks with 1 layer and 10 neurons, (B) networks with 2 layers and 50 neurons, (C) networks with 3 layers and 300 neurons, and (D) networks with 4 layers and 500 neurons.	62
3.17	Neural network of shape (3,300) prediction curve, compared to a cubed and a fitted power regression where the fitted power is 2.2494.	64
3.18	Power prediction mean Mean Absolute Relative Error, with error bars of maximum and minimum observed error, from networks of size (3,300) trained on randomly and literally sampled months of data. Up to 6 months of data are illustrated for readability of the figure, as randomly sampled dataset errors are in line with the full dataset, however the literally sampled errors continue to decrease steadily and reach errors in line with the full dataset at 20 months.	66
3.19	The size of each vessel's dataset, with each class of vessel signified by a different colour.	68
3.20	The distribution of the observed shaft powers for half knot bins of speed through the water for ship F. In the box and whisker plots the boxes contain 50% of the distribution and the whiskers extend to the datum which is at 1.5 times the interquartile range.	69
3.21	Comparison of the average observed power at half knot intervals of speed through the water for all vessels in the fleet (A) and with \pm one standard deviation of the distribution of shaft powers in each interval (B).	70
3.22	Comparison of the average observed power at half knot intervals of wind speed for all vessels in the fleet (A) and with \pm one standard deviation of the distribution of shaft powers in each interval (B).	71
3.23	Comparison of the average observed power at 20cm intervals of draught for all vessels in the fleet (A) and with \pm one standard deviation of the distribution of shaft powers in each interval (B).	72
3.24	The distribution of mean absolute relative error from multiple networks of size (3, 300) for individual vessels, showing consistent predictions for individual ships and a maximum difference of 1% between the mean error for different ships. In the box and whisker plots the boxes contain 50% of the distribution and the whiskers contain 90% of the distribution and the circles show outliers.	73
3.25	The distribution of mean absolute relative errors from multiple networks trained on all of the vessels apart from the ship tested on.	74
4.1	Euler Diagram of Regression Error Measures.	80
4.2	Illustrations of Jensen's inequality $\phi(E[X]) \leq E[\phi(X)]$ for convex function ϕ where A) the transformed mean of a distribution $\phi(E[X])$ is less than the mean of the transformed distribution $E[\phi(X)]$ (Jensen 1906) and B) where the gap caused by this inequality is smaller for the median of the distributions than for the mean. (Merkle 2005).	83

4.3	Illustration of (A) expectation of X and Y distribution, (B) partitioning X distribution to create proxy curve.	85
4.4	Illustrations of different datasets generated for this study: A) boxplots showing the spread of y values for each interval of x_0 along with f_0 from a dataset of type a from Table 4.2, shading of each boxplot shows the quantity of datapoints in any given interval, illustrating the sparse areas of data created at the tails of the x_1 distribution; B) scatter plot of x_0 to y for a dataset of type a showing f_0 ; C) boxplots showing the spread of y values for each interval of x_3 along with f_3 from a dataset of type h ; D) scatter plot of x_3 to y for a dataset of type h showing f_3 ; E) boxplots showing the spread of y values for each interval of x_5 along with f_5 from a dataset of type l ; F) scatter plot of x_5 to y for a dataset of type l showing f_5	88
4.5	1,500 networks assessed on the dataset of type d , where the data is fit with a linear regression a) the Mean Absolute Relative Error against Mean Fit To Ground Truth with R^2 of 0.0036 and b) the Mean Fit To Median against Mean Fit To Ground Truth with R^2 of 0.9801.	91
4.6	1,500 networks assessed on the dataset of type b , where the data is fit with a linear regression a) the Mean Absolute Relative Error against Mean Fit To Ground Truth with R^2 of 0.0016 and b) the Mean Fit To Median against Mean Fit To Ground Truth with R^2 of 0.8464.	92
4.7	1,500 networks assessed on the dataset of type l , where the data is fit with a linear regression a) the Mean Absolute Relative Error against Mean Fit To Ground Truth with R^2 of 0.2704 and b) the Mean Fit To Median against Mean Fit To Ground Truth with R^2 of 0.5184.	93
4.8	Heatmaps illustrating the average R^2 of a error measure to the Mean Fit To Ground Truth from the different types of dataset, stipulated in Table 4.2 a) the average R^2 of the traditional, Mean Absolute Relative Error, error measure against the Mean Fit To Ground Truth and b) the average R^2 of the proposed, Mean Fit To Median, error measure against the Mean Fit To Ground Truth.	94
4.9	1,500 networks assessed on the dataset of type e with a) Mean Absolute Relative Error against Mean Fit To Ground Truth and b) Mean Fit To Median against Mean Fit To Ground Truth. Where complexity, or number of connections in each network, is indicated by the colour of each point and shows no pattern for either plot.	95
4.10	1,500 networks assessed on the dataset of type l with a) Mean Absolute Relative Error against Mean Fit To Ground Truth and b) Mean Fit To Median against Mean Fit To Ground Truth. The difference between the Mean Fit To Median and ‘Fit to Mean’ of each network is indicated by their colour, with networks where the ‘Fit to Mean’ is lower coloured lighter and networks where the Mean Fit To Median is lower coloured darker.	96
4.11	Histogram illustrating the multi-modal distributions of Mean Absolute Relative Errors from 1,500 randomly sized neural networks for 4 of the 36 datasets. The bins which contain the 10 networks which best approximate the ground truth are coloured pink, none of which produce the lowest observed Mean Absolute Relative Errors. A) dataset of type b has a p-value of 8^{-8} , B) dataset of type a has a p-value of 1^{-61} , C) dataset of type l has a p-value of 3^{-24} , and D) dataset of type h has a p-value of 2^{-4}	97
4.12	(A) Mean Fit to Median Error and Mean Absolute Relative Error of the 1,000 neural networks to predict ship powering, with an R^2 of 3^{-32} , (B) the bimodal distribution of Mean Absolute Relative Error.	99
4.13	Mean Fit to Median Error against Training Time with Mean Absolute Relative Error illustrated by colour of point.	100
4.14	Visualisations of isolated relationships learnt, between ship speed and power. (A) from the 5 networks out of 1,000 with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.	101

4.15	Visualisations of isolated relationships learnt, between wave height and power. (A) from the 5 networks out of 1,000 with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error. .	102
4.16	Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error.	103
4.17	Visualisations of range of isolated relationships learnt, between wind speed and power for Ship A. (A) from the 5 networks with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error. .	105
4.18	Visualisations of range of isolated relationships learnt, between ship speed and power for Ship F. (A) from the 5 networks with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error. .	106
4.19	Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error for unseen ship prediction, averaged over all input-output curves for all ships.	107
4.20	Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error, averaged over all input-output curves for (A) Ship A, (B) Ship C2, (C) Ship E1, (D) Ship E2, and (E) Ship F.	108
A.1	Boxplots showing the distribution of Mean Absolute Relative Error of networks trained and tested on data from the same ship, for all ships, for varying network sizes.	129
A.2	Histograms of draught from full dataset and random sample of size 150,000, for ship C1.	130
A.3	Histograms of shaft power from full dataset and random sample of size 150,000, for ship C2.	130
A.4	Histograms of wind speed from full dataset and random sample of size 150,000, for ship C4.	130
A.5	Histograms of vessel speed from full dataset and random sample of size 150,000, for ship D1.	131
A.6	Histograms of trim from full dataset and random sample of size 150,000, for ship D2.	131
A.7	Histograms of wind speed from full dataset and random sample of size 150,000, for ship E1.	131
A.8	Histograms of trim from full dataset and random sample of size 150,000, for ship E2.	132
A.9	Histograms of draught from full dataset and random sample of size 150,000, for ship F.	132
B.1	Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error, averaged over all input-output curves.	133
B.2	Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error, for each input-output curve individually. (A) Wind Speed, (B) Wind Direction, (C) Draft, (D) Ship Speed, (E) Trim and (F) Wave Height.	134
B.3	Visualisations of isolated relationships learnt, between wind speed and power. (A) from the 5 networks out of 1,000 with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.	135
B.4	Visualisations of isolated relationships learnt, between wind direction and power. (A) from the 5 networks out of 1,000 with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error. .	136

C.1	Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error, averaged over all input-output curves for (A) Ship A, (B) Ship C1, (C) Ship C2, (D) Ship C4.	138
C.2	Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error, averaged over all input-output curves for (A) Ship D1, (B) Ship D2, (C) Ship E1, (D) Ship E2, and (E) Ship F.	139
C.3	Visualisations of range of isolated relationships learnt, between ship speed and power for Ship E2. (A) from the 5 networks with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.	140
C.4	Visualisations of range of isolated relationships learnt, between wind speed and power for Ship C1. (A) from the 5 networks with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.	141

List of Tables

2.1	Physics-Based Methods	25
2.2	Studies comparing traditional regression methods to neural networks for ship power prediction.	30
2.3	Studies comparing support vector machines to neural networks for ship power prediction.	31
2.4	Studies comparing Gaussian processes to neural networks for ship power prediction.	31
2.5	Applications of neural networks to predict ship propulsion, network sizes used, coefficient of variation of shaft power and ship speed.	32
2.6	Applications of neural networks to predict ship propulsion, network sizes used, dataset specifics including coefficient of variation of shaft power and ship speed, and errors reported.	36
3.1	Selected Hyperparameters	40
3.2	Some common activation functions	41
3.3	Neural network model selected for ship power prediction	63
3.4	The coefficients of variation for ship speed and powering and quantity of data used in this chapter compared to other datasets used for neural network applications	77
4.1	Point based error measures, derivatives of Minkowski-r Metrics	81
4.2	Varieties of Dataset Generated for Trialling Error Measure, illustrated in Figure 4.4.	87
4.3	Selected hyperparameters for assessing new error measure	90

Nomenclature

draught (m)	the length from the bottom of the hull to the waterline
trim (m)	a measure of what angle in the water a vessel is
artificial neural network	a machine learning method inspired by biological brains
epoch	a training cycle of a neural network, where the network ‘sees’ the entire training dataset
ground truth	the data generating process, for the ship power prediction scenario this equates to the physics of drag and propulsion
regression	a statistical modelling technique for estimating the relationships between a dependent variable and one or more independent variables
inductive bias	the inherent prioritisation of one solution over another
conditional average	the average value of a variable conditioned on a related variable
error measure	a value calculated to assess the quality of a statistical approximation
extrapolation	prediction in regions of the input domain where there is no training data, or sparse training data
y	output variable (shaft power for the ship case study)
\hat{y}	predicted/estimate output value
Y	vector of output variable values
ϕ	relation from x to y
x^i	input variables such that $y = \phi(x^1, x^2, \dots, x^m)$
X^i	vector of input variable (x^i) values
a	activation of a neuron
m, n, r	integers
V	ship velocity

Declaration of Authorship

I, Amy Isabel Parkes, declare that this thesis entitled The Importance of Error Measures for Machine Learning Regression to Approximate the Ground Truth and the work presented in it are 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. Either none of this work has been published before submission, or parts of this work have been published as: (Parkes et al. 2018), (Parkes et al. 2019).

Signed:

Date:

Acknowledgements

I would like to thank my supervisors, Prof. Dominic Hudson and Dr. Adam Sobey for their guidance, time and support. I would also like to thank Shell Shipping and Maritime for providing data and industrial guidance. Finally, I would like to thank my parents, family, and friends for their support and Matthew Jones for his constant sanity-checking.

Chapter 1

Introduction

1.1 Machine Learning Error Measures to Improve Inductive Bias

Regression problems, where relationships between inputs and continuous output(s) are identified, comprise a large portion of the cutting edge of the machine learning field. Often there is not a full understanding of the input-output relationships of a system being modelled and it is desirable to gain some insight into these from the trained method. However, to use a trained method with confidence, or to infer information about input-output relationships, there must be a certainty that it has modelled the correct relationships.

Machine learning methods, like neural networks, have large modelling flexibility: given enough data they can find structure and patterns in problems where complexity prohibits the explicit programming of a system's exact physical nature. They are, however, known to produce physically inconsistent results and cannot generalise off test set, so cannot be relied upon to model the ground truth of the system (Willard et al. 2020). Due to their flexibility, any number of arbitrary patterns could be modelled which provide a good approximation to the fundamental relationship over a limited input-output domain, despite these functions having no similarities outside of this range; this effect is illustrated for two functions in Figure 1.1. This demonstrates the common issue of poor extrapolation for techniques like neural networks. For example, a method trained on the x domain between 0 – 10, with limited or poor quality data for the x domain above 10, would have no way to discern which of the two curves was the true $x - y$ relationship. Both curves would produce similar error profiles for $x < 10$, using current error measures.

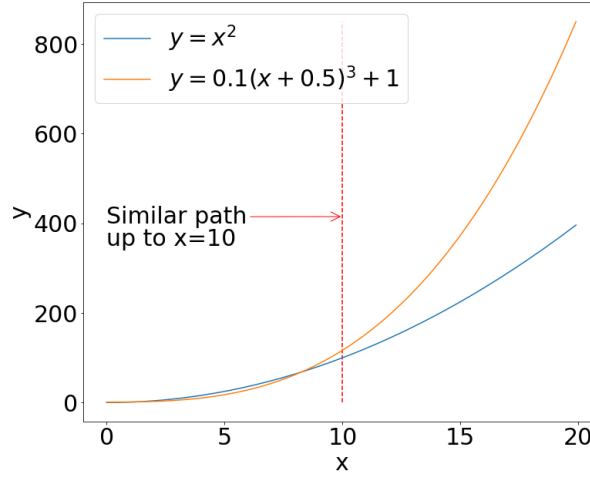


FIGURE 1.1: Example of two curves tracking similar patterns in the x range from 0-10 but diverging above x values of 10.

It cannot therefore be assumed that neural networks producing low conventional, or Minkowski-r, error measures such as Mean Squared or Mean Absolute Errors model the ground truth input-output relationships accurately. In fact, there are restrictive assumptions about a dataset that must be met to ensure that these networks approximate the conditional average, the average of the output conditioned on each isolated input variable of the dataset (Bishop 1995), illustrated in Figure 1.2. The key assumption is sufficient data across the prediction domain, if prediction outside the regions of dense data is required there is not certainty that the correct relationships have been modelled.

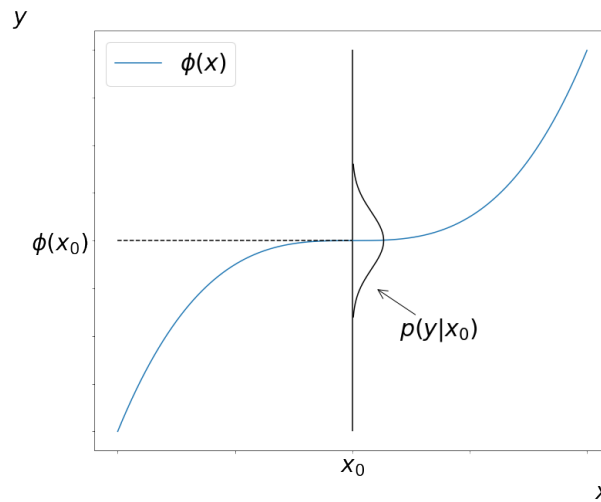


FIGURE 1.2: A schematic illustration of the conditional average of y with respect to x , represented by the function ϕ . At any given value x_0 of the input variable, the conditional average $\phi(x_0)$ is given by the average of y with respect to the distribution $p(y|x_0)$ of the target variable, for that value of x (Bishop 1995).

The conditional averages do not necessarily approximate the fundamental relationships which generated the dataset—the ground truth. This can be caused by sparse areas of data which can be affected by the ‘law of small numbers’ creating misleading averages (Tversky and Kahneman 1971). Additionally, the interaction of noise and high convexity or concavity in input-output relationships creates a gap between the conditional averages of a dataset and the underlying relationships which generated them, a direct result from Jensen’s inequality (Jensen 1906). This means that even if there is certainty that the network producing minimal loss approximates the conditional averages of the dataset, there may not be certainty that the network approximates the ground truth accurately.

To address this problem a number of methods exist to improve a network’s approximations of the ground truth. These methods include biasing the architecture of learning methods to known relationships (Park and Park 2019) (Anderson et al. 2019) (Zhang et al. 2018); using physics-guided initialization (Read et al. 2019) (Sultan et al. 2018); and adding a ‘distance from the ground truth’ measure to the loss function (Karpatne et al. 2018). They can be used to produce Reduced Order Models (Lucia et al. 2004) or improve the prediction from the physical model alone. These have been shown to produce better extrapolation predictions as their approximated patterns are robust to sparse regions of data (Willard et al. 2020). However, these methods cannot be used in a number of cases as they train models to have an inbuilt bias to the known input-output relationships and for many cases there is not a full understanding of these relationships. There is currently no way to ensure a method approximates the ground truth if the ground truth is not already known.

As many neural networks still predict poorly in extrapolated or off test set situations, despite producing low testing errors; it is clear that the input-output relationships approximated by a trained network are not representative of the ground truth in many applications (Willard et al. 2020). Low error, or high accuracy, is possible without the method modelling the correct internal functions. All regression methods require a loss function to produce predictions on a test set, and since these loss measures are all based on the same family of error measures, no regression method can be guaranteed to model the ground truth of a dataset if it is not already known. Error measures which assess whether the ground truth of a dataset has been approximated are needed to ensure accurate off test set prediction and improved trust in regions where data are gathered.

An excellent example of this scenario is ship powering, where correct modelling of the ground truth and accurate prediction is required but where there is limited understanding of that ground truth. Accurate prediction of power in weather is therefore essential to reduce fuel consumption through all aspects of the shipping industry, including design, operation and informing contract writing. Reduction of emissions in the marine industry is important as it is estimated to be responsible for 2.89% of Global Greenhouse Emissions (International Maritime Organisation

2020). Therefore, the IMO's Marine Environment Protection Committee (MEPC) require a 40% reduction in annual shipping CO_2 emissions by 2030 compared to 2008, aiming for a 70% reduction by 2050 (International Maritime Organization 2018). To meet this aim, accurate prediction of vessel fuel consumption will allow benchmarking of fuel saving devices or new design concepts, allowing current and future vessels to be more fuel efficient. In addition, weather routing optimisation can be used to ensure that the most efficient routes are taken to avoid bad weather or adverse currents. However, the prediction of emissions is difficult as the range of weather conditions encountered during operations rarely equates to the 'calm water' conditions traditional powering prediction methods are designed for (Holtrop 1984), Figure 1.3.



FIGURE 1.3: Size and operating environment makes predicting merchant vessel behaviour difficult (NOAA 2017).

Predicting ship power requirements in weather is challenging; traditional naval architecture techniques based on experimental towing tank data are expensive, as they require a number of tests and each new vessel will require these to be repeated. Due to this expense, models are rarely tested in complex sea states (Kristensen and Lützen 2012), so this data is less applicable to realistic vessel operation. To reduce this expense, empirical formulae based on these tests have been developed for calm water conditions (Holtrop and Mennen 1982) (Holtrop 1984). These methods require multiple non-trivial vessel parameters which are known during the design phase of a vessel but may not be known for a ship on charter. Such parameters are used to predict powering, but therefore methods which require these parameters are of limited practical use (Aiguero and Jia-wei 2009). Some extensions to these methods account for weather using Beaufort number, which is a coarse measure of wind strength, to infer typical wave height (Townsin et al. 1993). However, the use of inflexible methods such as polynomial regressions as well as coarse bins of 'sea state', instead of measured or hindcast wave height, reduces the accuracy of any predictions.

While simple empirical formulae are inaccurate in rough seas, it is possible to model vessel powering accurately in waves using Computational Fluid Dynamics (CFD) simulations but these may require prohibitive computational expense, as the air-sea interface is complex to model (Wackers et al. 2011). More recently continuous monitoring data, with frequencies around 30 seconds, provides an opportunity to produce fast and accurate powering estimates without the need for specific vessel parameters. However, it is difficult to analyse continuous monitoring performance data in waves using traditional regression approaches (Lakshminarayanan and Hudson 2017); due to the heavy-tailed distributions and noise levels within the data. Therefore machine learning techniques are increasingly being used on large datasets to make these predictions. They are used as they have the ability to approximate complex relationships, allowing them to model the vessel-weather relationships present in operational data (Le et al. 2020).

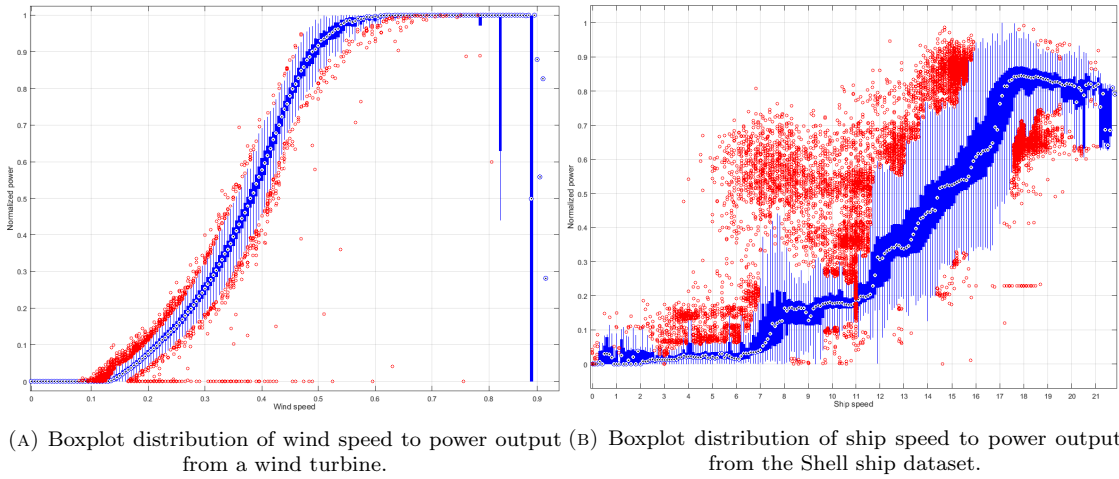


FIGURE 1.4: Comparing the ship data distribution to the distribution of data from a wind turbine (Parri 2017). Input variables of wind speed for (A) the wind turbine, and (B) ship speed for the ship, as they are the inputs most highly correlated to the output power.

Interactions between weather, specifically wave height, and vessel propulsion have been shown to be approximated by neural networks to increase prediction accuracy (Petersen et al. 2012) (Hu et al. 2019), which is not possible with traditional techniques from naval architecture. Neural networks trained for regression can be used to analyse data gathered on a specific ship, and model the unique propulsion relationships relating to hull form and unmeasurable variables such as piloting style, without the need to provide any vessel specifics (Jeon et al. 2018) (Liang et al. 2019). This power prediction is not trivial, as illustrated in Figure 1.4¹, the datasets used have multiple noisy inputs, with irregular distributions of data, where some variables are essentially discrete. In addition, datasets do not have uniform data distributions across the areas of interest in the input domain. For instance prediction in extreme situations, like hurricane force wind speeds, may have very limited or no data available but accurate prediction in these regions is

¹Both Figures 1.4a and 1.4b are taken from (Parri 2017), because direct access to the wind turbine data was not possible.

required when they are encountered. This means modelling the real relationships between input and output variables is not as straightforward as it may be for applications with less variation.

This complexity of the underlying relationships and data irregularity make the ship powering dataset a good example for determining how neural networks can model underlying relationships within a dataset. The case study of ship power prediction will therefore be used in this thesis to demonstrate the need for richer regression error measures. First, regression analysis is performed to predict vessel powering given operational data using standard machine learning techniques, next it is shown for the first time that accurate predictions for a vessel which does not gather data is possible using a data fusion from a fleet of similar vessels.

Then a new error measure is proposed to assess how accurately the ground truth is approximated by a neural network and is shown to be effective. Synthetic datasets, designed to be complex and noisy to ensure they violate the requirements for minimum Mean Square Error or Mean Absolute Error to approximate the conditional averages, are used to validate the new error measure. It is demonstrated that minimising traditional error measures cannot guarantee an accurate approximation of the ground truth. As well as this the potential causes for the poor inductive bias, the inherent prioritisation of one solution over another (Battaglia et al. 2018), caused by the error measures are discussed. Finally, the ship powering case study is used to validate the novel error measure derived, which ensures a method approximates the conditional average of a system.

1.2 Aim and Objectives

The aim of this thesis is to produce a method which improves a neural network's ability to model the relationships between inputs and outputs, and not map arbitrary patterns that produce low interpolation errors. This will increase the extrapolation capabilities and hence applicability of regression networks. This aim will be met through the following objectives:

1. Investigate the input-output relationships learnt by artificial neural networks when predicting ship powering and illustrate that they do not consistently model the ground truth.
2. Create synthetic data with similar characteristics to real data, but where all underlying functions are smooth and known, to allow evaluation of how well a network models the ground truth of the dataset.
3. Create an error measure which captures how well a regression method approximates the ground truth of a dataset and use this error measure on the synthetic data to verify this.

4. Apply the new error measure to the ship data to produce trained networks which model the ground truth of ship propulsion closer and more consistently than networks using traditional error measures.

1.3 Research Novelty

The novelty in this study is the investigation into how regression neural networks can approximate the underlying relationships between inputs and outputs for ship powering where these relationships are not already known analytically. The development of a new error measure to assess how closely a network models the relationships between inputs and outputs will allow networks which approximate the ground truth accurately to be identified. These networks should extrapolate more accurately than traditional networks, and therefore apply to different datasets more consistently. The immediate result of this is the ability to better predict ship powering performance from data, and a new method that can help us understand the relationships behind ship powering, where these were previously unknown.

1.4 Scope of the Work

This study will focus on determining whether neural networks approximate the underlying relationships within a dataset rather than mapping arbitrary functions. Neural networks are chosen as the only machine learning tool as they are the most suitable regression methods for the case study: they are suitably flexible to model the complex, interrelated relationships which govern ship powering, they also do not require any prior knowledge of the system which would be required for any physics-based approaches, as a full understanding of these relationships is not available. Gaussian processes or other Bayesian methods are not used as they have been shown to require impractical compute and do not produce any improved performance over neural networks. No laboratory experiments will be undertaken as data with known relationships is manufactured for validating the methods. Data for the power prediction case study has also been provided by Shell Trading and Shipping Company, no data collection will be undertaken as the data available from Shell is suitable in terms of quantity, quality, frequency and reliability for training the methods developed in this thesis.

1.5 Outline of the Study

The literature relating to regression methods and why they do not ensure an accurate modelling of the ground truth, will first be reviewed in Chapter 2. Ship power requirements are modelled

using artificial neural networks in Chapter 3, to demonstrate high accuracy of prediction is possible but that there is a lack of consistency in approximating relationships between inputs and outputs. An error measure to assess how accurately the ground truth of a dataset is modelled by a regression method is derived in Chapter 4; the effectiveness of this measure is verified on artificial datasets, as the ground truth is known, and then the new measure is shown to model input-output relationships which are closer to the ground truth on the ship power prediction dataset. The real world and artificial dataset usage in each section of the thesis are summarised in Figure 1.5. Chapter 5 discusses and proposes potential avenues for future research, finally the key conclusions are developed in Chapter 6.

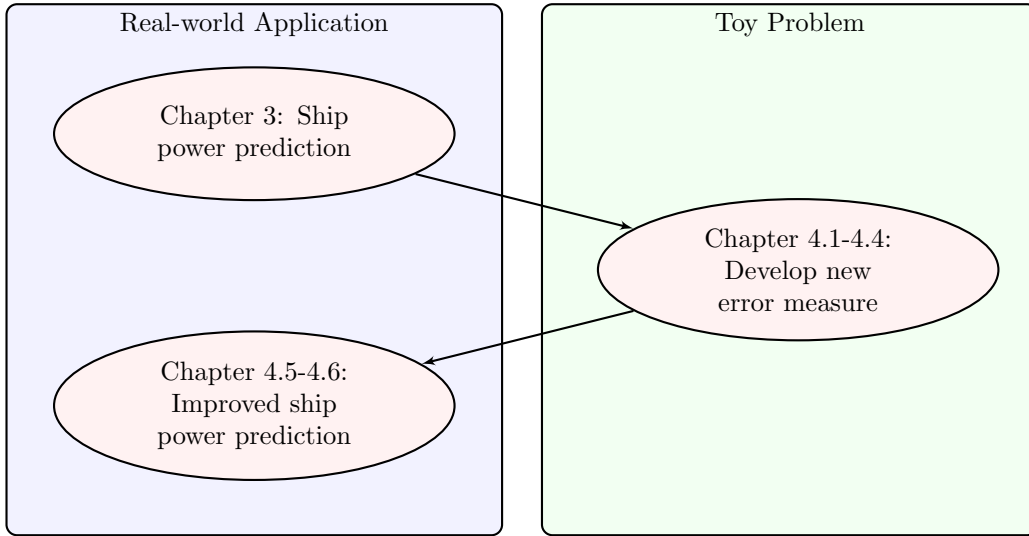


FIGURE 1.5: Dataset usage in relation to thesis chapters.

The following provides more detail outlining Chapters 3 - 4.

Chapter 3: Ship Power Prediction Using Neural Networks

Ship power prediction is performed in two different contexts in Chapter 3; single ship prediction where the isolated effect of inputs on the output is analysed, and fleet prediction where the transfer learning capabilities of single networks are assessed.

- Single Ship Prediction.** There is a limited literature investigating power prediction for a ship which gathers operational data using neural networks as shown in Chapter 2, but no application discusses the accuracy of input-output relationships modelled by the methods. The relationships modelled by networks with varying parameters are analysed and discussed in Section 3.4. The networks produce traditional error values low enough to allow significant savings by use on-board energy saving devices. However, it is noted that consistency in predicted relationships, and predictions in areas of sparse data is poor in Section 3.3.

- **Fleet Prediction.** Power prediction for a ship which does not gather operational data, from a network trained on a fleet of different vessels, is performed in Section 3.5. This has not been investigated, and the results suggest prediction is accurate enough to be used for energy saving devices. However, performance in areas of sparse or extrapolated data is poor, demonstrating that the ground truth has not been modelled accurately. The same lack of consistency in predicted relationships as shown for the single ship is noted.

Chapter 4: ‘Fit to Median’ Error Measure

No regression error measure exists which ensures that a neural network models the ground truth of a dataset. Error measures, ‘Fit to Median’ and ‘Fit to Mean’ are developed to ensure the input-output relationships are accurately approximated by a regression method in Chapter 4. They measure the distance from the predicted isolated input-output curves to the conditional average of the dataset.

- **Artificial Datasets.** To assess the abilities of the new error measures, a dataset where the ground truth relationships are known is required. Therefore a set of artificial datasets are used where the input-output relationships are explicitly defined in Section 4.3. Datasets are made to mimic the ship dataset, with simplified characteristics to allow investigation in what types of data the error measures are best suited to.
- **Improved Ship Power Prediction.** It is illustrated that the ‘Fit to Median’ error measure produces more consistent approximations of the input-output relationships for the ship powering example in Section 4.5, and that these relationships are closer to the conditional average of the dataset compared to networks using traditional error measures. For the prediction for a ship which does not gather data, an example which explicitly provides easy assessment of the extrapolation capabilities, the extrapolated predictions from networks with low Mean Fit to Median Errors are significantly more consistent and correct than networks with low Mean Absolute Relative Error values in Section 4.6.

1.6 Publications

1.6.1 Journal Publications

Parkes, A. I., Sobey, A. J. and Hudson, D. A., (2018) “*Physics-based shaft power prediction for large merchant ships using Neural Networks*”, Ocean Engineering, Vol. 166, 92-104.

Parkes, A. I., Sobey, A. J. and Hudson, D. A., Savasta, T., (2021) “*Power prediction for a vessel without recorded data using data fusion from a fleet of vessels*”, Expert Systems and Applications (Accepted for publication)

Parkes, A. I., Sobey, A. J. and Hudson, D. A. *“Towards Error Measures which Influence a Learners Inductive Bias to the Ground Truth”*, Journal of Machine Learning Research (JMLR), (Under review)

Albertelli, R. B., Parkes, A. I. and Sobey, A. J. *“Benchmarking GAs for Hyperparameter Tuning of Neural Networks”*, Neural Networks (Under preparation)

Parkes, A. I., Sobey, A. J. and Hudson, D. A. *“New Error Measure Improves Machine Learning Modelling of the Ground Truth”*, Nature (Under preparation)

1.6.2 Conferences and Technical Reports

Parkes, A. I., Savasta, T., Sobey, A. J. and Hudson, D. A., (2019) *“Efficient vessel power prediction in operational conditions using machine learning”*, Practical Design of Ships and Other Floating Structures (PRADS)

Parkes, A. I., Lakshminarayanan, P. A., Sobey, A. J. and Hudson, D. A., (2019) *“Feasibility Study for an air bubble power controller for Silverstream using Machine Learning”*, Consultancy Report for Silverstream Technologies.

Parkes, A. I., Camilleri, J. and Sobey, A. J. *“Automated, Interpretable Machine Learning for Regression”* Conference on Neural Information Processing Systems (NeurIPS), (Under review)

1.7 Research Impact

The approach developed in Chapter 3 has been applied directly to fuel saving devices on-board Liquefied Natural Gas carriers operated by Shell Shipping and Maritime, called the Just Add Water System (JAWS). The fuel saving ‘app’ shows the ship’s captain the optimal draught and trim to operate at in real time and can reduce fuel consumption of a ship by up to 7% .

The neural network methods are used to predict a ship’s power requirements for any given speed and operating condition. The current power requirement is compared to predictions for similar situations within a change in draught and trim deemed appropriate by ship operators. The energy required to change the draught and trim of the vessel is taken into account, then the draught and trim which requires least power to maintain the desired vessel speed is suggested to the ship operator for its given weather conditions.

These draught and trim optimisation devices are used on over a dozen merchant vessels operated by Shell Shipping, saving over 1/4 million tonnes of CO₂ emissions in 2020. This has been reported in marine publications in 6 countries including Mirage News Australia and Bunker

Ports News Worldwide. A link to the University of Southampton article *“250,000 tonnes of shipping CO2 emissions saved thanks to machine learning insight”* is provided².

This technology has saved Shell \$90 million in 2020 and has been patented and licensed by Shell to Kongsberg Maritime. Kongsberg provide information management platforms for over 18,000 merchant vessels and will include the fuel saving device in their interactive dashboard³.

²<https://www.southampton.ac.uk/news/2020/07/machine-learning-saves-co2.page>

³<https://vpoglobal.com/2020/09/04/kongsberg-to-deliver-shells-draft-and-trim-optimisation-software/>

Chapter 2

Literature Review

2.1 Overview

To model complex regression problems, the choice of method architecture depends on two main parameters: the amount of domain knowledge available about the problem, and the complexity of the relationships to be modelled. Some common regression methods are arranged in context of their flexibility and the required domain knowledge in Figure 2.1. This literature review will illustrate that if there is not sufficient domain knowledge to implement physics-based methods or an analytical physical model, then there is currently no way to ensure a regression method models the ground truth of the system.

If a full understanding of the ground truth input-output relationships is available then it is possible to create a physical model, which will by definition model the ground truth accurately. However, for many applications it is not possible to create a physical model as there is not sufficient knowledge of the input-output relationships. If the ground truth is partially understood then physics-based methods allow a data-driven approach to be constrained towards what is understood about the relationships, again these methods require a certain level of understanding which is not always available.

If there is minimal understanding of the system being modelled, then traditional regression, support vector machines and neural networks can be used. A small amount of understanding of the ground truth can inform parameter choices for these methods such as the kernel in a support vector machine, the type of neural network used, or the degree of a polynomial for traditional regression. The choice of regression method also relies heavily on the complexity of the system being modelled. Traditional regression approaches do not have sufficient flexibility to model multiple, interrelated, nonlinear inputs to a system. Physical models can model such complex

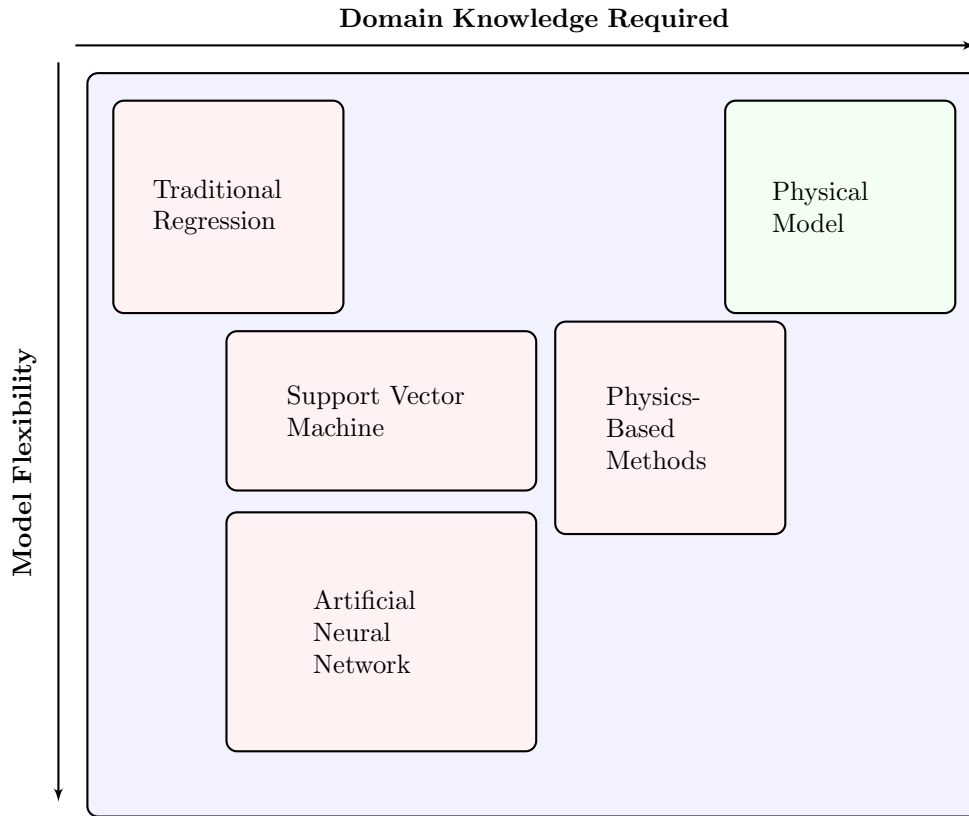


FIGURE 2.1: Frequentist regression methods contextualised in relation to modelling flexibility and required domain knowledge. Bayesian approaches to all methods can be taken but do not provide any improved modelling of the ground truth.

relationships but once designed are static and cannot adapt to changes in scenario. Physics-based methods overcome this by combining a data-driven approach with the domain knowledge, although incorporating this knowledge does impose some limits on the model flexibility.

Regardless of architecture, all regression methods require an error function to assess performance. No common regression error measure assesses how accurately the ground truth has been modelled by a method. If there is a full understanding of the input-output relationships in a dataset then error measures can be created to assess this. However for applications where this knowledge is not available; such as when traditional regression, support vector machines or neural networks are used; there do not exist any error measures to assess this.

Discussed above is the frequentist approach to regression. Bayesian approaches to all of these methods exist, where Bayes theorem is used to update probabilities, which allow natural uncertainty estimation. However the use of Bayesian estimation does not circumvent the problem that for applications without prior knowledge of the ground truth, regression methods do not accurately model the ground truth. This is because, although Bayesian methods allow inference to be performed without reference to a loss functions, a loss function is required to make predictions.

2.2 Loss Functions

The choice of loss function is similarly essential to prediction accuracy as the choice of regression method; however this choice is rarely documented as thoroughly in studies. Although countless distinct loss functions exist, they are all based on the same basic principle of the distance from the predicted value to the target value. As datasets used for regression applications increase in complexity, loss functions fail to ensure methods identify the true input-output relationships.

The most popular error measures used for regression neural networks are the Minkowski- r distance measures, where the distance measure is the Euclidean distance, to some power r , and aggregation over set of size n is the mean, equation 2.1, from Hanson and Burr (1987),

$$\text{Minkowski-}r \text{ Derivative} = \frac{1}{n} \sum_i \mathbb{N}[(y_i - \hat{y}_i)^r], r \in \mathbb{R}. \quad (2.1)$$

In these error measures higher r values penalise large deviations more, and smaller r values reduce the influence of outliers in feature space during learning. When the regression output only has one dimension, these measures equate to Mean Squared Error for $r = 2$ and Mean Absolute Error for $r = 1$. Along with Mean Absolute Percentage Error and Root Mean Squared Error these 4 error measures are the most common regression error measures used (Fildes and Goodwin 2007) (McCarthy et al. 2006).

The most common regression loss functions are Mean Absolute Error, Mean Squared Error, Mean Absolute Percentage Error and Root Mean Squared Error (Fildes and Goodwin 2007) (McCarthy et al. 2006) which are all Minkowski- r (Hanson and Burr 1987) measures or derivatives. All these functions are based around the Euclidean distance measure from the predicted value to the target value from a test set, normalised in some way, and are then aggregated. Measures based on probabilistic divergences such as Kullback–Leibler collapse down to a scaled log quotient between the predicted value to the target value when predicting single values. For applications where there is a significant domain knowledge of the input-output relationships, loss functions can be tailored to ensure an accurate model of the ground truth, discussed further in Section 2.7. However, if there is not a full understanding of the ground truth, the commonly used loss functions do not guarantee the ground truth is modelled accurately.

If y_i is a target value from the test set of size n , \hat{y}_i is the predicted value from the neural network when shown input values corresponding to y_i , the Mean Absolute Error is defined as:

$$\frac{1}{n} \sum_{i=1}^n 100 \left| \frac{y_i - \hat{y}_i}{y_i} \right|,$$

and the Mean Absolute Relative Error,

$$\frac{1}{n} \sum_{i=1}^n 100 |\bar{y}_i - \hat{\bar{y}}_i|, \text{ where } \bar{y}_i = \frac{y_i - \min(y)}{\max(y)},$$

$$\implies \frac{1}{n} \sum_{i=1}^n 100 \left| \frac{y_i - \hat{y}_i}{\max(y)} \right|.$$

Theoretically, minimising the Mean Squared Error of a neural network identically approximates the conditional mean of the dataset (Bishop 1995), and a network which minimises Mean Absolute Error identically approximates the conditional median of the data, as this is less affected by outliers. However, these proofs rely heavily on several assumptions from Bishop (1995):

- (i) the datapoints are independent;
- (ii) the distribution of the target variable is to be deterministic of the input with Gaussian noise, e.g. $y = \phi(x) + \epsilon$ where ϕ depends only on input variable x , and $\epsilon \sim N(0, \sigma^2)$;
- (iii) the dataset is homoscedastic—the standard deviation of noise, σ , is not dependent on the input x ;
- (iv) and the dataset and neural network must be sufficiently large.

These are not always realistic assumptions, and in many applications do not hold. Despite this, accurate results may still be produced by applying neural networks with Mean Squared Error or Mean Absolute Error metrics, although there is no certainty that the trained networks approximate the conditional average or the ground truth of the dataset. For example, if noise in the target variable is not Gaussian, then the results cannot distinguish between the true distribution and any other distribution with the same mean and variance. It is suggested that this is the reason regression networks can model a dataset with low errors, but model arbitrary patterns rather than the conditional average of the dataset.

The assumptions discussed above are specifically for the application of a neural network to a regression problem where there is not sufficient domain knowledge to use physics-based methods or a physical model. As neural networks have greater flexibility than other regression methods, and one of the assumptions is for sufficiently large networks; it is suggested that in scenarios where neural networks fail to model the conditional averages, no regression method with less flexibility would succeed unless the reduced flexibility steers the method towards the conditional average explicitly.

The use of Mean Squared Error and Mean Absolute Error, on datasets which do not adhere to the above assumptions leads to poor performance outside areas of dense data as the ground truth has not been approximated. In areas of dense data the ground truth is modelled, but this approximation becomes worse as the data density decreases. There is no way to assess at what

data density the quality of ground truth modelling becomes unreliable. This reduces the level of trust a user has for a model and inhibits our understanding of why certain predictions are correct and some are incorrect. This lack of reproducibility is a growing problem in the field, (Hutson 2018) and our lack of understanding means that these models are unusable in some domains such as healthcare (Voosen 2017). It also reduces the transferability of trained networks as the causal relationships are not identified by these models.

All regression methods discussed in the proceeding sections rely heavily on these loss, or error, functions. Error functions are adapted for specific applications where the ground truth is known explicitly in Section 2.7. However, no error measure exists to assess how well the ground truth is modelled, if the ground truth is not known.

2.3 Traditional Regression Methods

Regression analysis is a method for predicting the value of a dependent output variable based on given values for its input variables. There are multiple base models which can be used to approximate this relationship, the simplest example being a linear combination. When used on data with one input and one output variable, this approximates the relationship with a straight line $y = \alpha + \beta x$, where α and β may be determined by the regression analysis. The parameters are chosen to minimise a measure of the distance of all datapoints to the line, or the error ϵ .

The earliest error measure used is the squared difference $\epsilon = (y - (\alpha + \beta x))^2$, this method is called ordinary least squares and was developed in the early 1800s (Legendre 1805) (Gauss and Davis 1857). This method relies on the same assumptions discussed for neural networks in the in the previous section, with the increased restriction that the ground truth relationship between the input and the output must be a linear relationship. To compute the analysis, the model parameters which produce minimum error measure ϵ are found, expression 2.2. For basic regressions such as a linear regression, this minimum can be found analytically, and the problem is reduced to a matrix inversion $\beta = (X^T X)^{-1} X^T y$. For more complicated base models or error functions, this matrix inversion becomes infeasible, and a gradient descent method is used to approximate the minimum;

$$\arg \min_{\alpha, \beta} \sum_{i=1}^n (y_i - \alpha - \beta x_i)^2. \quad (2.2)$$

Least absolute shrinkage and selection operator (LASSO) is a least squares regression with feature selection and l_1 regularisation (Tibshirani 1996). Regularisation, where solutions involving large parameter values are penalised to improve generalisation, is introduced to some regression

methods to restrict the magnitude of parameters. l_1 and l_2 regularisation are defined as in equation 2.3 and 2.4:

$$l_1 \text{ regularised loss} = \sum_{i=1}^n (y_i - f_{\theta}(x_i))^2 + \lambda \sum_{j=1}^{|\theta|} |\theta_j|, \quad (2.3)$$

$$l_2 \text{ regularised loss} = \sum_{i=1}^n (y_i - f_{\theta}(x_i))^2 + \lambda \sum_{j=1}^{|\theta|} \theta_j^2. \quad (2.4)$$

For any base model f , with trainable parameter(s) θ , λ is pre-specified and allows the amount of regularisation to be controlled. The inclusion of this regularisation term, particularly in l_1 regularisation, automatically allows a mechanism for feature selection, as some parameters can be suppressed to near-zero. Least squares regression with l_2 regularisation is often referred to as ridge regression (Marquardt 1970).

Regression methods were significantly improved in the latter half of the 1900s, due to the increase in accessible computational power (Ramcharan 2006). This includes the creation of generalised linear models (Nelder and Wedderburn 1972), where the error model of the output variable can be approximated with a distribution different to the normal distribution. The use of more sophisticated base models such as polynomials, logistic and multinomial were also developed. These advances allow non-linear, multivariate relationships with non-Gaussian heteroscedastic error distributions to be modelled. However, for a regression to be valid the base and error model used must have the same form as the true relationship and true error model of the system. For example a polynomial regression of order three is not valid for a system where the true input-output relationship is a polynomial of order five. This is straightforward for some applications, but in scenarios where the ground truth is not fully understood it is not possible to know which base model or error model to use.

Methods, like those discussed above, which model data with the sole objective of minimising an error measure can be referred to as maximum likelihood estimation. Minimising some loss function, such as sum of squared errors used in ordinary least squares approaches, is equivalent to maximising the likelihood that the estimated parameters approximate the conditional average of the data (Bishop 1995), under the same assumptions specified above. For non-linear traditional regression methods there are similar assumptions required for the conditional averages to be approximated; the key assumptions being that the form of model is the same as the true form of the ground truth relationship and that data quantity is sufficient. For many applications there is not enough knowledge of the ground truth relationship for the first assumption and many datasets do not adhere to the second.

2.4 Artificial Neural Networks

Inspired by biological neural networks, (McCulloch and Pitts 1943), artificial neural networks are collections of nodes, also called neurons organised into layers, with directed weighted connections between neurons in sequential layers. Each neuron is connected to other neurons in the proceeding and preceding layer, each connection carries a unique weighting, and has at least one input from other neurons and exactly one output value, (Rosenblatt 1957). Explicitly, for one neuron with N inputs, let $\{x_i\}$ and $\{w_i\}$ be the inputs into the neuron and the weights on the connections, respectively, with $i \in I = \{0, \dots, N - 1\}$. This includes a bias input x_0 which is permanently set to either 1 or 0 with a corresponding variable weight value w_0 . When the neuron receives the input values from the neurons it is connected to, the activation of the neuron a is computed,

$$a = \sum_{i \in I} w_i x_i. \quad (2.5)$$

The output of the neuron is $y = f(a)$, where f is the activation function, which can be a range of pre-determined functions. Many of these neurons are stacked into layers as shown in Figure 2.2. The number of neurons in the first, known as input, layer of a network is equal to the number of independent variables of the dataset. The number of layers in between the input and output layers is variable, as is the number of neurons in each of these layers. The number of neurons in the final, known as output, layer is equal to the number of variables being modelled.

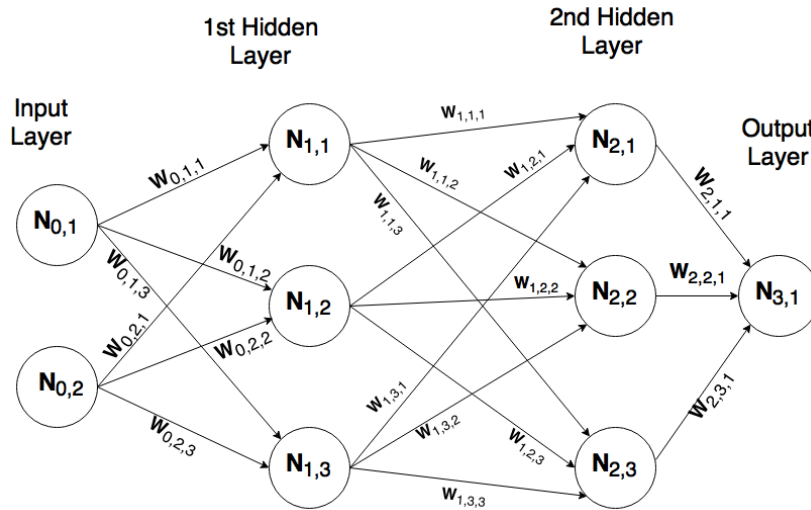


FIGURE 2.2: Neural Network with four layers (two hidden layers) and three neurons in each hidden layer

To train a network, it takes a single datapoint as input at a time, propagates this through its layers and outputs a single predicted output value. Once a prediction is produced from the output layer of a neural network the chosen loss function is used to identify how close to the

target value the prediction is. This error is then propagated back through the layers, where a gradient descent method is used to find weights which produce the minimum error for the given batch (Rumelhart et al. 1985), which is the quantity of datapoints error is aggregated over before backpropagation occurs. The process is repeated until all datapoints in the training dataset have been ‘seen’ by the network; which is called an epoch. Training involves repeating the epoch procedure until the network error is sufficiently low.

Although a sufficiently large single layer network is a universal approximator, (Cybenko 1989), infinitely sized networks are not computationally feasible. Therefore, using activation functions that are non-linear allows layers to be stacked to increase the modelling flexibility of a network. Explicitly, if all activation functions in a network with N hidden layers are linear, then the final layer is purely a linear combination of the input variables, so the N layers are equivalent to a single layer, (Minsky and Papert 1969). If the activation function at a neuron is linear, this will mean the partial derivative at the neuron is constant, i.e. with no relation to the inputs. If all activation functions in a network are linear, the gradient descent fails as all gradients are constant. The use of non-linear activation functions is a requirement for a neural network to be a universal approximator (Leshno et al. 1993).

Neural networks are utilised prolifically in the machine learning field for regression applications. Their architecture has been adapted in multiple ways, including: the addition of recurrent units to model time series data, the incorporation of kernels, and adaptations to train using Bayesian logic. Neural networks rely heavily on the error functions used to identify which relationships most accurately model the input-output relationships from the dataset. As discussed in Section 2.2, the use of a neural network with the most common error functions does not ensure the input-output relationships modelled are close to the ground truth of the dataset.

2.5 Kernel Methods

A kernel k is a measure of similarity between two points x and x' in a non-linear feature space, mapped to a Hilbert space for ease of computation,

$$k(x, x') = \langle \phi(x), \phi(x') \rangle, \quad (2.6)$$

where ϕ maps to a dot product space which is complicated to work in. So kernels allow computation in complex feature spaces to be performed in a linear space. The most common kernel is the Gaussian or radial basis function (RBF) (Mulgrew 1996), originally referred to as a potential function (Aizerman 1964), which uses the mapping

$$\phi(x) = \exp\left(-\frac{x^2}{2\sigma^2}\right), \quad (2.7)$$

to produce the kernel

$$k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right). \quad (2.8)$$

The parameter σ allows control over the smoothness of the interpolating function (Bishop 1995). Numerous other kernels are used, such as: polynomial, logistic, Taylor series expansion, and linear. By definition, sums and products of kernels are also kernels, so infinite distinct kernels exist (Burges 1998).

A kernel is described as positive semi-definite when any finite matrix constructed by pairwise evaluation $k(x_i, x_j)$ has entirely non-negative eigenvalues. The use of positive semi-definite kernels in kernel regression methods ensures that the space mapped to is a reproducing kernel Hilbert space (RKHS) (Aronszajn 1950), and so the optimal solution in the feature space is unique. Kernels are used to allow regression to be performed in complex vector spaces, without having to interact directly with them (Vapnik 1998).

Classification and regression can be performed in the reproducing kernel Hilbert space which is then projected back into the original complex vector space. A direct consequence of this is that the vectors which define the region of separation in a classification problem, or the region of acceptable error in a regression problem, are identified. These vectors are called ‘support vectors’, and the method of classification or regression in this manner is called support vector classification (SVC) or support vector regression (SVR) (Drucker et al. 1996) (Hofmann et al. 2008). As support vector classification is more common than its regression counterpart, developments in kernel methods tend to be based on classification problems (Boser et al. 1992) (Burges et al. 1996).

Other kernel methods use the negative log-likelihood function kernel which allows for a natural estimate of probability for classification and also reduces the computational cost for multi-class classification (Zhu and Hastie 2005). Although this method is called kernel logistic regression, it is only used for classification problems.

The use of multiple radial basis function kernels can be viewed as being equivalent to a neural network with one hidden layer and Gaussian activation functions (Cortes and Vapnik 1995) (Broomhead and Lowe 1988), coined radial basis function networks. However, due to the use of a kernel, the kernels must be stored as the matrix of distances between datapoints to compute a kernel method. This significantly increases the training time for kernel methods. Therefore, radial basis function networks can become infeasible to use on certain regression problems.

Support vector machines and other kernel methods all rely on the validity of the regression performed in the RKHS. As discussed in the previous section, many regressions are applied in scenarios where there is no certainty that the methods model the conditional average, or the ground truth. It therefore follows logically that the regressions performed in the RKHS may also have similar drawbacks, meaning there is also no certainty that a support vector machine or other kernel method models the ground truth of a system accurately.

2.6 Bayesian Methods

The methods discussed so far adhere to frequentist statistical principles, where for a fixed model, training dataset D , parameters θ are optimised to minimise the loss function between predictions \hat{y} and target variables y . Typically frequentist approaches produce point estimates for input values in a test set where Bayesian methods produce a probability distribution of possible predictions. The frequentist approach dominated statistical inference during the 20th century due to its tendency to be less computationally expensive than the Bayesian approach (Orloff and Bloom 2014) but capabilities of modern computing has allowed Bayesian methods to become prevalent in the 21st century.

Bayesian methods can be used for two purposes: statistical inference and prediction. Statistical inference is the process of using data analysis to infer properties of an underlying distribution. Bayesian inference is therefore statistical inference using Bayesian updating principles. Prediction is the process of using these inferred properties to predict behaviour for a specific scenario. For Bayesian methods, inference—but not prediction—can be performed without reference to a loss function. The loss functions required for performing prediction are the same as those used for any other regression problem, so still suffer the fundamental problem of not accurately modelling the ground truth when certain data assumptions are violated.

The aim of Bayesian inference is to construct posterior probability distributions for all unknown entities in a model, given the data sample. To produce these probability distributions a ‘first guess’ distribution must be defined, called the prior. The posterior probability for the parameters θ , in a model A , given the dataset D , is defined by the specific interpretation of Bayes’ theorem (Price 1763),

$$P(\theta|D, A) = \frac{P(D|\theta, A)P(\theta|A)}{P(D|A)}, \quad (2.9)$$

which underpins Bayesian inference. Where $P(D|\theta, A)$ is the likelihood of the parameters θ , $P(\theta|A)$ is the prior probability of θ given that A has occurred, and $P(D|A)$ is the evidence of the model A which acts as a normalising constant and so is often ignored in implementation.

Bayesian inference is therefore characterised by using probability distributions to model all unknown quantities, these quantities include the parameters θ discussed for the frequentist approach, as well as parameters normally fixed in the frequentist approach such as intrinsic model parameters and noise model parameters. For ease of notation θ will denote all unknown quantities to be estimated by a Bayesian approach.

In a full Bayesian approach the posterior distribution,

$$p(\theta|D) = \int p(D|\theta)p(\theta)d\theta, \quad (2.10)$$

is computed to allow model predictions which requires integration over the parameters. This process is called marginalization. To produce model predictions on a test set the posterior predictive distribution is produced by integrating the predictions of the model with respect to the posterior distribution,

$$p(y|\mathbf{x}, D) = \int p(y|\mathbf{x}, \theta)p(\theta|D)d\theta. \quad (2.11)$$

Where D denotes the training dataset, x the test input and y the output. The process of marginalisation and calculating the posterior predictive distribution often involve multiple complex integrals with no closed form solution. Therefore the full Bayesian problem is reduced to the accuracy of integration method such as nested sampling, Markov chain Monte Carlo methods or Gibbs sampling. This creates a large computational expense for implementing Bayesian methods which grows with the quantity of data (Lampinen and Vehtari 2001).

There are a multitude of methods to approximate this integration, variational inference is the field of statistical inference associated with approximating the posterior distribution, where the Kullback-Leibler divergence is used to measure how well a distribution approximates the posterior (Kullback and Leibler 1951). This often still comes with high computational expense. The most common approach to approximate the integrals in Bayesian methods in a computationally efficient manner is the maximum a posteriori approach.

The maximum a posteriori (MAP) approach to Bayesian methods is the closest to the frequentist approach as it does not model the posterior distribution of the parameters but instead aims to choose the parameters which maximise the posterior probability. The construction of a posterior distribution allows inference, but not prediction, to be performed without reference to a loss function. The likelihood describes how noisy measurements are assumed to deviate from the underlying noise-free function, whereas in this context the loss function captures the consequences of making specific parameter choices.

To use a Bayesian method to make a prediction, the posterior probability is maximised which requires an error function. Maximising the posterior probability equates to minimising the negative log-posterior loss function, this method is common as it avoids any approximation of

the integration. Explicitly, to make point predictions \hat{y} using this approach, the expected loss is minimised. For any loss function f this equates to

$$y|\mathbf{x} = \arg \min_{\hat{y}} \int f(y, \hat{y}) p(y|\mathbf{x}, D) dy. \quad (2.12)$$

The Bayesian approach allows a mechanism for performing inference when some of the prior knowledge is lacking, integrating over the posterior distribution of the unknown variables. Therefore introducing further complexity into the integration. There are also ‘non-informative priors’ (Bernardo 1979) (Berger and Bernardo 1992) which are designed to be used when no knowledge exists about the distribution of a parameter value. However, applications of the ‘no free lunch’ theorem state that if you make no assumptions concerning the target, then you have no assurances about how well you generalise (MacKay 1992). Therefore if ‘non-informative priors’ are used exclusively, there is no guarantee that the model will perform better than random on an off training dataset, so the model will have limited practical uses.

An example of Bayesian methods are Gaussian processes, where the priors on model parameters are defined with normal distributions. This reduces some of the computational expense in approximating the posterior as a single Gaussian process $a(X)$ can be defined just with the parameters mean $m(X)$ and covariance $c(X, X)$ of the process with:

$$m(X) = \mathbb{E}[a(X)], \text{ and} \quad (2.13)$$

$$c(X, X) = \mathbb{E}[(a(X) - m(X))^2]. \quad (2.14)$$

All positive definite covariance functions have an expansion in terms of kernels, discussed in Section 2.5, meaning that although treated as distinct disciplines, Gaussian processes can be defined as a kernel method (Kanagawa et al. 2018) (Smola and Schölkopf 2003).

Bayesian neural networks are Bayesian methods where artificial neural networks are used as the prior. A trained Bayesian neural network produces a distribution over weight space, as opposed to fixed weight values produced by non-Bayesian networks, therefore often requiring longer training times. They have Gaussian process limits, meaning that an infinitely large Bayesian neural network is mathematically equivalent to a Gaussian process (Lee et al. 2017). Hence the performance of a Bayesian neural network approaches that of a Gaussian process with increasing size (Matthews et al. 2018).

Normal distributions are used in Gaussian processes to simplify the minimisation of certain loss functions when approximating a point prediction. The result discussed in Section 2.2 holds in the context of Gaussian processes: the median of y conditioned on \mathbf{x} —the conditional median $p(y'|\mathbf{x}', D)$ —is often the general value which minimises the Mean Absolute Error in equation

2.12, and the conditional mean is often the value which minimises the Mean Squared Error (Rasmussen and Williams 2006). This exemplifies that once prediction from a Gaussian process is required, and an error function chosen, its abilities to model the ground truth are no better than a sufficiently large non-Bayesian neural network. This is due to the use of the same error functions which leads to accurate approximation of conditional averages where data is dense, but poor approximation where data is sparse.

2.7 Physics-Based Methods

If the ground truth is known, but a learning method is still necessary or preferable, the prior knowledge can be incorporated into the learning machine. These approaches include biasing the architecture of learning methods to known relationships; using physics-guided initialization; physics-guided kernels; and augmenting the loss function with the error of how far any prediction is from the prior knowledge model prediction. Different types of physics-based neural networks and a loose categorisation of model ‘purpose’ is illustrated in Table 2.1. Used in climate science, turbulence modelling, quantum chemistry, and biological sciences, these methods have been shown to produce better extrapolation predictions as their approximated patterns are robust to sparse regions of data (Willard et al. 2020).

TABLE 2.1: Physics-Based Methods

Purpose	Physics-Based			Hybrid
	Loss Function	Architecture	Initialisation	
Reduced Error	(Álvarez et al. 2009) (Pukrit-tayakamee et al. 2009) (Liu and Wang 2019)	(Park and Park 2019) (Anderson et al. 2019)(Zhang et al. 2018)	(Read et al. 2019)(Sultan et al. 2018) (Weymouth and Yue 2014)	(Karpatne et al. 2018) (Yao et al. 2018) (Sadowski et al. 2016)
Construct Reduced Order Model	(Lee and Carlberg 2020)	(Mardt et al. 2018)		
Inverse Modelling	(Raissi et al. 2019)	(Fan and Ying 2020)		
Solve Partial Differential Equations		(Chen et al. 2018) (Ruthotto and Haber 2019)		

Physics-based methods can be used to improve the prediction from a physical model alone (Swischuk et al. 2019) but are mostly employed to reduce the computational time required to model a system (Merwe et al. 2007). Surrogate modelling is the process by which a system is approximated by a data driven approach because it is too computationally expensive to emulate theoretically (Kim and Boukouvala 2019). Surrogate models are most commonly used for scenarios where the system is well understood, so could be modelled theoretically with infinite compute time (Forrester et al. 2008). For example weather and structural modelling. They are often utilised to produce reduced order models (Lucia et al. 2004), which allow for an acceptable level of reduced accuracy to enable a feasible method to compute.

A method producing a reduced order model may use neural networks to model the residuals from the method (Karpatne et al. 2018), in an attempt to increase accuracy without sacrificing computational requirements. This is an example of a hybrid method, shown in Table 2.1, which includes any method using multiple physics-based principles. Other hybrid methods include using a neural network to replace a component of a physical model of a system (Zhang et al. 2018), or to calibrate a physical model.

Inverse modelling refers to the practice of modelling relations between inputs and outputs which are not functions. This poses a problem to many learning algorithms as they are often designed with the assumption they will always model functions. This is particularly highlighted by the use of Minkowski-r metrics in neural networks, which approximate the conditional average of a dataset at minimal error. Where the conditional average of a function is close to the ground truth of the system, the conditional average of a relation can hold no similarities to the ground truth, Figure 2.3. Physics-based methods allow for improved modelling of inverse problems as if the relations between inputs and outputs are known they can be ‘hard-wired’ into loss functions (Raissi et al. 2019) or architectures (Fan and Ying 2020).

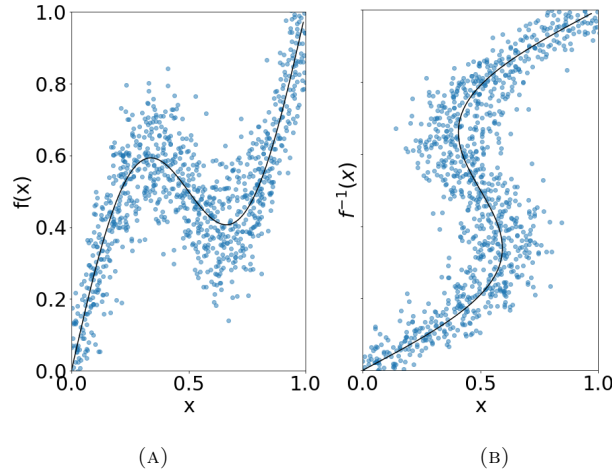


FIGURE 2.3: Illustrations of an inverse problem, where $f(x) = x + 0.3\sin(2\pi x) + \epsilon$ with $\epsilon \sim N(0, 0.1)$. A) $f(x)$ against x , where f is a function also showing the ground truth without noise and B) the inverse, $f^{-1}(x)$ which is not a function, where the conditional average of $f^{-1}(x)$ does not approximate the ground truth relationship f^{-1} .

All physics-based methods discussed here require the existence of prior knowledge of a system to improve modelling. They improve modelling of the ground truth but require sufficient domain understanding to implement. This means if there is not understanding of a scenario they cannot be employed to improve prediction accuracy or reduce training time.

2.8 Ship Power Prediction Example

Predicting power requirements for any given 300m long vessel in rough seas is not straightforward. It is not feasible to derive analytical solutions to this problem because vessel-specific parameters such as hull form are seldom known due to the tendency for vessels to be chartered not owned, and also because of complications associated with modelling the air-sea interface. Most domain knowledge for ship power prediction is in the form of empirical speed-power curves produced by regression analysis on multiple different tow tank models (Molland et al. 2011). The formulae hence require multiple vessel parameters which are not always known in a charter situation. This is therefore not sufficient domain knowledge to produce a physical model or a physics-based model.

This problem is complex, as there are multiple interrelated input variables, and latent variables such as piloting style. This means that methods such as traditional regression and support vector regression do not have adequate flexibility to accurately model power prediction. The quantity of domain knowledge available and the required method flexibility is contextualised in Figure 2.4. This shows that the most appropriate frequentist regression method for predicting ship powering is an artificial neural network.

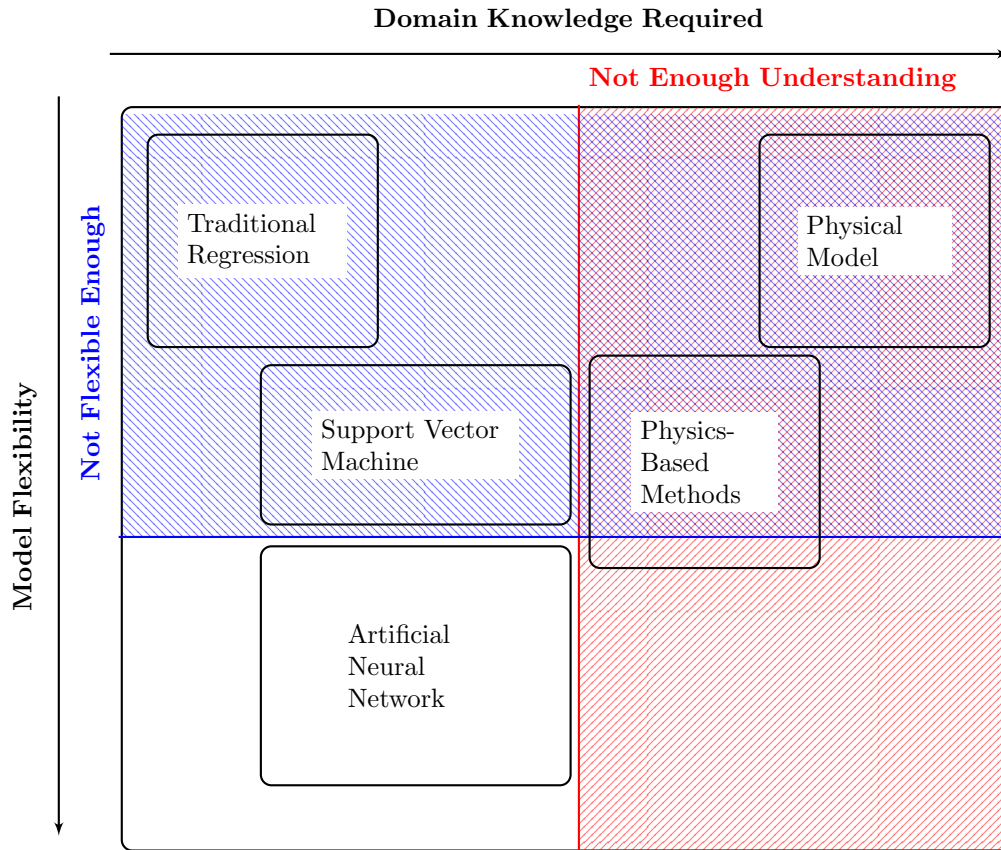


FIGURE 2.4: Frequentist regression methods suitable for ship power prediction example contextualised in relation to modelling flexibility required and domain knowledge available.

Many studies predicting ship powering compare multiple regression methods. The following sections briefly illustrate that in every study which compares method accuracy, neural networks have produced more accurate predictions than traditional regression approaches and support vector machines. It is posited that this is because traditional regression and support vector regression are not flexible enough to model the powering accurately. Studies comparing Bayesian methods to non-Bayesian neural networks are also discussed, where Bayesian methods produce predictions of a similar accuracy but require significantly longer training times.

2.8.1 Traditional Regression

Although regression methods are used in other aspects of vessel powering analysis (Abramowski et al. 2018), when large quantities of operational data are available they are outperformed by more complex machine learning modelling tools. This may be due to the inflexibility of the methods, and the requirement to pre-specify the base model as it is noted that regressions with larger modelling capabilities perform better than those with smaller modelling capabilities.

As detailed weather data is expensive to obtain, data is often partitioned into sets of wind and wave intensity, wind direction, and ‘days in service’. In one study multiple speed-power

regressions are performed for each partition (Bialystocki and Konovessis 2016), in which the relationship between fuel consumption and ship speed is assumed to be a 2nd order polynomial from theoretical calculations, however no other base model is trialled to assess the validity of this assumption. The R^2 of the base model fitted to the whole dataset is 0.7557, when split into partitions based on operational condition this increased to 0.8829. This is higher than the R^2 from a neural network application to ship power prediction, (Liang et al. 2019), however due to the partitioning of datasets used each regression is only applicable to one specific operating condition, so is not as powerful as the neural network which covers all operating conditions.

When fine grain weather data is available and variables such as wind speed, wind direction and wave height, with a data frequency higher than 1 datapoint per hour, they can be used as independent input variable(s) to improve regression accuracy. Wave swell height is used as an input variable in a LASSO regression achieving fuel consumption Mean Absolute Error of 4.9 metric tonnes per day (mt/d) (Wang et al. 2018), although this number cannot be contextualised as no information about the fuel consumption distribution is provided. Least squares regressions with l_1 and l_2 regularisations, or LASSO and ridge regression, are compared (Coraddu et al. 2017). It is noted that using l_2 regularisation produces a lower shaft power prediction error than l_1 , with a reduction of 0.4% Mean Absolute Percentage Error, from 2.62% to 3.02%. This suggests that feature selection in this scenario does not improve prediction; all input variables are required to predict the output.

In a study comparing the effectiveness of different methods, two traditional regression methods one with two input variables and one with three, using base models as cubic polynomials for the relationship between speed and fuel consumption, are shown to both produce higher errors than an artificial neural network (Le et al. 2020). Over 5 different datasets from varying size vessel, a neural network with 4 input variables outperforms both regressions consistently, the error values are shown in Table 2.2. The error for the regression using 3 input variables is 30% less than the error for 2 input variables, but this is still over 3 times higher than the error from the neural network used, illustrating the complexity of the problem.

Four different regression methods: linear, interaction, quadratic, and full quadratic, produce powering predictions with significantly less correlation to target variables than a neural network with 7 input variables (Jeon et al. 2018). A full quadratic method refers to a regression where each coefficient for terms with x^n for $n \leq 4$ are fitted, whereas the quadratic only fits the coefficient where $n = 4$, leaving the rest as 0. The best regression method was the full quadratic achieving 0.36 R value, however the best neural network achieved 0.94. An R value of less than 0.5 means the predicted values and the target values are closer to being not related than to a one-to-one relationship. This means the traditional regression methods are not accurate enough to be used in any practical application for predicting ship powering.

TABLE 2.2: Studies comparing traditional regression methods to neural networks for ship power prediction.

Study	Error Type	Regression Type	Error Value	
			Traditional Regression	Neural Network
(Le et al. 2020)	Mean Squared Percentage Error	2 input variables	8.46	1.68
		3 input variables	5.58	
(Jeon et al. 2018)	R	Linear	0.04	0.94
		Quadratic	0.15	
		Interaction	0.27	
		Fully Quadratic	0.36	

2.8.2 Support Vector Machines

Support vector regression has been used to predict powering of merchant vessels (Kim et al. 2020), with more success than traditional regression approaches, but less success than artificial neural networks. Where a neural network produces predictions with 0.94 correlation to the targets, the best support vector regression method achieved only 0.48 correlation using a quadratic kernel (Jeon et al. 2018), Table 2.3. Other support vector regression methods used a Gaussian kernel and a linear kernel producing 0.081 and 0.01 correlation to targets, which illustrates the non-linearity of the problem. The traditional regression approaches used in this study had similar levels of accuracy as the kernel methods, ranging between 0.04-0.36. Gaussian kernels are criticised for predicting negative vessel powering values which is a clear indication the method does not fit to the ground truth relationships well, as well as being outperformed by neural network methods (Liang et al. 2019).

These methods reduce the importance of specific regression parameter values, but instead increase the computational complexity of the modelling and require the type of kernel to be chosen, due to the computational complexity associated with large datasets and kernel methods. A quadratic support vector regression which is significantly outperformed by a neural network for ship power prediction, also took 800 times longer to train, taking 380.99 seconds compared to 0.46 (Jeon et al. 2018).

TABLE 2.3: Studies comparing support vector machines to neural networks for ship power prediction.

	Error Type	Kernel	Error Value	
			Support Vector Regression	Neural Network
(Liang et al. 2019)	R^2	Gaussian	0.71	0.85
(Jeon et al. 2018)	R	Linear	0.01	0.94
		Gaussian	0.081	
		Quadratic	0.48	

2.8.3 Bayesian Methods

Bayesian models have been used to predict marine vessel powering, using priors based on simplified theoretical ship powering models and a zero mean Gaussian noise prior. It is shown to be more accurate than traditional empirical ship powering models, although the computational expense of the Bayesian method is so great that the authors suggest it is not an appropriate method for practical implementations (Solonen et al. 2020).

Gaussian processes have been used to predict ship powering (Yoo and Kim 2019), when compared to artificial neural networks they were less accurate, producing higher Root Mean Squared Errors of 0.63 ± 0.055 compared to 0.32 ± 0.012 from the neural networks (Petersen et al. 2012), Table 2.4. Training time is noted as a limitation to predicting ship powering with Gaussian processes (Yuan and Nian 2018), compared to neural networks they produce higher Mean Absolute Errors and take 2,500 times longer to train (Wang et al. 2018).

TABLE 2.4: Studies comparing Gaussian processes to neural networks for ship power prediction.

Study	Error Type	Error Value	
		Gaussian Process	Neural Network
(Wang et al. 2018)	Root Mean Squared Deviation (mt/d)	27.5	19.5
(Petersen et al. 2012)	Root Mean Squared Error	0.63 ± 0.055	0.32 ± 0.012

TABLE 2.5: Applications of neural networks to predict ship propulsion, network sizes used, coefficient of variation of shaft power and ship speed.

	(Hidden layers, neurons per layer)	Dataset		Variation (%)	
		Frequency	Size	Speed	Power
(Le et al. 2020)	(2,200)		31,397	19.9	
Section 3.5	(3,300)	30s	14,000,000	16-36	41-58
(Liang et al. 2019)	(1,20)	1h	75,000		
Section 3.4	(3,300)	5min	142,196	30	59
(Jeon et al. 2018)	(2,5)		4,000		
(Bal Beşikçi et al. 2016)	(1,12)	24h	233		26
(Grabowska and Szczuko 2015)	(1,24)		89		
(Petersen et al. 2012)	(1,?)	3min	9,001	17	22
(Pedersen and Larsen 2009)	(1,5-10)	10min	679	0.6	0.1
(Leifsson et al. 2008)	(1,5)	15s			

Bayesian neural networks have been used to predict ship powering. When compared to a range of machine learning methods, including radial basis functions and support vector regressions, Bayesian neural networks and non-Bayesian neural networks produced the best predictions with no significant difference in performance (Panapakidis et al. 2020).

2.8.4 Artificial Neural Networks: Data Treatment

Neural networks are easier to implement than traditional ship power prediction methods, do not require any vessel-specific parameters, and can be implemented earlier in a vessels operating life than other performance analysis techniques. If there is trust that methods model the ground truth of a dataset, then trained networks can provide valuable insight into these relationships to optimise design and operation of the vessels.

However, none of the studies surveyed—Table 2.5—apart from Section 3.4 and Section 3.5 discuss the relationships modelled by the trained networks; reporting only the point-based error values or the correlation between predictions and target values. Therefore it is impossible to assess whether the relationships modelled by the neural networks mimics reality. It is highlighted in Section 3.4 that the input-output relationships modelled by the networks do not approximate the ground truth, as the relationships are inconsistent and on occasion violate the rules of physics. This

lack of understanding of input-output relationships is exacerbated by the trimming, averaging and partitioning of data often employed in the pre-processing stage.

Data is often averaged over a timestep (Pedersen and Larsen 2009) (Petersen et al. 2012) (Bal Beşikçi et al. 2016), rather than using the measured value at the timestep (Leifsson et al. 2008). Using the point value ensures all datapoints reflect feasible situations, as there is an error inherent in averaging multiple variables over any time period and this may cause the relationships between variables to be harder to identify, Figure 2.5. The larger the timestep averaged over, the greater loss of information, therefore applications using datapoints averaged over the last 24 hours (Bal Beşikçi et al. 2016) suffer much greater information loss than applications averaging over 3 minutes (Petersen et al. 2012). This loss of information will not be apparent in the prediction accuracy as the test set will be comprised of similarly averaged datapoints, but it will affect accuracy of point value predictions in operation.

Studies often employ the same extensive trimming procedures used in naval architecture methods to their datasets before modelling with the networks. Most notably this involves splitting the datasets into time traces based on operating mode (Leifsson et al. 2008), operating condition (Pedersen and Larsen 2009), and arbitrary consecutive non-overlapping windows (Petersen et al. 2012) (Pedersen and Larsen 2009). If the traces are short, it is common to remove any window with a missing or ‘erroneous’ value. It is suggested this pre-processing is common because the ISO standard for ship data trimming involves a similar process where any time trace with excessive variance of any variable is removed (ISO 19030 2016). This process causes more data loss than is necessary, as no application models the system as time-dependent there is no need for full 3 or 10 minute windows for network training. This may reduce how well the methods model the ground truth, as well as the prediction accuracy, as the applications which use this trimming procedure often have small initial datasets.

More recent studies do not attempt to approximate the traditional naval architecture trimming methods. To identify the boundaries of non-erroneous data, studies use one of two approaches: manually defining boundaries or fitting boundaries with data modelling techniques. Studies such as (Liang et al. 2019) manually define these boundaries, for example identifying any engine power higher than the maximum design power as outliers. Manual definition requires specific domain knowledge, often the maximum design power of an engine is not known, and will vary for each vessel. This method may not be feasible to scale for prediction for large fleets of vessels due to the requirement for multiple, subjective, manual processes.

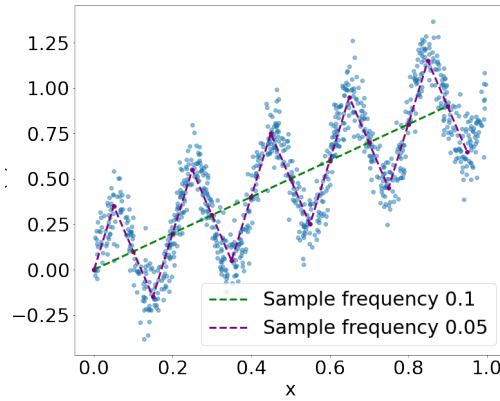


FIGURE 2.5: Example of sinusoidal function not identified by a lower sampling frequency.

To avoid manually defining the decision boundary, some studies fit the data with techniques such as regression splines (Jeon et al. 2018) or multiple Gaussian confidence intervals (Le et al. 2020) to identify outliers. This automation of the process allows scaling more easily. However, there is a possibility that the method to identify outliers underfits the data and therefore removes valid datapoints. This would not only unnecessarily reduce the size of the dataset but also could also produce a trimmed dataset which only captures the dominant relationships in a dataset, removing the effects of second order or less dominant variables as they are not modelled sufficiently in the data fitting method used.

The heavy filtering is apparent in the coefficients of variation reported for input variables. The coefficient of variation is a measure of spread of the data relative to the mean. A coefficient of 100% corresponds to a situation where the standard deviation is the same value as the mean, so the distribution has a very high spread, a coefficient of 0% corresponds to a dataset which is all one value. The coefficients range from 0.1-26% for shaft power, and from 0.6-19.9% for ship speed, Table 2.5. Often studies which split the dataset based on operating condition use a different network on each of the smaller datasets. This significantly reduces the quantity of data available for training, and produces datasets which are potentially more appropriate for a more traditional analysis tool such as multivariate regression. This is exemplified in (Pedersen and Larsen 2009) where a dataset of size 679 is split into 4 different sets, based on date, creating some sets as small as 109 points. This is suggested as the cause for the low variation in each dataset, of 0.1% for power and 0.6%.

The use of small datasets with low variance are suggested to be the cause of the small network sizes used. Few studies use more than one hidden layer, Table 2.5. A network of this size (2,5) has half the number of trainable weights as a network of size (1,20) used in (Liang et al. 2019). The datasets may have been artificially simplified by trimming processes since the small

networks employed produce low errors in all studies, but the complex nature of the problem would normally require much larger networks to accurately model the ground truth.

No averaging or filtering, beyond removing unfilled datapoints and zero power values, is employed in Section 3.4 and Section 3.5. This increases the size of datasets used, with these studies reporting the largest datasets. It also supports the hypothesis that heavy filtering reduces the variation in the datasets of other studies as the variation is nearly double that of any other study. As well as the largest datasets with the largest variation, these studies also use the largest networks and it is posited that this size of network is required to model the complex relationships and noisy data available after minimal filtering and averaging practices are employed.

2.8.5 Artificial Neural Networks: Performance Measures

It is hard to compare propulsion prediction errors from neural networks across the literature, as no consistent error measures are used throughout, Table 2.6. Errors are either reported as the correlation between predicted and actual values, R and R^2 values; a normalised mean squared, Mean Squared Percentage Error and Root Mean Squared Error; or a relative mean absolute error, Mean Absolute Relative Error and Mean Absolute Percentage Error. No measure of how well the relationships modelled approximate the ground truth is used in any of the literature, although a requirement for such a error measure is discussed in Section 3.4.

Applications reporting Mean Absolute Relative Error or Mean Absolute Percentage Error have errors from 1.65% to 2.7% with an exception of 0.1% (Grabowska and Szczuko 2015) where only 89 datapoints are used so it is suggested the study trims the dataset excessively, leading to it only representing very simple relationships which are easily modelled. The numbers between 1.65-2.7% are in line with the range of expected sensor error magnitude (Aldous et al. 2015), implying all variance in the data apart from sensor noise is modelled, although the noise magnitude will vary greatly based on the type of sensors used. For applications where Mean Absolute Relative Errors of around 2% are reported, it can be interpreted that, on average, a single test prediction will have an error of 2% of the range of data, assuming the test point is within the ranges of the training data.

Interpreting the performance from R and R^2 values is more difficult than interpreting point-based error measures (Taylor 1990). Values of 0.85 (Liang et al. 2019) and 0.75 (Bal Beşikçi et al. 2016) R^2 and 0.86 (Jeon et al. 2018) R show there is a high correlation between the predicted values and the target values. However, this does not allow for intuition around the accuracy of test set predictions.

The standard deviation of prediction accuracy for the final network is discussed in (Petersen et al. 2012), and the standard deviation of Root Mean Squared Error is reported, providing

TABLE 2.6: Applications of neural networks to predict ship propulsion, network sizes used, dataset specifics including coefficient of variation of shaft power and ship speed, and errors reported.

Error Measure	Study	(Hidden layers, neurons per layer)	Error Value
R^2	(Bal Beşikçi et al. 2016)	(1,12)	0.75
	(Liang et al. 2019)	(1,20)	0.85
R	(Jeon et al. 2018)	(2,5)	0.93
Mean Absolute Relative Error	(Grabowska and Szczuko 2015)	(1,24)	0.1
	(Parkes et al. 2019)	(3,300)	$(1.75 \pm 0.2)\%$
	Section 3.5	(3,300)	1.78-2.13%
	Section 3.4	(3,300)	$(1.96 \pm 0.1)\%$
	(Petersen et al. 2012)	(1,?)	1.65
Mean Absolute Percentage Error	(Pedersen and Larsen 2009)	(1,5-10)	2.7
Mean Squared Percentage Error	(Le et al. 2020)	(2,200)	0.97-1.83
Root Mean Squared Error	(Leifsson et al. 2008)	(1,5)	3

some intuition about the usefulness of predictions in operational applications. The distribution of prediction accuracy is discussed in more detail in Section 3.5 where power prediction for a vessel which does not gather data is shown to be feasible. As well as discussing the distribution of prediction errors, input variable sensitivity analysis is performed in (Parkes et al. 2019). Where it is shown that, if available, wave height data reduces prediction error by 0.5%.

No study other than the above reports the repeatability of results, with the majority preferring to concentrate on choosing optimal network parameters from small parametric studies. The only study to report the effect of different optimisers is (Grabowska and Szczuko 2015), others analyse the effect of the number of layers and neurons in the network on the prediction accuracy. No study reports a parametric study larger than 14 different networks. Although some of the networks surveyed in the parametric studies have as many as 4 hidden layers, all networks surveyed with more than one layer have less than 100 neurons in each layer. It is suggested that the reason small network sizes are used across all studies, Table 2.6, is the small datasets used, as well as the parametric studies only exploring small networks. These studies could be interpreted as saliency analysis to the network parameters, although the consequences of these results are not discussed beyond choosing the parameters producing the lowest error.

The only study to visualise the distributions of errors is (Pedersen and Larsen 2009). Where values of training and testing residuals are visualised and their distributions approximated, however the implications these distributions have on expected network accuracy is not discussed. Many studies show plots of their predicted against actual values, (Le et al. 2020) (Jeon et al. 2018) (Bal Beşikçi et al. 2016) (Grabowska and Szczuko 2015); since all studies produce predictions with low errors no new information about the predictive capabilities is gained. Other studies illustrate their networks prediction accuracy by visualising the prediction value over time for a chronological test set (Petersen et al. 2012) (Leifsson et al. 2008) (Liang et al. 2019). Although this method has stronger associations with usage in operation, it does not provide any new information about the predictive abilities of the network.

The only studies to visualise any input-output relationships learnt by a network are Section 3.4, (Parkes et al. 2019), Section 3.5 and (Grabowska and Szczuko 2015). This visualisation is generated by running the input variable to be isolated from its minimum to maximum value while holding all other inputs at their mean. To compare this prediction to the expectation of the dataset the isolated relationship in the data is visualised by plotting the average power conditioned on a single input variable for each input variable Section 3.5. All studies which analyse the relationships modelled by neural networks suggest that the trained network can be trusted to predict resistance within areas of dense data, but not necessarily in sparse areas.

Although all studies report low relative error or high correlation of predictions to target values, only Section 3.4 discusses the stability of the input-output relationships modelled by the trained network; concluding that within areas of dense data relationships model the ground truth, but in sparse areas of data the ground truth is not modelled. A network producing 2% error on a test set is accurate enough for use on a fuel saving device, such as a system to predict optimal operating conditions to minimise fuel consumption at a specific ship speed¹. However, in this situation there is only certainty that predictions within the dense areas of the training domain have an average of 2% error and it is difficult to confirm that prediction will not be required out of this domain for a vessel in operation. Without confirmation that the input-output relationships approximated by the trained networks are close to the ground truth throughout the prediction domain, they cannot be trusted to perform off test set in real dynamic environments.

2.9 Summary

All regression methods are only as good as their loss function, or error measures. All commonly used regression error measures are based on the distance between the target and the predicted value. The ground truth input-output relationships in a dataset are not necessarily accurately

¹<https://www.southampton.ac.uk/news/2020/07/machine-learning-saves-co2.page>

approximated by a regression method as many datasets violate the assumptions required for methods producing minimal error to approximate the conditional averages, which are often good approximations of the ground truth.

This means that no regression method reporting low traditional error measures guarantees that the ground truth of a dataset is modelled. This problem is exemplified by the ship power prediction literature; where neural networks are used to produce accurate predictions, but the relationships modelled by the networks between the inputs and outputs are illogical and inconsistent in areas of sparse or extrapolated data.

In the majority of the literature, no investigation into the input-output relationships modelled by the networks are made, and no attempt is made to ascertain if these relationships obey the rules of physics, let alone approximate the ground truth. All ship power prediction literature predicts for a single ship which gathers data, no transfer learning of learnt relationships from one ship to predict for a different ship is attempted.

Chapter 3

Neural Networks for Ship Power Prediction

This Chapter discusses the use of neural networks for regression, with the focus on analysing predictions to determine how well they model the input-output relationships within the dataset. Neural networks have been applied to the problem of ship power prediction in weather in the literature, producing low Minkowski-r error measures such as low Mean Absolute Errors. However, it is noted that extrapolation capabilities of networks are poor, and no study attempts to analyse the input-output relationships learnt by a trained network. The data has been gathered by Shell Trading and Shipping Company and specifics of the dataset are discussed further in Section 3.2.

3.1 Artificial Neural Networks

As discussed in Section 2.4, there are many parameters which need to be decided during the implementation of a neural network to a new problem. The hyperparameters which produce the largest effect on the output are the size, and type of network. The size of network required is tested and analysed in depth in Section 3.3. This section discusses the choices of the other parameters, detailed in Table 3.1, used in the neural networks in this thesis. All networks used are built using the Keras libraries (Chollet et al. 2015) in Python, this provides efficient, parallelisable networks with sufficient flexibility.

TABLE 3.1: Selected Hyperparameters

Hyperparameter	Value or set
Epochs	1,000
Early Stopping Patience	5
Early Stopping Tolerance	0
Loss function	Mean Absolute Relative Error
Performance Measures	Mean Absolute Relative Error and Relationship Visualisation
Optimiser	AdaMax (Kingma and Ba 2014)
Learning rate, β_1 , β_2 , ϵ	0.001, 0.9, 0.999, 1^{-7}
Activation Function	ReLU
Regulariser	None
Dropout	None
Initialiser	Random Normal ($\mu = 0, \sigma = 0.1$)

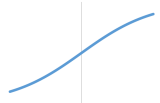
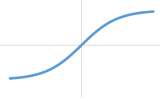



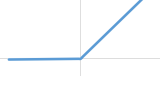
Feed forward networks are used exclusively. From domain knowledge there should exist temporal effects within the dataset, such as breaking waves on the ship or manoeuvring (Simsir and Ertugrul 2009). But from time series analysis and trialling the use of LSTM networks (Alston 2019), it is discovered that the data frequency of between 30 seconds and 5 minutes are too coarse to identify any temporal correlations as wave frequencies. This is because waves tend to have frequencies of 6-12 seconds in the open ocean, and typically a frequency twice what is being measured is required. The datasets are therefore treated as time invariant and no networks involve recursive elements and all network outputs are predicted independently.

Parameter selection is complex, as assuming an extremely sparse selection of parameter configurations, which only tests ten options for: number of neurons per layer, number of layers, optimiser, initialiser, regulariser, number of epochs and batch size, then the size of the search space is of order $\mathcal{O}(10^{11})$. Despite this complexity, hyperparameters are often tuned by trial and error (Maier and Dandy 1998), (Wang et al. 2017), (Bal Beşikçi et al. 2016), (Rajakarunakaran et al. 2008). Expert judgement can be used for rough estimates of what the hyperparameters should be and then a local, manual search for different parameters which improves the accuracy is usually conducted. A common limitation to neural networks is that there is no guidance for determining these hyperparameters and optimum values will vary between different data sets. The rule of thumb that there should be no more connections, or trainable parameters, than there are datapoints is common but this does not specify a format for the connections. More sophisticated methods of hyperparameter tuning include using other optimisation techniques like genetic algorithms and particle swarm optimisation. A study benchmarking the use of genetic

algorithms to automate the selection of neural network parameters confirmed that the parameter values used in this thesis; selected based on a combination of domain knowledge, experience with the dataset, and trail and error; are close to the optimal parameters for a neural network predicting ship powering (Albertelli 2020).

3.1.1 Activation Functions and Initialisers

TABLE 3.2: Some common activation functions

Activation Function	Equation	Graph
Sigmoid	$\frac{1}{1 + e^{-a}}$	
Hyperbolic Tan	$\tanh(a)$	
Linear	a	
Heaviside	$\begin{cases} -1, & \text{for } a \leq 0 \\ 1, & \text{for } a > 0 \end{cases}$	
Inverse Tan	$\arctan(a)$	
Leaky ReLU	$\begin{cases} 0.01a, & \text{for } a \leq 0 \\ a, & \text{for } a > 0 \end{cases}$	

Sigmoid, hyperbolic tan and inverse tan are all continuously differentiable activation functions, which is desirable for backpropagation as it requires the calculation of partial derivatives at each neuron. They all have the disadvantage of producing vanishing gradients because as they approach either end of the function the gradient becomes smaller, meaning changes in input in extreme regions will produce smaller changes in output than changes around 0. Although popular before the concept of ‘deep learning’, activation functions creating vanishing gradients are rarely used, as the problem of vanishing gradients increases for networks with more hidden layers.

Leaky ReLU, or leaky rectified linear unit, is an activation function which does not suffer from vanishing gradients. Even though it is piecewise linear, the function as a whole is non-linear and a combination of multiple ReLUs is also non-linear, so neural network layers with leaky ReLU

activation functions can be stacked. The leaky part of leaky ReLU refers to the small negative activation for negative inputs. This stops neurons with negative inputs ‘dying’, or producing zero activation indefinitely, which is a problem with normal ReLU, where all negative inputs produce a 0 output.

Although non-linear activation functions are required in the hidden layers of a neural network, classically for regression networks the output layer always uses a linear activation function, the identity. This is because some non-linear activation functions, such as the Heaviside function, produce only two values and so are suited to the final layer of networks used for classification. Also the characteristics of non-linear activation functions mean that certain values are more likely to be predicted if used in the output layer, this may be beneficial to some applications but is not explored in this thesis. Therefore, all neural networks in this thesis use a linear activation for the output layer.

Prior to training, the weight of the input to each neuron is initialised, this is often done randomly. There are multiple common ways to initialise network weights, such as random normal, random uniform, truncated normal, Glorot normal and Glorot uniform (Glorot and Bengio 2010). These all produce random numbers based on the stated distribution, where Glorot initialisers base the distribution parameters on the number of inputs. The only ship power prediction study to report which initialiser they use is (Liang et al. 2019) which uses a ‘normal’ initialiser.

A random normal initialiser is used in this study, with a mean of 0 and a standard deviation of 0.1. This was noted to produce lower errors than the truncated normal, and was straightforward enough to implement directly, allowing the seed of the random number generator to be controlled to produce fully deterministic results for method validation.

3.1.2 Optimisers

The learning rule, or optimiser, is the method by which the optimum weights are found during one training cycle. This is normally a gradient descent method which uses the gradients calculated by backpropagation.

Gradient descent, also known as steepest descent, is where after starting in a randomly initialised location, the gradient of the error surface is calculated and used to work out which direction to travel in to find the global minima. Once the direction of travel is calculated a step in that direction is taken, then the gradient is recalculated at the new location for a new direction and this process is iterated. Conjugate gradient descent uses the conjugate to the current gradient at each step instead of the calculated gradient in the new location ensuring that a step does not ‘go back’ to any of the previous steps (Møller 1993), this is illustrated in Figure 3.1. For normal conjugate gradient descent a line search, in the direction of the conjugate gradient, is performed

to find the optimum step size at every location. The scaled conjugate gradient descent bypasses the need for a computationally expensive line search by using a Levenberg-Marquardt approach to scale the step size by adding a scaling factor to ensure positive-definiteness of the Hessian.

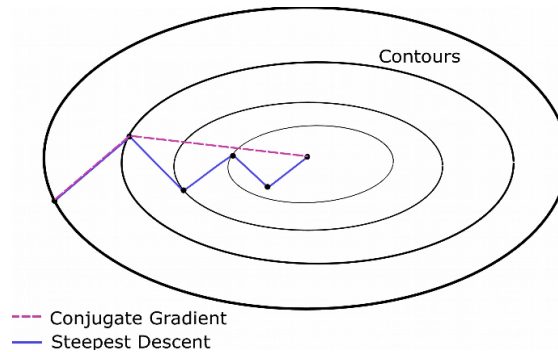


FIGURE 3.1: Illustration of conjugate gradient approach compared to standard steepest descent, where contours represent an error surface with the global minimum in the middle circle (Tasneem 2019).

More modern optimisers require specification of a learning rate as a way to control step size during the gradient descent. Larger learning rates create larger steps, meaning quicker convergence with the risk the optimiser will overshoot a minima. Multiple different variants of gradient descent exist. The follow section briefly describes each method. It is noted that all methods after AdaGrad build on the previous method exactly.

- (i) **SGD (Stochastic gradient descent)** - is a computationally efficient version of gradient descent, where the gradient is calculated on random subsets of the data to reduce memory requirements. It also reduces the chances of converging to a local minima as the different subsets should have similar global minima but different local minima.
- (ii) **Momentum** - has an adaptive learning rate which increases for steps pointing ‘downhill’, to increase convergence.
- (iii) **NAG (Nesterov accelerated gradient)** - calculates gradient based on approximate future position of parameters rather than current position, to improve convergence towards the end of the optimisation process.
- (iv.a) **AdaGrad (Duchi et al. 2011)** - each variable has its own adaptive learning rate, rather than one learning rate for all parameters. These learning rates are based on the sum of squared previous gradients, allowing a low learning rate to be assigned to regularly occurring features, and high learning rates to infrequent features. The downside is that the learning rate decreases regardless of whether a solution is found.
- (iv.b) **AdaDelta (Zeiler 2012) and RMSProp (Hinton et al. 2012)** - stores past gradients as a exponentially decaying average of the squared gradients, equivalent to the l_2 norm,

to avoid the problems in AdaGrad. RMSProp was developed independently, at the same time as AdaDelta and works in exactly the same way, also scaling the learning rate by a decaying average.

- (iv.c) **Adam (Kingma and Ba 2014)** - as well as storing exponentially decaying average of squared gradients stores an exponentially decaying average of (unsquared) gradients; similar to momentum terms. These decaying ‘moments’ are corrected for their natural bias towards 0, and then both used to define the learning rate at any given timestep.
- (iv.d) **AdaMax (Kingma and Ba 2014)** - AdaMax uses the l_∞ norm instead of the l_2 norm used in previous optimisers, as this produces more stable values, and a max term to remove the need to correct the bias towards 0 which is corrected in Adam.
- (iv.e) **Nadam (Dozat 2016)** - Adam with Nesterov momentum to improve convergence through the search.
- (iv.f) **AMSGrad (Reddi et al. 2019)** - incorporates SGD approaches with Adam. Is designed for specific scenarios such as image recognition where SGD outperforms state-of-the-art adaptive gradient methods such as AdaMax.

All optimisers have particular problems they are best suited to, i.e. size of network and size of dataset. Of the documented applications of neural networks to the problem of ship propulsion prediction a range of optimisers are compared, the majority being used in (Radonjic and Vukadinovic 2015), namely quazi-Newton and truncated Newton methods, resilient backpropagation, conjugate gradient descent, steepest descent and Levenberg-Marquardt. These optimisers are much older and less effective than the discussed SGD and Ada-family optimisers. Other studies concluded that no noticeable difference was found between applications of different optimisers.

In most ship power prediction studies the optimiser is not specified. The Adam optimiser is used in (Liang et al. 2019), for this reason, and after a brief study of the state of the art optimisers, AdaMax is used. The specific optimiser parameters: initial learning rate, β_1 , β_2 and ϵ are kept to their default, as the continued development of the Ada-family has the aim of reducing sensitivity to parameter selection. AMSGrad is not used, as the datasets used are noisier than many used to train neural networks, so extra stochasticity is not required to stop premature convergence.

3.1.3 Regularisation

There are many ways in which neural network training is regularised, such as weight decay, dropout and early stopping. Traditionally weight decay is the only method specifically called ‘regularisation’, all the above methods are discussed in this section as they all perform similar tasks.

Weight decay is used to stop gradients becoming too large, as this creates unstable learning, by penalising large gradients. If the error function is $E(w)$, with w the weights, then the regularised error function is $\tilde{E}(w) = E(w) + \frac{\lambda}{2} \sum_i w_i^2$. The parameter which needs tuning is λ ; a value close to 0 means there is little regularisation, or little weight decay, whereas a higher value, for example 0.2, would produce lots of regularisation and the training may be stunted. More specific constraints can also be put on the weights during training, for example not allowing any negative weights.

Dropout is an approach where a small random selection of neurons are temporarily ‘turned off’ during training. This means the weights are temporarily set to 0, so no output is produced and no learning can occur. After a set amount of time, such as a set number of batches or an epoch, the off neurons are turned back on and a new selection of neurons are suppressed. Neural networks using dropout have been shown to approximate Gaussian processes by using the stochastic nature and the principle of dropout to produce natural uncertainty estimations (Gal and Ghahramani 2016).

No weight decay or dropout is employed in the networks in this thesis, after trialling their use, networks without these types of regularisation consistently out performed those networks using them. The noisy characteristic of the dataset is suggested as a potential reason why regularisation did not improve performance. Explicitly for weight decay; noisy data creates a rough error surface which makes it unlikely for momentum-based optimisers, like Adamax, to produce exploding gradients where gradients are ‘saturated’ as activations are so large that a change in their value only produces an extremely small change in output.

Early stopping allows termination of training when a minimum error has been found. It requires the use of a validation dataset. A dataset is split into 3 partitions for training, validation and testing. Validation data is only used at the end of each epoch to validate whether the network has improved in accuracy compared to the last epoch. Note that no backpropagation occurs when using the validation data. The error from the validation run can then be used for an indicator for early termination of training.

Early stopping has two hyperparameters associated with it. Patience is how many epochs the network will wait, without the validation error improving, before terminating training. Tolerance is how much the error function must decrease to be classified as improving. Some networks use a validation error goal as an indicator of when to stop training as well as, or instead of, the early stopping procedure detailed above. If stopping criteria are used then ideally the criteria will terminate training rather than the maximum number of epochs being reached. This is preferential because the network is more likely to be close to the global minimum on the error surface if training is terminated by reaching a low validation error, as opposed to an arbitrary number of epochs being reached. The stopping criterion is specified to ensure the probability

that a better solution exists is minimal before termination. A large number of epochs is therefore chosen such that it is expected that it will not be reached.

The early stopping patience is set based on analysis of the validation error for each epoch, a patience which terminates training after minimum validation error is achieved but before it begins to increase as the network over fits to the training data. This number is based on the smoothness of the validation error curve, for the ship power application the optimum value from this analysis is 5.

3.1.4 Performance Assessment

After training a neural network, its capability of fitting the relationships within the dataset can be quantified as the average error of prediction. This must be calculated separately, using the testing dataset. Testing data should not be touched during any of the training or validation process. This untouched set is used after training has terminated, the test data is run through the network, and like the validation data, the error is calculated but no backpropagation occurs. The errors from all members of the testing set are averaged to give an indication of how well the network has been trained. These errors are statistically different from the errors calculated during the training process and the validation process as every error calculated in the training and validation process is used to change the network or training procedure in some way, so is no longer valid as a representation of how well the network performs.

While attempting to approximate, or fit, a dataset a network can under or overfit. Underfitting is where a network is too simple to model complex relationships so both testing and training errors are high. Overfitting is where the network replicated functions that are too complex for the data set as the intricate relationships present in the training data, that are not representative of the entire dataset, are modelled by the network meaning it fails to generalise well. This is characterised with low training error but high testing error, as the network fits the training data well but fails to generalise to any unseen data points. To combat this problem the regularisation methods discussed above are employed.

There are many error measures used to assess the performance of a trained neural network for regression. A more detailed study into these is provided in Section 4.1. The most common error measures used with neural networks for ship power prediction are Mean Absolute Relative Error, Mean Absolute Percentage Error, Root Mean Squared Error and Mean Squared Error.

As the data is scaled between 0-1 before being used for training, the Mean Absolute Relative Error is an appropriate error measure as all variables have the same effect on the overall value, it is also most frequently used in the data driven ship power prediction literature and therefore

allows benchmarking. The error values produced multiplied by 100 can be called percentage errors as they are pre-scaled by the range in shaft power, like the training output values.

A trained network can also be used to provide insight into the original dataset or provide predictions for similar data without target values. Values for the input variables where the dependent variable is unknown or uncertain are input into the network. The trained network then outputs the predicted dependent variable, for interpolated results it can be assumed the error of the predictions are no more than the test error calculated in the testing phase.

Benchmarking, or cross-validation, should then be performed to provide information about the predictions. Multiple different measures of error in prediction should be used, to ensure a full picture this includes descriptive statistics of multiple different error profiles. As well as benchmarking the method against other methods, saliency or sensitivity analysis within the method for example validating the use of certain hyperparameters or types of data can be useful.

3.2 Merchant Vessel Data

The data used for training the network in this study are from three large merchant vessels of the same design ('sister ships') from January 2014 to February 2016. Most of the vessel movement is repeated routes with occasional variations to the standard, thus covering a large range of geographic locations and recorded weather conditions. The total number of months worth of data is around 27, totalling 206,090 datapoints. Data is recorded as a point value every 5 minutes on board for each variable apart from wave data, which is hindcast weather data collected at the same intervals. No aggregating over the timestep is performed, which reduces error caused by averaging. The aim is to predict the shaft power (MW) of the vessels in all weather and operating conditions, given input variables recorded on-board.

3.2.1 Variable Selection

Shaft power is the measure of how much power the engine transmits to the propeller via the shaft. Measuring this is a more direct measure of power required by the vessel than the quantity of fuel used, as this is affected by engine performance. An accurate estimation of required power in given conditions allows optimal vessel operation for reducing engine power.

Shaft power is the product of the shaft torque (T) and its angular velocity (ω), equation 3.1, which can also be expressed in terms of the RPM of the engine (N), equation 3.2,

$$\text{Shaft power} = T\omega, \quad (3.1)$$

$$= \frac{2\pi NT}{60}. \quad (3.2)$$

Over 100 variables are recorded on board including: engine, cargo, vessel condition and vessel movement data. A detailed study into variable selection for shaft power prediction has been performed, (Parkes et al. 2019), where data quantity was prioritised. Of the 100 input variables, nearly half of the variables did not measure correctly for over 30% of the time, leaving a set of 43 usable variables for prediction. In this study, 3 different sets of input variables are compared. In this study, 3 different sets of input variables are compared. The first set is 6 variables selected from a Naval Architecture perspective:

- (i) GPS ship speed (knots);
- (ii) wave height (m);
- (iii) true wind speed (m/s);
- (iv) apparent wind direction (degrees);
- (v) draught (m); and
- (vi) trim (m).

This set of six is compared to two other input variable selection methods: incrementally increasing the quantity of input variables based on their correlation to shaft power and incrementally increasing the quantity of principal components from a Principal Component Analysis (PCA) of the full usable dataset.

The 6 most highly correlated variables to shaft power are:

- (i) GPS ship speed (knots);
- (ii) relative wind speed (m/s);
- (iii) gas methane content (%);
- (iv) astern response;
- (v) compass heading of the course over ground (deg); and
- (vi) fuel oil boiler density.

The study notes minimal difference in prediction accuracy between each of the approaches. This thesis uses the 6 Naval Architecture selected variables discussed in (Parkes et al. 2019). As domain knowledge identifies that these 6 have a causal connection to the output; a change in

any one of them causes an increase or decrease in required shaft power. Since how closely the ground truth relationships are modelled by networks are of interest, the causally related input variables will produce the most relevant results.

Of these variables the ship speed, wave height, wind direction and draught are measured directly on board, whereas the draught, trim and wind speed are derived from measured variables. The true wind speed (V_{true}) is derived using the apparent wind speed (V_{app}) and ship speed (V_{ship}) in equation 3.3 using the cosine rule,

$$V_{true} = \sqrt{V_{app}^2 + V_{ship}^2 - 2V_{app}V_{ship}} \quad (3.3)$$

The draught and trim are derived from measurements of the draught at the front (fore) and the draught at the back (aft) of the vessel, equations 3.4 and 3.5:

$$\text{draught (m)} = \frac{\text{draught}_{fore} + \text{draught}_{aft}}{2}, \quad (3.4)$$

$$\text{trim (m)} = \text{draught}_{aft} - \text{draught}_{fore}. \quad (3.5)$$

3.2.2 Filtering

Operational data contains erroneous datapoints from sensor errors or inaccuracies. Although high frequency data has been shown to have significantly less uncertainty than noon report data (Aldous et al. 2015), erroneous datapoints increase uncertainty within the dataset. All literature relevant to vessel power prediction applies manual, or extensive, trimming processes to ensure a minimal quantity of erroneous datapoints in the dataset used for training a neural network. This process is often time consuming and since neural networks are robust to low levels of Gaussian noise, can be unnecessary for some applications.

Therefore, minimal filtering is applied to the data, only removing datapoints with a shaft power equal to zero. As these measurements are observed across the entire range of ship speeds, Figure 3.2a, they are clearly erroneous. The filtering performed does not aim to remove all erroneous datapoints, as identifying them is non-trivial and changes from ship to ship. The cluster of datapoints between 5-11 knots and 10-15MW exemplifies the complexity of the dataset, representing times of extreme weather. Therefore it is time consuming, and becomes infeasible when scaling the process to fleets of ships. After this filtering all the variables retain their distribution shape, with a reduced total dataset size of 142,196.

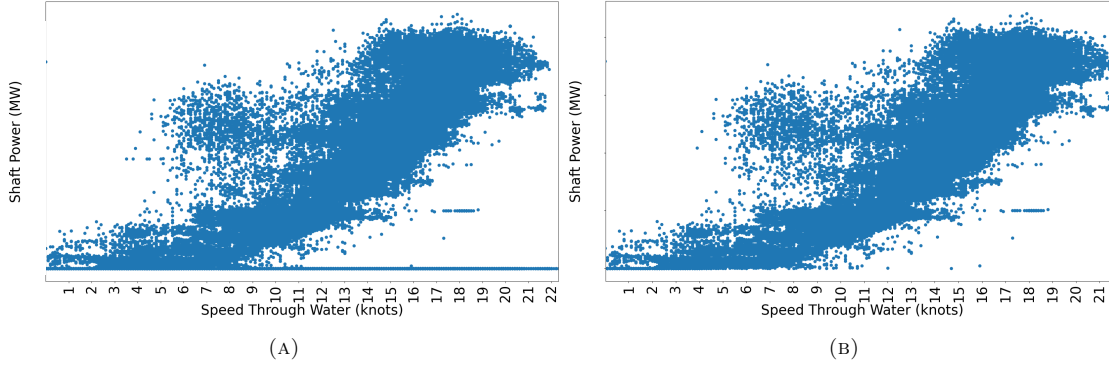


FIGURE 3.2: Scatter graph of speed through the water and shaft power of the dataset (A) before any trimming, (B) after removing all datapoints where shaft power is 0.

3.2.3 Data Analysis

The relationships between input variables and shaft power are briefly analysed to contextualise the dataset. The widely accepted approximate relationship between speed through the water and shaft power is $y = kV^n$ with y the shaft power, V the ship speed and k and n to be derived from a regression of tow-tank or heavily filtered data (Molland et al. 2011). The value of n is normally around 2.7-3, so a coarse approximation of the relationship is a cubic polynomial.

The dataset could approximate this cubic relationship but with significant spread, Figure 3.2b. Although it is difficult to ascertain the density of points from the figure, and therefore the significance of this spread. For each input variable, the dataset is partitioned into 150 equally spaced bins across the range of observed input values. These partitions are plotted as vertical boxplots showing the distribution of shaft power values within the partition as the y axis. The boxplot is placed at the midpoint of the partitions' edge values on the x axis, Figures 3.3-3.9.

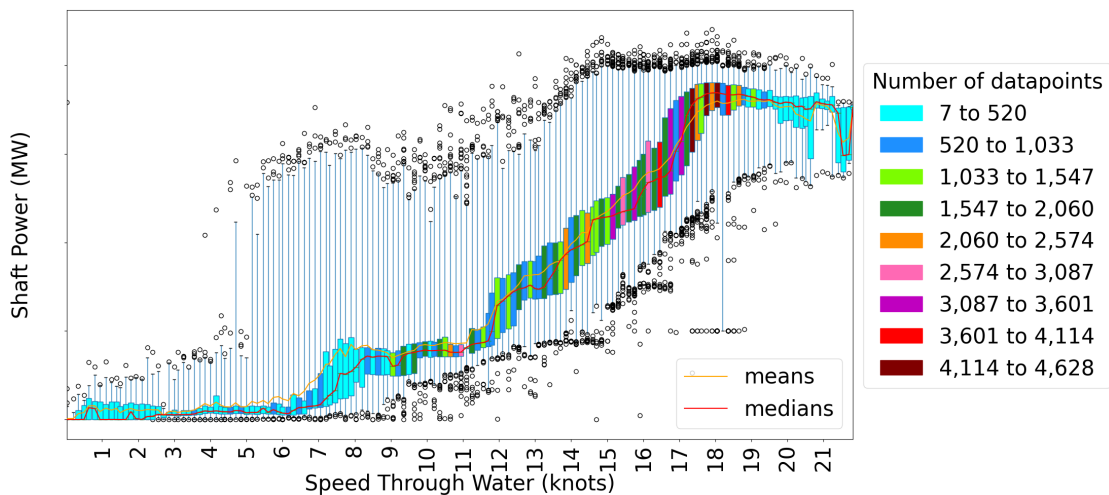


FIGURE 3.3: Box plots for every recorded value of shaft power at each recorded speed after trimming

The trend of the average power being approximately proportional to the cube of the speed is apparent. The middle darker boxes are bounded by their upper and lower quartiles and so contain half the datapoints for each plot, Figure 3.3. However, the variation in the data is also clear, especially from 5-13 knots of ship speed where the variation of shaft power is two thirds the range of the observed shaft power and from around 11 knots of ship speed upwards there is a variance of nearly half the range. This demonstrates the difficulty in creating a linear relationship between the input variables, such as speed, and power. Much of the data fits simplified relationships, as speed is so dominant, but will give inaccurate results at other points due to the variation caused by other factors.

The distribution of ship speed is also illustrated by the colouring of the boxplots in Figure 3.3 where most of the data is in the 14-20 knot range, with the values of 16 knots and above with the highest densities. This further demonstrates the difficulty in modelling the ship speed to power relationship across the full range of ship speeds, as some of the lightest coloured boxplots have as few as 7 datapoints in their corresponding partition.

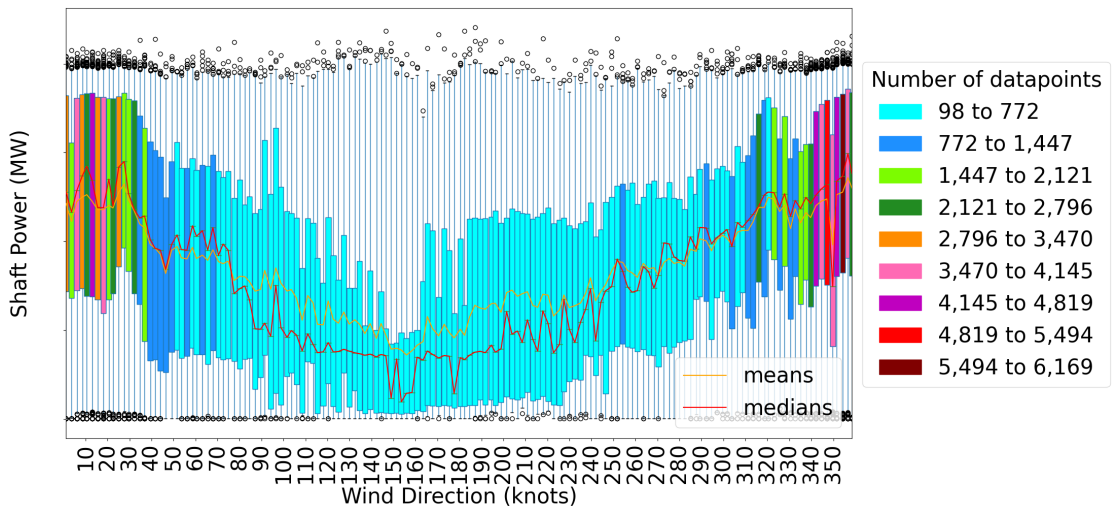


FIGURE 3.4: Box plots for every recorded value of shaft power for each recorded wind direction

For the other input variables, the majority of the boxplot whiskers span the entire range of observed shaft power, Figures 3.4-3.9. This is because the most highly correlated variable to shaft power is the ship speed, so this will overpower the effects of the second order variables in most cases. However, trends in the inter quartile ranges and averages illustrate the isolated relationships between other input variables and shaft power. For example the wind direction plot shows that there is a clear relationship between wind direction and shaft power, that higher powers are correlated to head-on winds of 0-30 and 320-360 degrees, Figure 3.4. This illustration of the isolated relationship captures interactions with other input variables, and is therefore slightly misleading. As some of the head-on wind instances are created by the ship moving faster

than the wind. The indication that this phenomenon may occur is that the only area of dense data in the wind direction domain is the head-on winds.

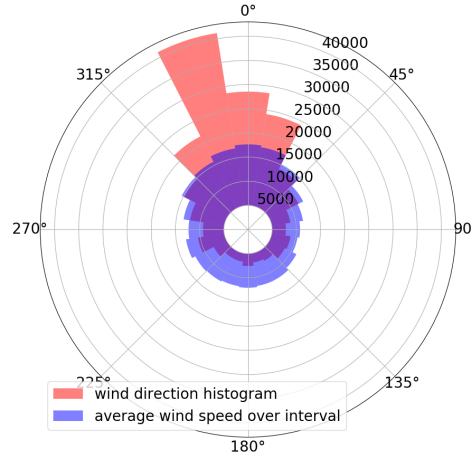


FIGURE 3.5: Wind direction histogram and average wind speed in each histogram bin

Comparing wind direction and wind speed frequency using a polar plot, Figure 3.5, illustrates that the average wind speed over each wind direction interval is almost constant, with a slight increase for head-on winds. The correlation between high wind speed and head-on winds is another indicator that some of the head-on wind instances are created by the ship moving faster than the wind. It is also clear that the most frequent wind direction is just off head-on winds at around 350 degrees, which agrees with Figure 3.4.

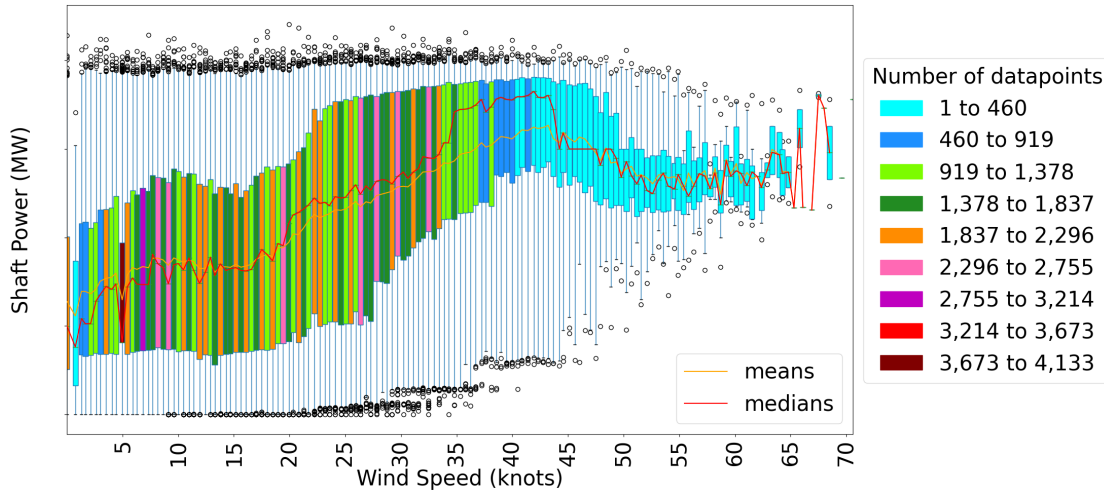


FIGURE 3.6: Box plots for every recorded value of shaft power at each recorded wind speed

The boxplot whiskers span the entire range of shaft power for changing wind speed up to 35knots, Figure 3.6. For higher wind speeds the whiskers range decrease steadily, although these boxplots all hold less than 0.3% of the overall dataset each. The sparsity of data means that due to the

‘law of small numbers’ it is hard to draw conclusions from this trend. This demonstrates the requirement for data visualisation which illustrates data density in both dimensions.

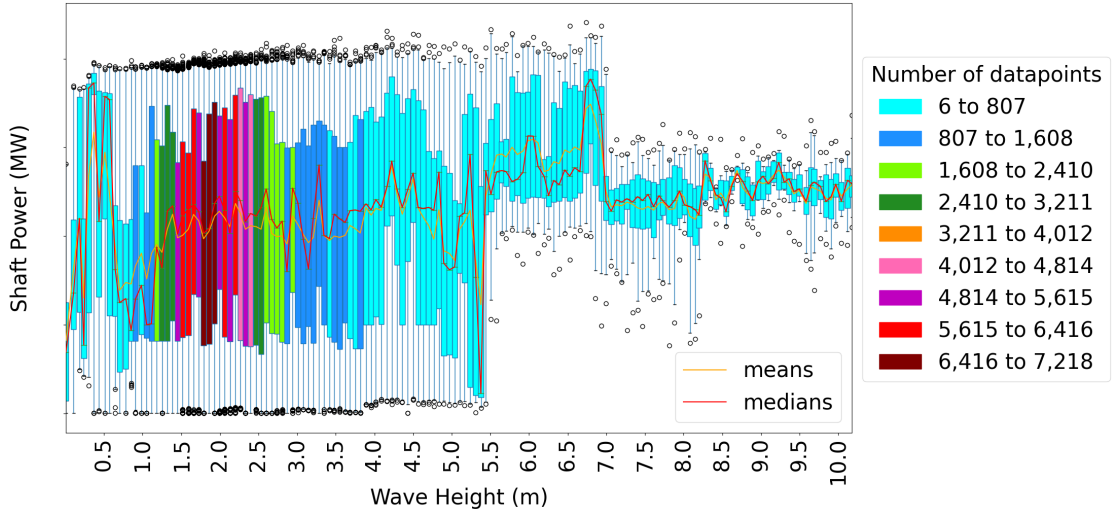


FIGURE 3.7: Box plots for every recorded value of shaft power at each recorded wave height

A similar trend of boxplot whiskers spanning the entire range of shaft power for boxplots containing larger quantities of data, is apparent across the range of observed wave heights, Figure 3.7. The range of boxplots with more than 0.5% of the dataset is 1-4m and the total range of wave height is 10m. Over half the range of observed wave heights have a very low density of data. The prevalence of areas of sparse data is apparent for many of the continuous input variables; wind direction, wind speed, wave height and to an extent ship speed, as the lightest colour boxplots span significant ranges of the relevant input variable domain.

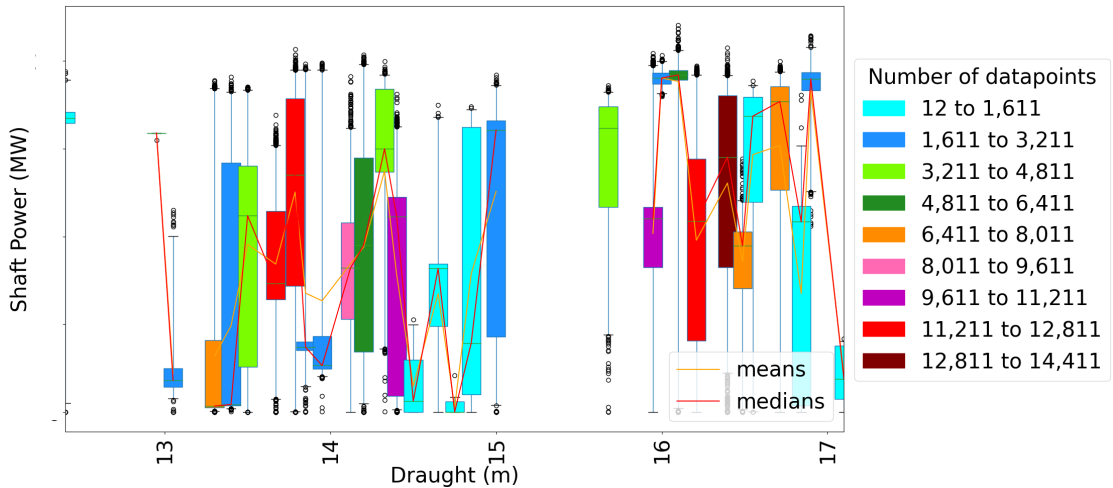


FIGURE 3.8: Box plots for every recorded value of shaft power at each recorded draught

The first four input variables discussed above all have continuous distributions, whereas the final two, draught and trim, show near-discrete distributions. The draught of a vessel is the length

from the bottom of the hull to the waterline or how deep in the water a ship is. Although theoretically this is a continuous variable, the vessels operate in either a loaded or ballast condition, therefore the draught tends to be around one of two values, Figure 3.8. Variation of less than 2m is observed in each condition, with the split for these vessels around 15.5m of draught. More traditional naval architecture approaches to empirical shaft power prediction split datasets into ballast and loaded and remove the draught dimension from the datasets. This causes two problems, the loss of information from creating a coarse measure of draught and the potential for unevenly sized datasets, for financial reasons it is preferable to operate a loaded vessel so more data is collected in these conditions.

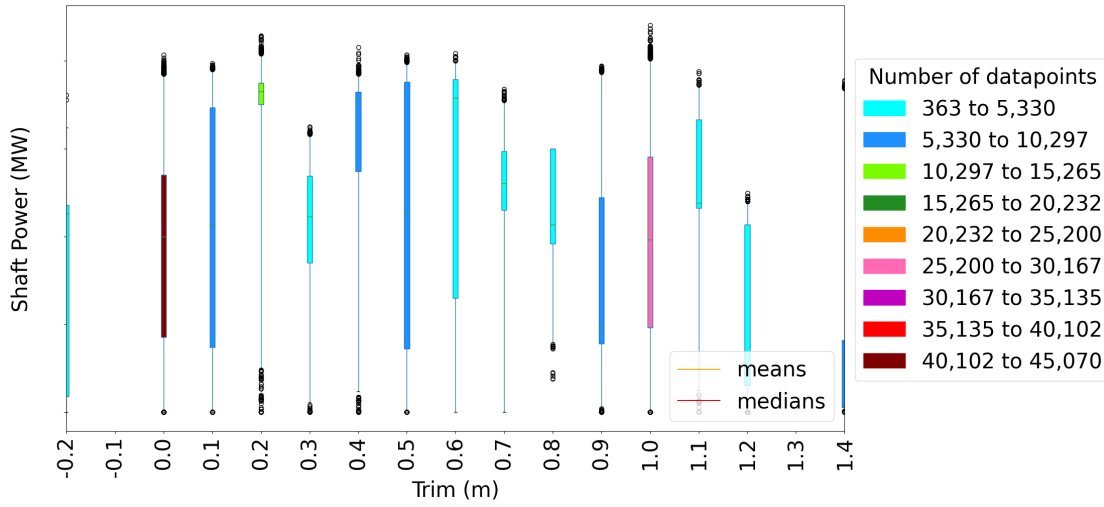


FIGURE 3.9: Box plots for every recorded value of shaft power at each recorded trim

Trim is a measure of what angle in the water a vessel is but is computed as the difference between the draught at the fore and the draught at the aft of the vessel so it takes units in metres. Unlike draught, trim can be altered by the captain at any time by pumping sea water in or out of ballast tanks. Positive trim refers to conditions where the aft of the vessel is lower in the water than the fore, so negative trim is rare as this corresponds to a vessel with its propeller closer to the surface of the water than the bow of the ship, which is potentially unsafe.

No trends between draught, trim, and shaft power are clear; Figures 3.8 and 3.9. Little is fully understood regarding the relationships between draught, trim, and propulsion of a vessel as draught and trim are closely related. Their relation to power requirements is also heavily dependent on the hull form of a vessel, which is complicated to describe, and often unique for every vessel, or at least class of vessel.

3.3 Model Selection

The neural network methodology detailed in Section 3.1 is used to predict shaft power for merchant vessels. An initial investigation into the most fundamental hyperparameters, number of layers and size of the layers, is conducted to provide initial scope. The mean, testing, Mean Absolute Relative Error of 5 repeats for varying size networks is documented, Figure 3.10. These network runs are identical apart from stochastic elements of the training process. Specifically the initialisation, the order the data is fed to the network, and the stochastic elements of the optimiser, Adamax (Kingma and Ba 2014). Combinations of 1-4 hidden layers and 1-500 neurons in each layer were tested, this decision was made based on the size of networks applied to similar problems in the literature. The number of connections in these networks ranges from 7 to 753,500. Given the dataset has 142,196 datasets after trimming, this range of networks is deemed appropriate. The training-testing-validation split is 70:15:15, and data is scaled between 0-1 before training. Multiple error measures are calculated but for conciseness only the Mean Absolute Relative Error is reported, as all error measures calculated show similar trends.

Error decreases as both numbers of layers and numbers of neurons increase, Figure 3.10, as the larger complexity of the network allows for a more flexible model of the data. Errors of around 2% are straightforward to achieve using any number of basic networks and limited treatment of the data. Although networks with the largest number of connections (4,500) produce the lowest average error of $(1.68 \pm 0.1)\%$, these are similar to the (4,300) network, $(1.76 \pm 0.05)\%$, and the (3,300) and (3,500) networks, $(1.96 \pm 0.1)\%$ and $(1.92 \pm 0.1)\%$.

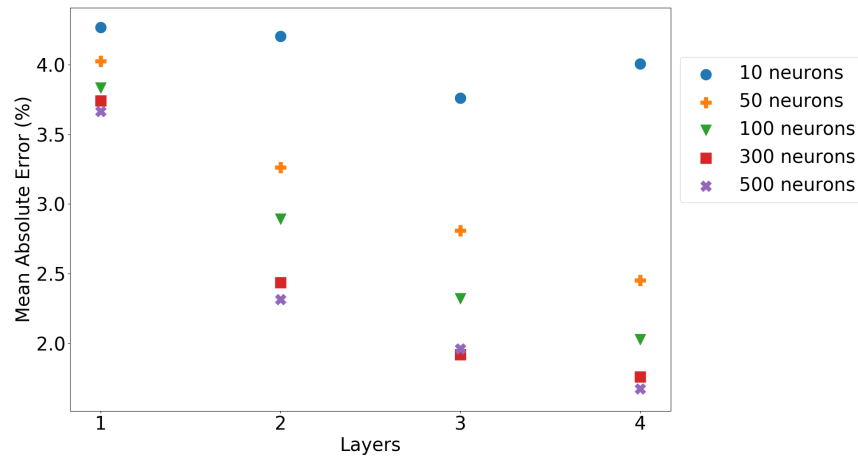


FIGURE 3.10: Mean Absolute Relative Error in power prediction for a range of network sizes, each point is the average of 5 individual network runs.

It has been noted that although theoretically proven that any continuous function can be approximated by a neural network with one layer and an unspecified number of neurons, these results support the speculations in Cybenko (1989)—that the number of neurons necessary to

approximate many functions will be far more than is feasible to use. It appears that using more than one hidden layer increases the modelling capabilities as expected, and errors reduce accordingly with more than one hidden layer, Figure 3.10.

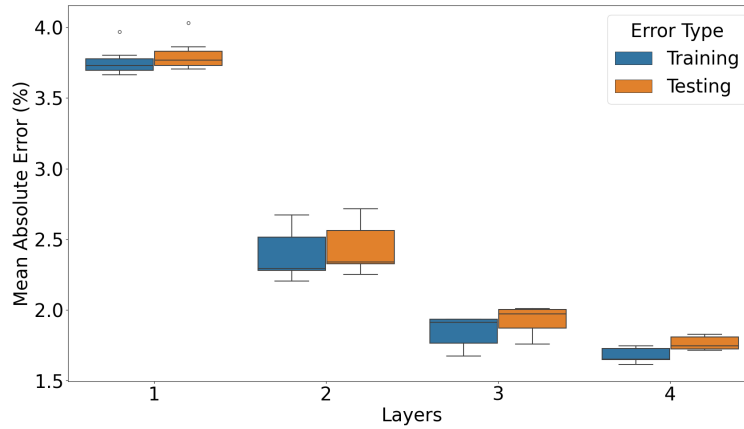


FIGURE 3.11: Mean absolute relative training and testing error in power prediction for networks with 300 neurons for a range of layers.

Testing and training errors are necessary to identify underfitting or overfitting. The difference in training and testing error distribution increases as the number of layers increase, Figure 3.11. For 1-3 hidden layers the interquartile range of training and testing errors overlap by more than 50% of the range, which suggests that networks of this size do not overfit the data. For networks with 4 hidden layers the interquartile range of errors overlap by only 0.004% error, or 4% of the boxplot interquartile ranges. This separation of training and testing errors could be an indicator that overfitting is beginning to occur for networks with more than 3 layers.

Prediction curves are used for further analysis of the affect of using different size networks. These curves show the learnt relationship between independent variables and the ship powering from a trained network, they are illustrated alongside the data visualisation technique used in Section 3.2 to aid the analysis, Figure 3.12. To visualise how the networks model the relationships within the data, prediction curves are produced with the following procedure:

1. train the network,
2. set all but one value to be constant, the modes,
3. cycle the remaining variable from its minimum to its maximum recorded values with 150 points evenly spaced along the domain,
4. run new dataset through the trained network.

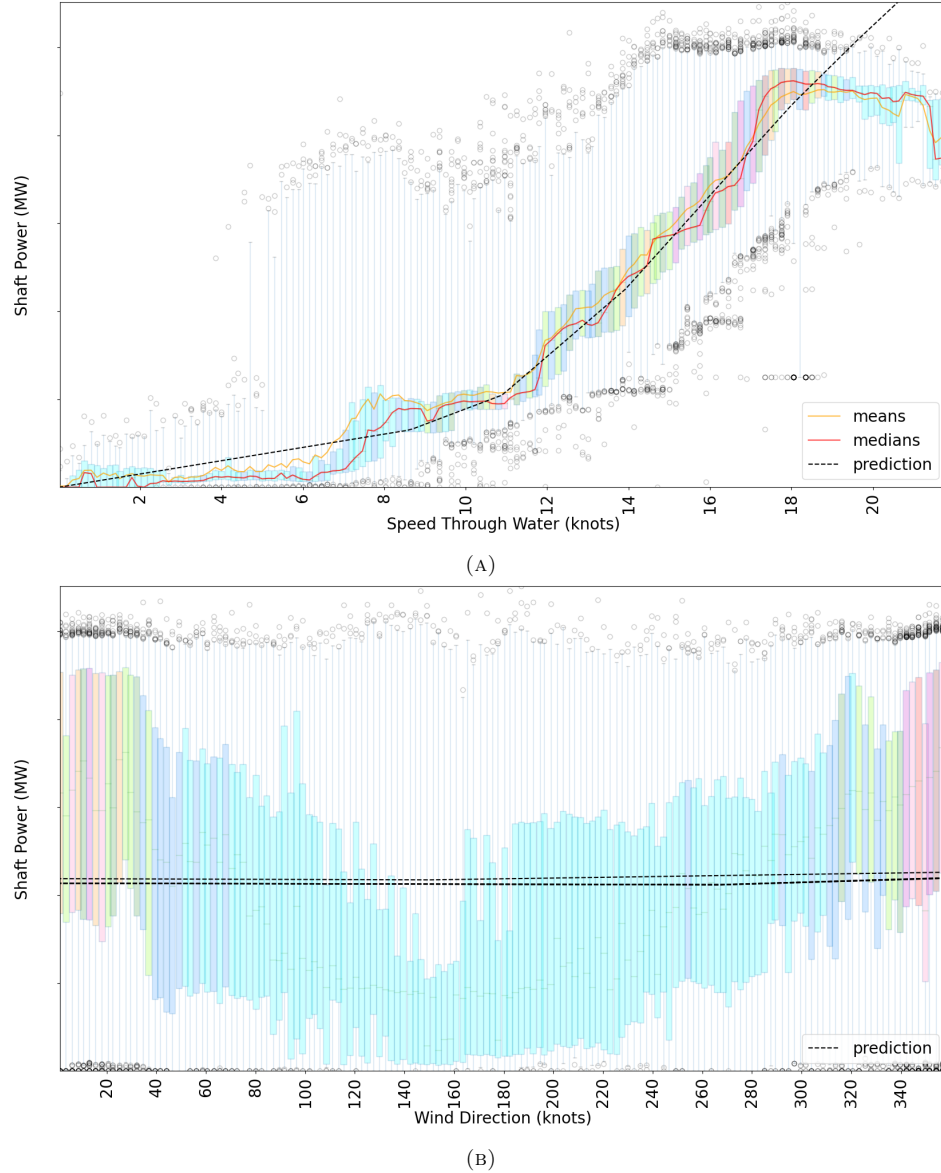


FIGURE 3.12: Power prediction from a neural network with 1 hidden layer and 10 neurons A) one network prediction of isolated speed alongside visualisation of the mean and median shaft power conditioned on the speed, to illustrate the accuracy of the predicted relationship and B) multiple network predictions for isolated wind direction, without conditional averages, to illustrate the consistency of predicted relationships.

Saliency analysis is employed to explore repeatability, extrapolation power and performance in regions of low density data. This is an indicator of whether the size of the network is appropriate, and so can be used as another hyperparameter tuning method. The more a network overfits a dataset the more likely it is to produce inconsistent results, as slight changes in the dataset will change the curves more dramatically. For each network shape, multiple networks are trained from the same dataset, the differences in prediction are therefore caused only by the stochastic elements of the training process. A reliable network should map the same line each time, and assuming the network has sufficient complexity to model the ground truth, this is a good indicator that it is the true relationship between the variable and the output.

Networks of size (1,10), produce errors over 4% and model the main characteristic of the speed-power curve approximating the middle of the areas of dense data but fail to approximate the data in more sparse regions of the ship speed domain, Figure 3.12a. Specifically for ship speeds between 0-7 knots the prediction is above the 75th quartile of powering data and from 7-11 knots the prediction is below the 25th quartile of powering data. However, the relationship modelled for ship speeds over 18 knots is logical, unlike the data observed in this region, which is 4MW lower than predictions. The expectation from Naval Architecture theory is that shaft power increases approximately as the cube of the ship speed, but with variations about this trend caused by wave interference between bow and stern wave patterns (Molland et al. 2011). These variations in draught and trim will affect wave patterns which will have an effect on the powering of the vessel, causing ‘humps and hollows’ specific to each vessel type and operational profile. The ‘humps and hollows’ are notable from the conditional averages of the data, but not from the network prediction of this relationship.

The wind direction-power curves produced from 5 separate network runs are shown together, Figure 3.12b. Although 5 networks approximate similar linear wind direction-power relationships, they are split into two distinct parallel trends, with a different of around 0.5MW across the entire wind direction domain. The physical relationship between wind direction and power requirements of the ship is a parabola, with head winds requiring more power than tail winds, which is illustrated by the observed data, Figure 3.4. The near-linear predictions from these networks suggest that the inflexibility of a small network does not provide adequate modelling capabilities to approximate the wind direction-power relationship. The fact that from 5 networks all predictions are near identical, although with equally poor estimates, suggests that from the limited set of relationships which could be modelled, each run converges to one or the other arbitrarily. Which relationship any given network converges to is determined by a small initial bias, as the only difference between each network run is the inherent stochastic elements of training.

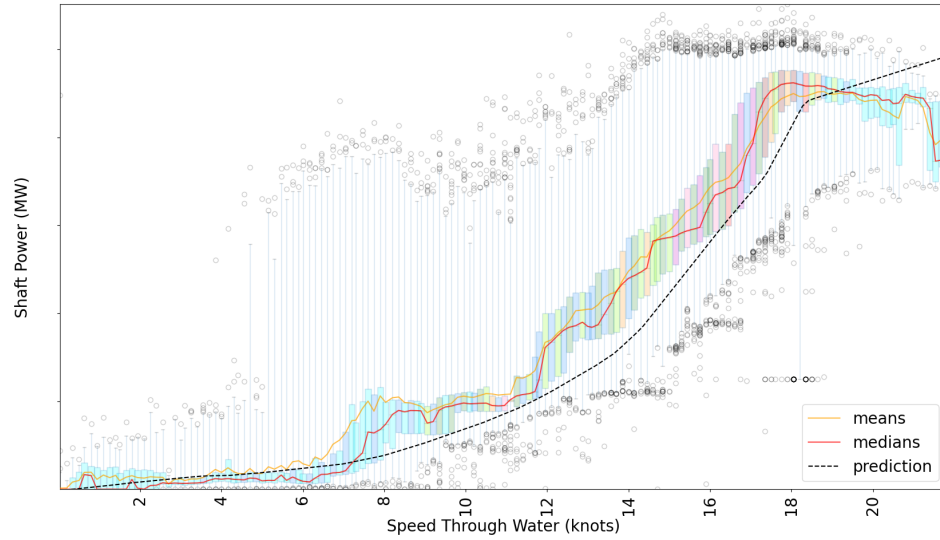


FIGURE 3.13: Power prediction from a neural network with 2 hidden layers and 50 neurons illustrating isolated speed alongside data visualisation

The predictions from networks of size (2,50) have more connections and therefore can model more complex patterns. The predicted speed-power curve for these networks models the area of sparse data in the speed domain from 0-6 knots, Figure 3.13. However the prediction is further away from the conditional averages of the data for the dense areas of data; power is predicted up to 2MW below the 25th quartile of observed data for speeds from 6-18 knots. This illustrates a similar modelling inflexibility as shown in (1,10) networks, although to a lesser degree.

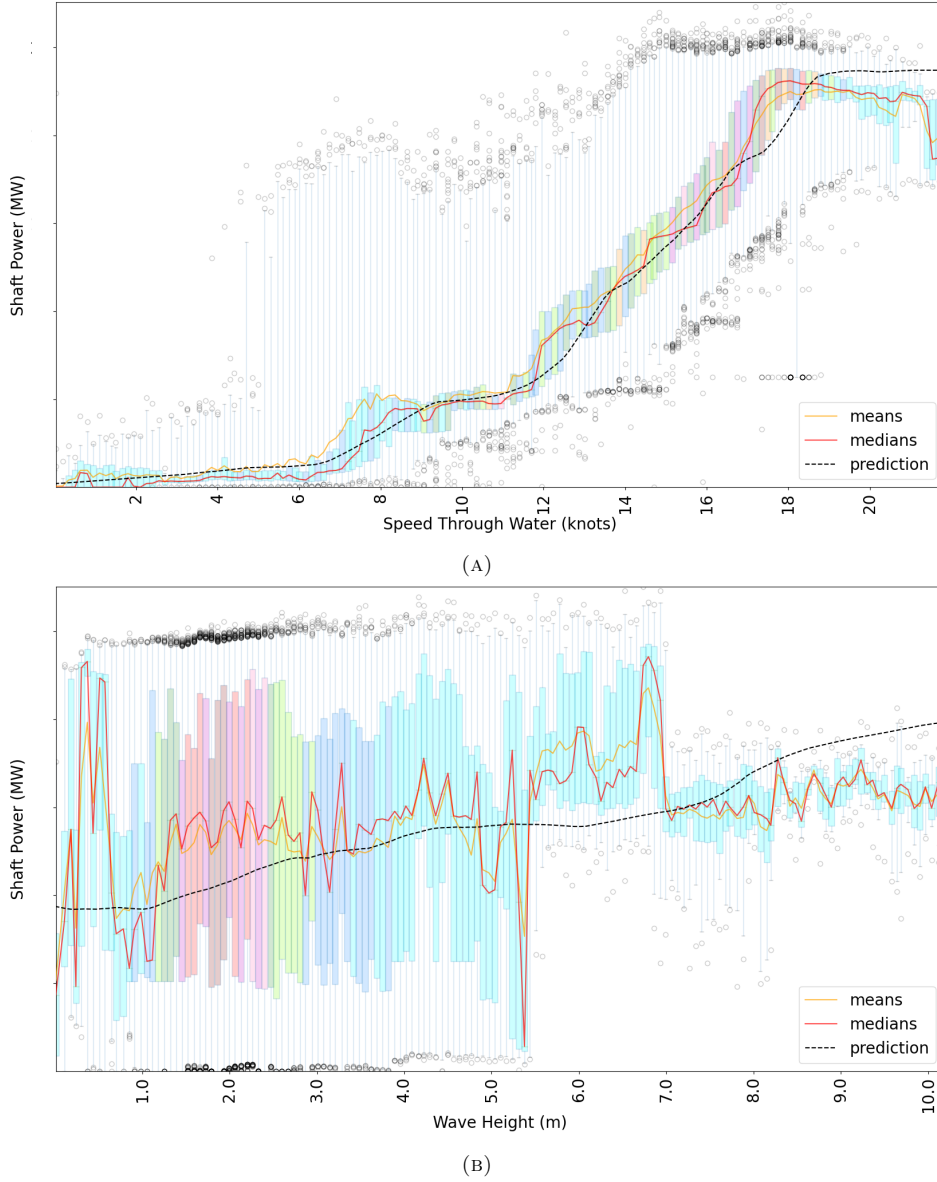


FIGURE 3.14: Power prediction from a neural network with 3 hidden layers and 300 neurons A) isolated speed-power curve and B) isolated wave height-power curve alongside data visualisation

Networks of size (3,300) produce errors in line with the largest networks tested, Figure 3.10; have a significant overlap of training and testing errors, Figure 3.11, so are unlikely to overfit the dataset; and the speed-power curve models the conditional average of dense areas of data as well, Figure 3.14a. This indicates that networks of size (3,300) produce good approximations of the input-output relationships within the dataset. The network also approximates relationships closer to the conditional averages of other input variables compared to smaller networks; such as the wave-power curve, Figure 3.14b. This approximates a realistic pattern, increase in wave resistance increases power requirement, while fitting the average of the dataset fairly well. However, this network still models a relationship between wave height and power requirements for

waves over 7m that is far from the conditional average, outside the inter-quartile range of the data.

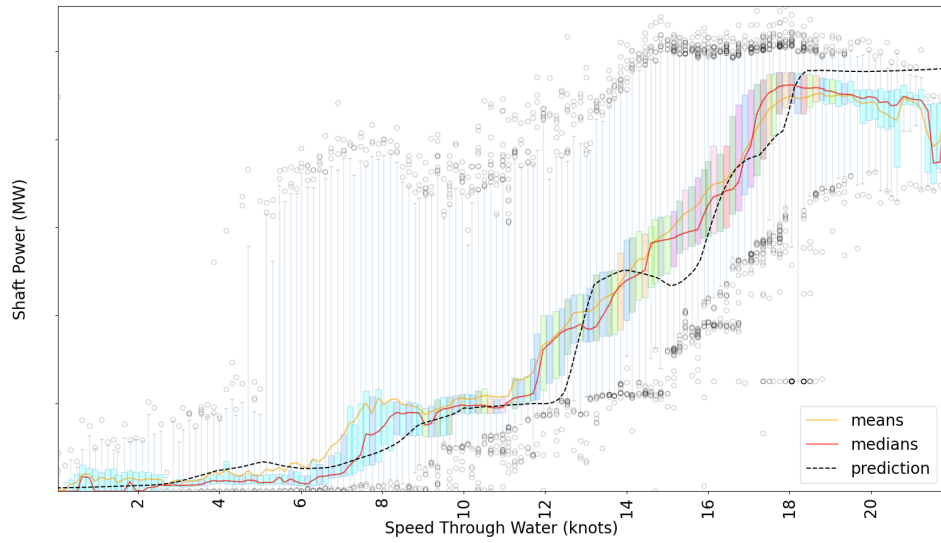


FIGURE 3.15: Power prediction from a neural network with 4 hidden layers and 500 neurons illustrating isolated speed alongside data visualisation.

The use of networks of size (3,300) for further analysis is supported by the speed-power curve produced from the largest network trialled (4,500), Figure 3.15. This curve approximates the average of the data for the majority of the speed domain, but has exaggerated ‘humps and hollows’ at 13 and 17 knots. These extend below the interquartile range of the observed power data, so are suggested as indicators of overfitting; since the networks model intricacies specific to the training data, which do not generalise to the testing data. The speed-power curve profile along with the separation of training and testing error distributions suggest networks of this size are overfitting the dataset.

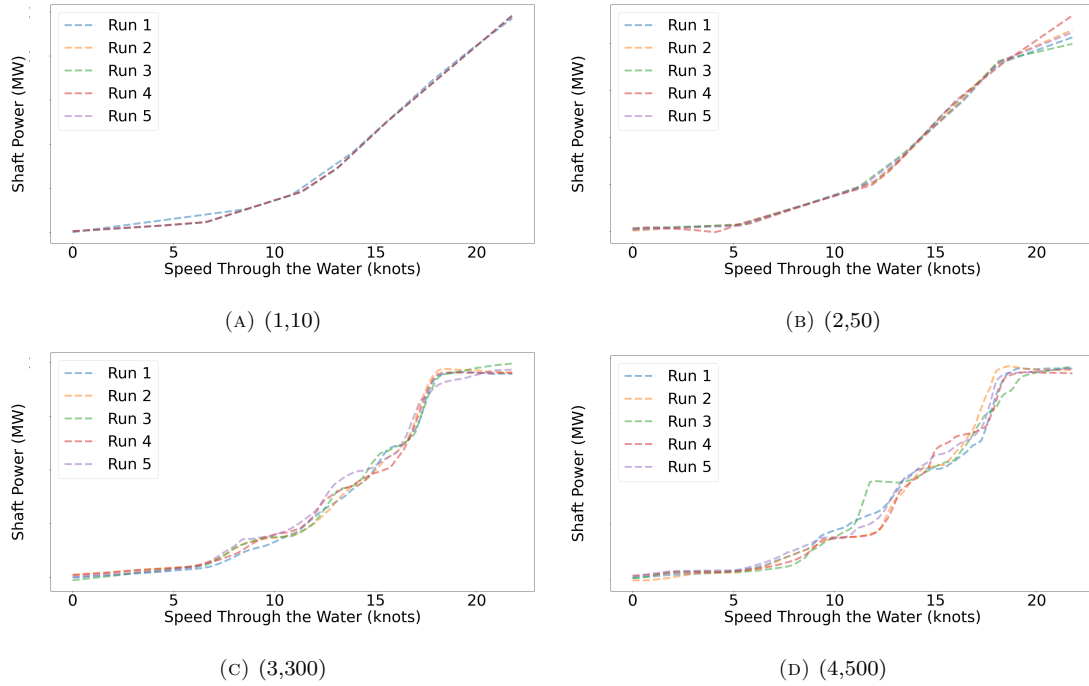


FIGURE 3.16: Predicted isolated speed power curves for 5 separate runs for differently sized networks (A) networks with 1 layer and 10 neurons, (B) networks with 2 layers and 50 neurons, (C) networks with 3 layers and 300 neurons, and (D) networks with 4 layers and 500 neurons.

Predicted speed-power curves from networks of size (1,10) are consistent and piece-wise linear, Figure 3.16a, which follow the expected extrapolation pattern of increased power requirement for increased speed above 18 knots¹. The consistency in prediction is suggested to be caused by the limited set of curves available to describe the system, due to the small network size. Networks of size (2,50) produce speed-power curves which approximate the conditional average of the shaft power better those from (1,10) networks, Figure 3.16b. There is less consistency between predictions, as extrapolated predictions over 18 knots range by 3MW, which is a difference of 13.7% of the observed shaft power range. This models a more realistic scenario than the observed data; it is expected higher power should always be required for faster ship speeds given these vessels are 300m merchant craft so do not plane. It is suggested that this inconsistency is due to the larger range of modelling capabilities. There are more functions available which produce low Mean Absolute Relative Error but do not necessarily correspond to modelling the ground truth relationships within the dataset.

The computational time to train the network increases substantially with increasing neurons and layers. Networks of size (4,500) take an average of 13 minutes 48 seconds to train², which is over double the time for networks with 200 less neurons per layer, which take an average of 4 minutes

¹The region in the ship speed domain with speed greater than 18 knots is not technically an area for network extrapolation, as there is measured data observed in this region. However the very sparse nature of this part of the speed domain means it shares many similarities with areas of extrapolation and for ease of writing, it will be called a region of 'extrapolation' in the rest of this thesis.

²On a desktop PC with 16GB RAM and i7 processor at 3.40GHz.

TABLE 3.3: Neural network model selected for ship power prediction

Hyperparameter	Value or set
Number of hidden layers	3
Number of neurons in each hidden layer	300
Epochs	1,000
Early Stopping Patience	5
Early Stopping Tolerance	0
Loss function	Mean Absolute Relative Error
Performance Measures	Mean Absolute Relative Error and Relationship Visualisation
Optimiser	AdaMax (Kingma and Ba 2014)
Learning rate, β_1 , β_2 , ϵ	0.001, 0.9, 0.999, 1^{-7}
Activation Function	ReLU
Regulariser	None
Dropout	None
Initialiser	Random Normal ($\mu = 0, \sigma = 0.1$)

38 seconds. The use of networks of size (3,300) is supported by these training times; since networks with 300 and 500 neurons produce almost indistinct error values, it is more efficient to use the lower number of neurons.

It is suggested that the variability in networks of size (4,500) is caused by the high quantity of curves available to model which produce a low Mean Absolute Relative Error, therefore the network models one of these possible curves arbitrarily. This produces results which do not model the input-output relationships within the dataset well. In this region, Figure 3.16d, run 3 produces a curve with a high concavity, almost approximating a step function, whereas run 4 produces a curve with high convexity. Although the ground truth relationship between ship speed and power is not fully understood, the variation in these predictions show that not all can approximate the ground truth accurately. All predictions from networks with shape (4,500) have Mean Absolute Relative Errors between 1.60-1.88%, these are the lowest errors produced by any network trialled. Therefore, it is suggested that the Mean Absolute Relative Error measure is not providing a full picture of how well a neural network models the underlying input-output relationships in dataset.

Networks of size (3,300) are therefore chosen to be used for further analysis. This is based on a combination of fundamental theory, domain knowledge and substantial analysis of input-output relationships modelled by ranging sizes of network. The final parameters for the networks used in proceeding sections are summarised in Table 3.3.

3.4 Benchmarking

The neural network method with chosen size (3,300) is benchmarked in two different ways. It is directly compared to a regression method applied to the dataset. As well as this, the quantity of data required for accurate predictions is investigated.

3.4.1 Agreement With Traditional Methods

The curves derived from two regression approaches common to traditional ship power prediction from operational data, are compared with simulations from the neural networks on the same dataset, Figure 3.17. The first approach is a polynomial regression, where the data is assumed to take the form $y = kV^3$ where y is the shaft power, V the ship speed, and k the coefficient to be fit. The second regression approach is polynomial of the form $y = kV^n$, where both the coefficient k and the power n are to be fit—this method is considered better practice in Naval Architecture than the first, although the cubic polynomial is still common. Other approaches are to use extra terms in the regression such as $y = kV^n + aV^2 + bV + c \tanh(V)$ (Uyanık et al. 2020) which adds complexity, yet does not improve the modelling significantly.

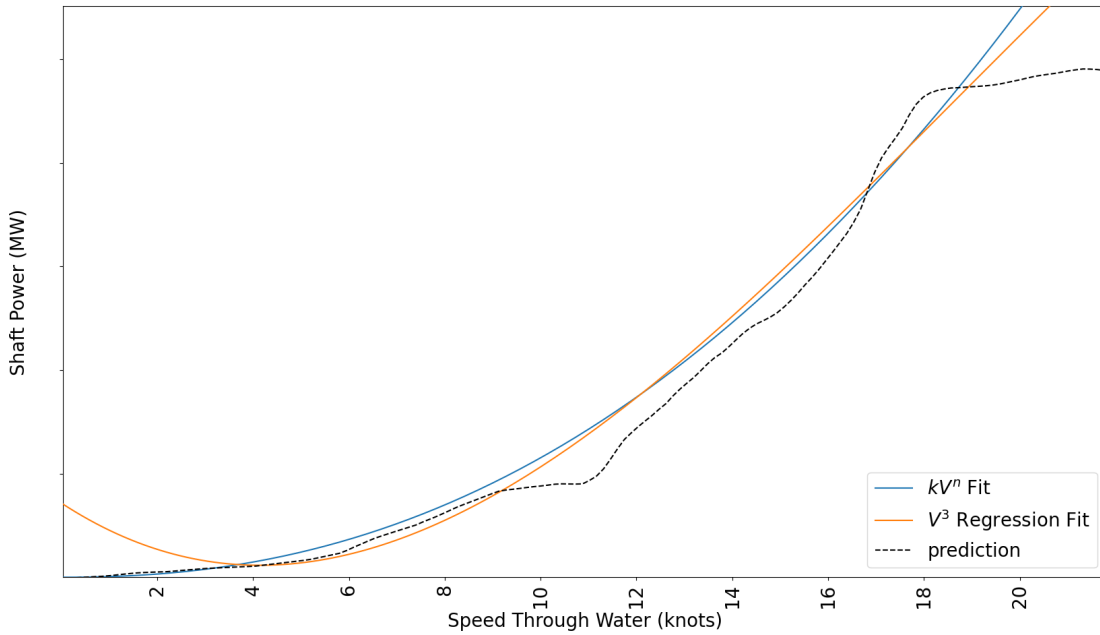


FIGURE 3.17: Neural network of shape (3,300) prediction curve, compared to a cubed and a fitted power regression where the fitted power is 2.2494.

The two regression curves agree across the majority of the speed domain, diverging for speeds less than 4 knots and more than 18 knots. Notably, the cubic regression models an infeasible pattern for less than 4 knots of ship speed, as slower speeds should require less power. This is due to the inherent inflexibility in the cubic polynomial modelling method. The fitted exponential coefficient for the $y = kV^n$ regression is 2.2494, this is close to the fitted exponential coefficient

for wind speeds between 10-20 knots from previous studies which fit this data with traditional regression approaches (Blomqvist 2016). This study only performs regression on data split into weather bins, so a direct comparison is not possible, although wind speeds between 10-20 knots are among the most common.

The expected ‘humps and hollows’ in the speed-power curve will not be modelled by either the cubic polynomial regression or the fitted exponential regression, as there is not sufficient flexibility in the methods. It is clear from Figure 3.17 that the neural network method has sufficient flexibility to model ‘humps and hollows’ in the curve. However, due to lack of understanding of the interactions between waves, the draft and trim of the vessel and the bulbous bow, there is no way to assess the accuracy of the ‘humps and hollows’ modelled.

Neither regression curve approximates the observed shaft power data above 18 knots of ship speed. It is unlikely that these types of vessel travelling at 20 knots require less, or equal power than when travelling at 18 knots. This ‘dropping off’ behaviour occurs at the point at which less data is available and the neural network predicts shaft power based on the data it has seen. This produces poor extrapolation because the underlying relationships are not modelled, therefore the network will not predict values higher than it has seen previously. The ship does not often operate at speeds above 17 knots, which stretches to beyond the designed maximum speed, and so it is less important for the analysis from a Naval Architecture perspective.

3.4.2 Sensitivity to Data Quantity

With a network of 3 layers and 300 neurons error in power prediction of less than 2% is common. The quantity of data required to achieve this error is investigated, Figure 3.18. Networks of size (3,300) are trained on datasets of increasing size, these are either sampled randomly or sequentially where consecutive samples are drawn, so the data used for the first point in Figure 3.18 is the first 288 datapoints in the database, equating to the first days’ worth of information. To imitate a real life scenario and to determine the number of days in operation required to use the prediction method accurately.

For randomly sampled datasets, the error in prediction when trained on only a days worth of data is 6%, this then decreases to be in line with errors from the whole dataset for 6 months worth of data. This low error shows that the network is reliable and accurate to predict shaft power for the majority of weather conditions and also shows that the network parameters are correct for the nature of the data it is being trained on. This also implies that a ship only needs to be at sea for 6 months to produce accurate results. However, this is misleading because the random sampling means the datasets will most likely include a larger spread of conditions than if the months were sampled as consecutive whole months.

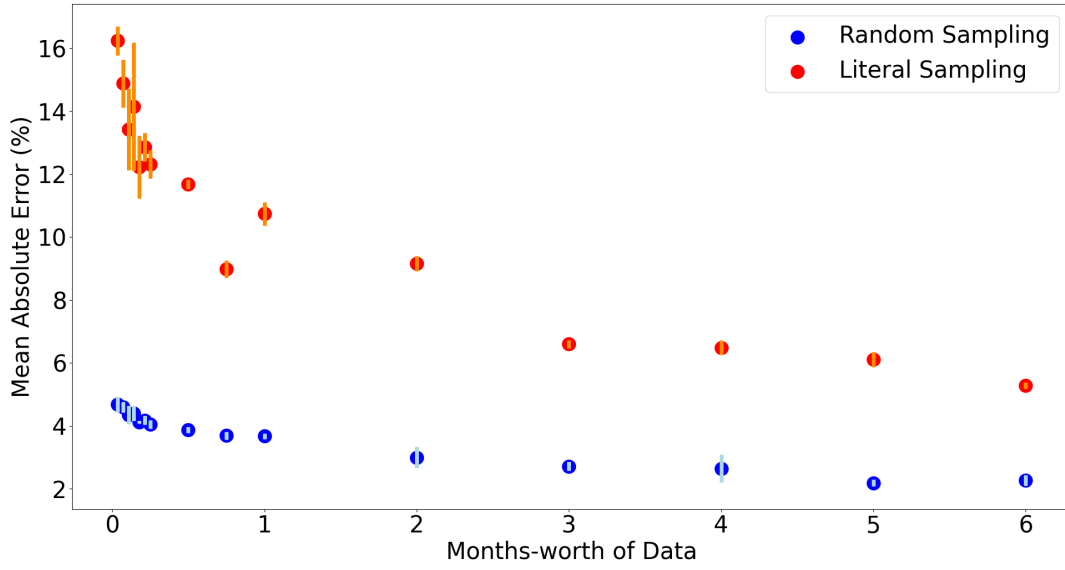


FIGURE 3.18: Power prediction mean Mean Absolute Relative Error, with error bars of maximum and minimum observed error, from networks of size (3,300) trained on randomly and literally sampled months of data. Up to 6 months of data are illustrated for readability of the figure, as randomly sampled dataset errors are in line with the full dataset, however the literally sampled errors continue to decrease steadily and reach errors in line with the full dataset at 20 months.

The sequential sampling errors start much higher, at over 16%. Networks trained on 12 months worth of literally sampled data produce errors double that from those trained on 6 months of randomly sampled data. This is because the network is exposed to a small range of conditions and therefore cannot predict the shaft power accurately for the conditions it has not yet been exposed to. The point at which the networks with literally sampled data converges, to be in line with errors from the full dataset, is based purely on when in the 27 recorded months of data the ship encountered sufficient range of conditions. This is a function of the routes the ships operate and the time of year they are in different locations.

3.5 Vessels Without Data

It is demonstrated that the power requirements can be predicted accurately for a vessel, using neural networks, and operational data in the previous section. However, the cost of gathering this data is high, in the region of £100,000s for each vessel. It is therefore impractical for every ship to be monitored purely for the determination of powering requirements. Utilising this data across fleets; making predictions for different vessels from the data collected from another, is important to reduce the cost.

Data fusion is increasingly used in machine learning to improve the accuracy or pertinence of results (Elmas and Sonmez 2011) (Melendez-Pastor et al. 2017) and the potential of blending data

from different instruments, time periods and applications is still under investigation. Operational ship data sets provide a particularly stochastic set of data, with continuous and discontinuous input distributions, as well as different operating profiles for each ship. The application of data fusion methods could allow power prediction for merchant vessels where there is no available data, such as those on charter agreements, and considerably reduce the cost for these methods.

This section provides insight into the potential of predicting powering for a vessel where no data exists by training networks on data from other vessels. Studies in similar areas, such as wind power prediction, show promising results (Tasnim et al. 2018). However, there are no known studies investigating this for vessel power prediction, which may be due to the comparative complexity of modelling vessel powering caused by the effect of second order variables, unmeasurable input variables such as piloting style, or the interaction of draught and trim. To perform this study a dataset containing more vessels are required, so this section uses data from 21 different Liquid Natural Gas carriers which does not include the 3 vessels from the previous analysis. Analysis is performed to understand the quantity of data that is required and the accuracy in prediction for vessels with different levels of similarity.

3.5.1 Fleet data

Of the 21 Liquid Natural Gas carriers, some are sister ships, so have identical hull forms and similar machinery, where others are the only example of their vessel type. To preserve the anonymity of the data, the vessel types are denoted by letters and for each vessel type each sister ship is denoted by a number. There are 8 different classes of vessel A-H, with quantities of sister ships for each vessel type ranging from 1 to 7. The size of each dataset ranges from nearly 2,500,000 observations to less than 60,000, Figure 3.19, all datasets have a frequency of 30 seconds per observation.

The same, minimal, trimming process employed in Section 3.2 is therefore also used on this dataset. This means that erroneous datapoints will still exist in the final datasets and to reduce the effect of these points, datasets larger than those used in the previous literature are employed in this study.

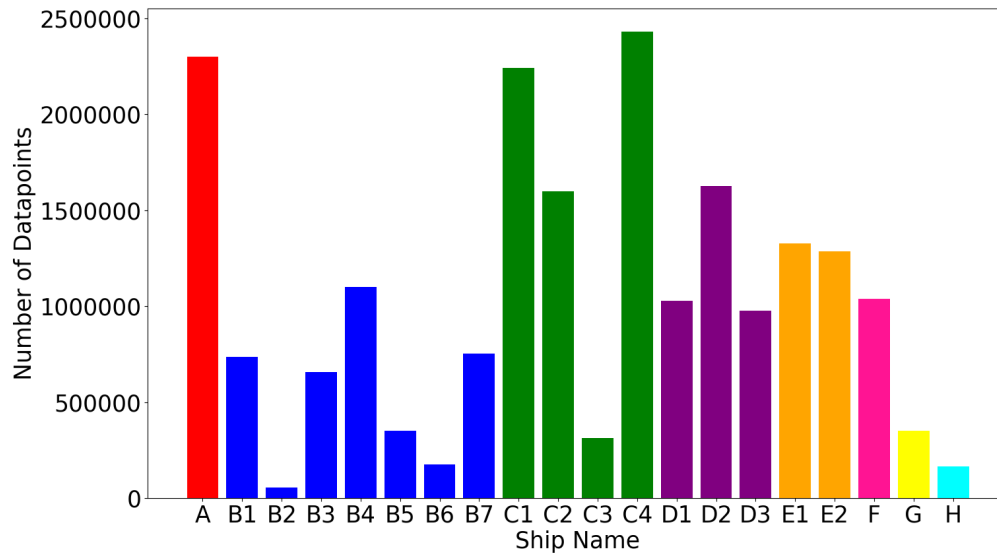


FIGURE 3.19: The size of each vessel’s dataset, with each class of vessel signified by a different colour.

To ensure the validity of the results and to accurately measure the change in error, the ship datasets should be of comparable size. Therefore, only datasets with over 1,000,000 datapoints are used. The size is chosen to ensure the testing and validation sets³ are over 150,000 datapoints, as statistical analysis showed that this size set approximates the distribution of whole dataset accurately for all variables used, and hence avoiding autocorrelation related bias in errors. Histograms illustrating the similarity in sample and full population distributions are available in Appendix A.

Out of the original 21 ships, 9 have more than the required 1,000,000 datapoints post trimming; with 2 sister ships in the D and E class, 3 sister ships in the C class and only 1 ship in each of the A and F classes. The size of the test set used for every vessel is 150,000, regardless of the overall size of the dataset. Although only 9 ships are used in this study, the methodology used is designed to be applicable to larger fleets, for example the data filtering applied is not tailored to the specific sensor errors found on each ship, instead only removing all points where vessel power is 0 or below.

The fleet of ships are all Liquefied Natural Gas carriers with build dates spanning 12 years from 2003 to 2015. Amongst the 9 ships there are two different types of propulsion system: steam and diesel-electric. All ships are of a similar size, with a 17m range in length and 3m range in beam. For these large merchant vessels 17m difference in size is not significant in relation to the overall size of the vessel, so powering relationships are expected to be similar. The C-class vessels have

³The randomly selected subset of 15%, of the dataset which is used for testing a networks power prediction abilities.

2 propellers, while all other vessels have one, larger, propeller. Propeller number and size are not expected to significantly affect powering but may change the routes operated by a vessel, as the single larger propeller will increase vessel draught and make some ports inaccessible.

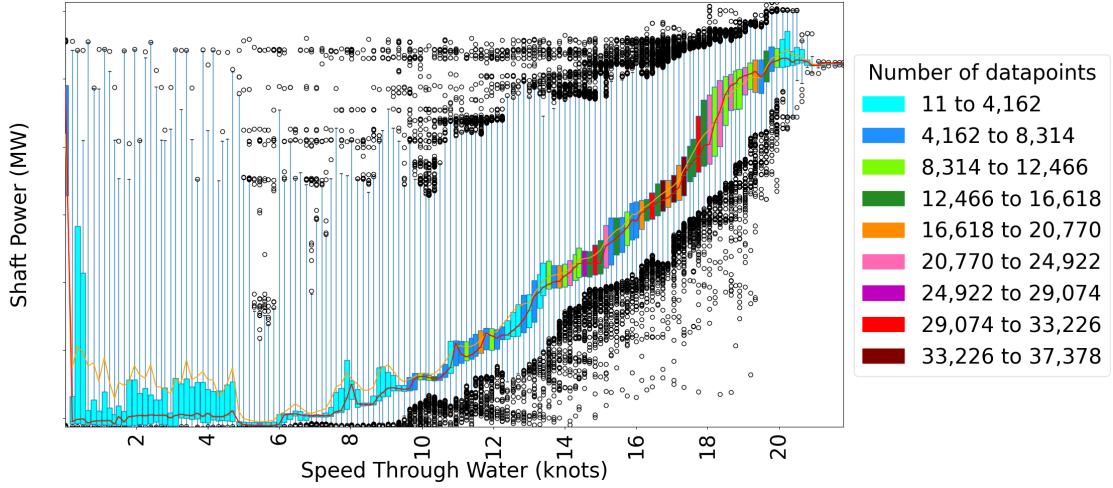


FIGURE 3.20: The distribution of the observed shaft powers for half knot bins of speed through the water for ship F. In the box and whisker plots the boxes contain 50% of the distribution and the whiskers extend to the datum which is at 1.5 times the interquartile range.

Each vessel has a similar operational profile to the ships analysed in Section 3.2, as all 12 are the same type of vessel with little variation. The speed-power relationship shows similar density of data around an approximately cubic with noise spanning nearly the entire range of observed shaft powers, Figure 3.20, as observed for vessels used in Section 3.2. Similar areas of sparse data occur in the 9 new ships discussed here as discussed in the previous Section. An example of this is the regions of data with speed lower than 10 knots and higher than 20 knots for ship F, Figure 3.20, as each of the lightest coloured boxplots holds less than 0.3% of the dataset.

To compare the power profiles of the vessels across the fleet, the mean power observed in each partition of the speed domain is plotted for all of the vessels, Figure 3.21a. No noise or secondary relationships are captured by these speed-power curves, however a difference in propulsion relationship is clear for ship A. The required power for ship A is around 20% of the maximum power higher than for all of the other vessels, for all speeds over 7 knots. When the standard deviation of the distribution of power at each interval is shown, Figure 3.21b, it is clear that there is no significant overlap in distributions between the power of ship A and the power for all other vessels, especially between the ranges of 10-20 knots. This vessel is the only steam powered ship, as well as the oldest vessel in the fleet by 7 years, so a difference in propulsion characteristics is expected. This difference in performance may cause problems if attempting to predict powering of ship A by training on vessel data from the other ships.

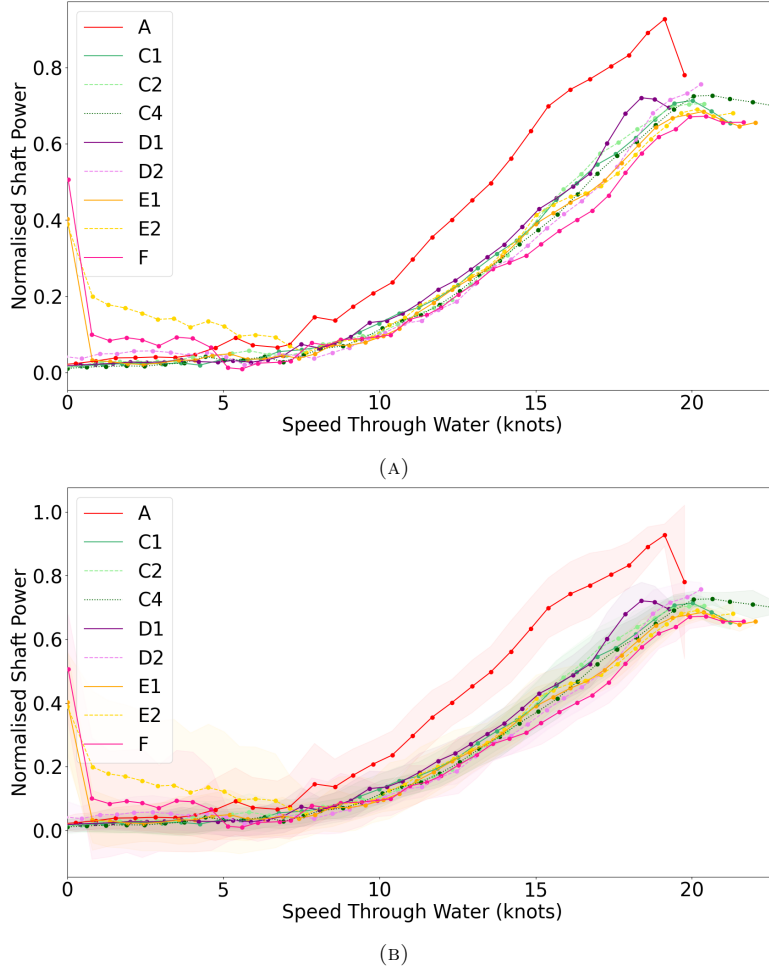


FIGURE 3.21: Comparison of the average observed power at half knot intervals of speed through the water for all vessels in the fleet (A) and with \pm one standard deviation of the distribution of shaft powers in each interval (B).

All ships, apart from ship D2, display the ‘dropping off’ of speed discussed in Section 3.2 due to low quantities of data in higher regions of the speed domain, Figure 3.21a. There is an increase in average shaft power for near zero speeds for ships E1, E2 and F, Figure 3.21a. This is unexpected, but an explanation for these phenomena is the ‘law of small numbers’; that the data distribution produces misleading statistics as these regions have low quantities of data they are skewed by small quantities of sensor error. This theory is supported by the spread of power for each ship, as all ships with increased power near 0 speed also show larger standard deviations in power at low speeds, Figure 3.21b. Since no trimming is performed to remove data from times when the vessel is in port or manoeuvring, this is suggested as a possible explanation.

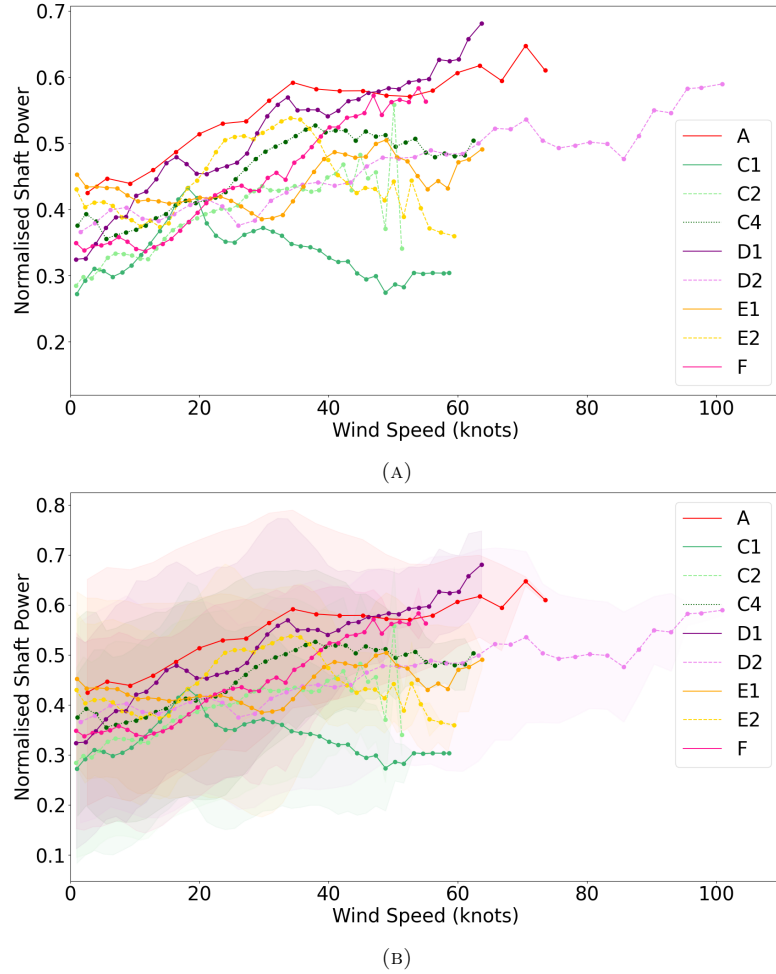


FIGURE 3.22: Comparison of the average observed power at half knot intervals of wind speed for all vessels in the fleet (A) and with \pm one standard deviation of the distribution of shaft powers in each interval (B).

To compare vessel datasets the power-variable curves are analysed and the power to wind direction curves show the same trend across the fleet for the wind direction domain from 0° to 360° . This means wind direction relationships learnt for one vessel should transfer to another vessel well. However, the wind speed curves show a less even distribution of data across the range of observed wind speeds. Ship D2 is the only vessel to experience the highest speeds of 60-100mph, Figure 3.22a, which is extreme, equivalent to Beaufort 12 or ‘hurricane force’. It is suggested this may explain the lack of ‘dropping off’ at high vessel speeds shown only by ship D2 in Figure 3.21a. Although only 0.7% of the dataset records wind speed values above 60mph, when plotted temporally, the set of datapoints containing these high wind readings create a smooth curve which suggests that these data is not anomalous. This 60+ knot area in the wind speed domain is sparsely populated, both in terms of the total number of datapoints and in the variety of vessels with datapoints populating it, making it difficult to predict behaviour in this region as the distributions are not representative of the behaviour.

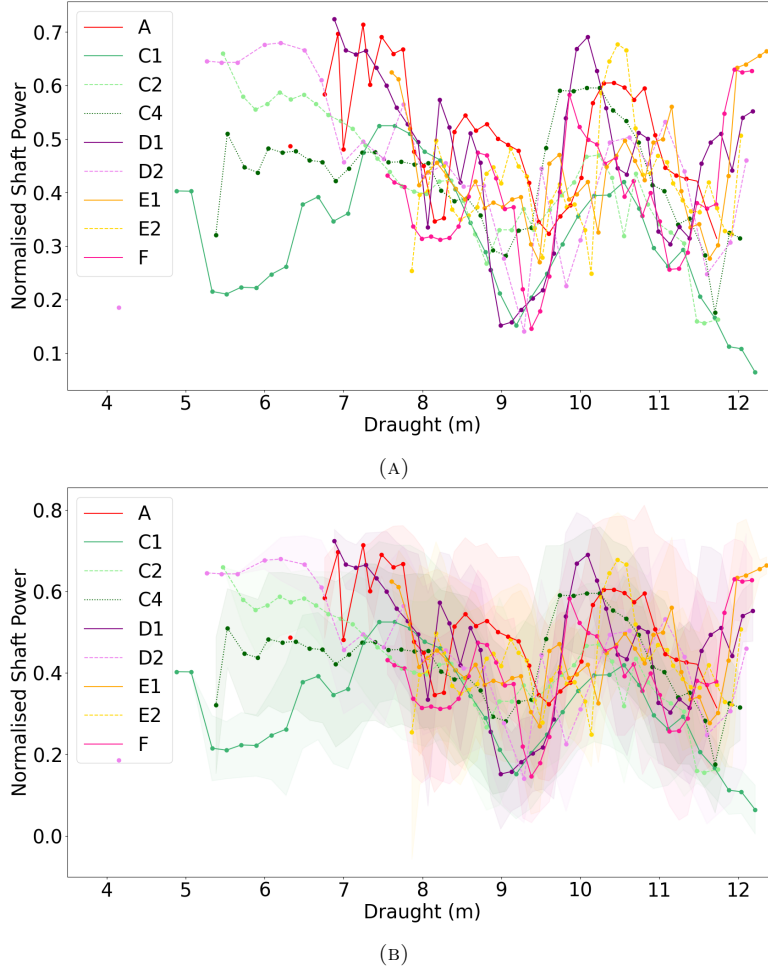


FIGURE 3.23: Comparison of the average observed power at 20cm intervals of draught for all vessels in the fleet (A) and with \pm one standard deviation of the distribution of shaft powers in each interval (B).

A similar area of sparse data can be observed in the draught variable domain, Figure 3.23b. The power-draught curves show that only 4 vessels; D2, C1, C2 and C4, operate at draughts below 7m. The quantity of datapoints below 7m of draught is 0.21% of the combined 4 vessel's data. The sparseness of this region means the data in it may not have representative distributions, which may be the cause of the separation between all four lines below 7m of draught, Figure 3.23b. The power-trim curve is also analysed, but due to the coupling of draught and trim, does not provide any further insight.

3.5.2 Validate Neural Network Parameters

First, it is verified that the neural network parameters used in Section 3.3 are appropriate for the 9 new vessels. For each ship, combinations of 1-4 hidden layers and 1-500 neurons in each layer are trained and tested on their operational dataset⁴. The only difference for each of the 10 network

⁴For all future results in this Section 10 repetitions are performed to increase validity.

runs is the randomly sampled training, testing and validation sets, the random initialiser and any stochastic elements of the optimiser. This acts as a benchmarking of the method and datasets against results in the literature and allows an initial comparison of the prediction accuracies of the ships. Figures detailing the small parametric study are available in Appendix A confirming that the network parameters are appropriate and networks of size (3,300) are used throughout this section⁵.

For most of the ships, the distribution of the mean prediction errors from the different vessels ranges between 1.78-2.13%. Ship D1 can be predicted the most accurately, and exhibits a lower average error of 1.35%, and ship A is the most difficult to predict, with all of the errors above 3% (Figure 3.24). These accuracies are similar to those in the literature and results for the previous dataset in Section 3.3. Powering is predicted to within a 5% error (Pedersen and Larsen 2009) with a best result of 1.5% from (Petersen et al. 2012), where the wave height is used as an additional input variable which has been shown to give an increase in accuracy of 0.5% (Parkes et al. 2019). Each ship shows consistent predictions, with low standard deviations, around 0.25%. Networks with the parameters in Table 3.1 produce errors inline with the power prediction literature for every ship, therefore this size of network is used in the following sections.

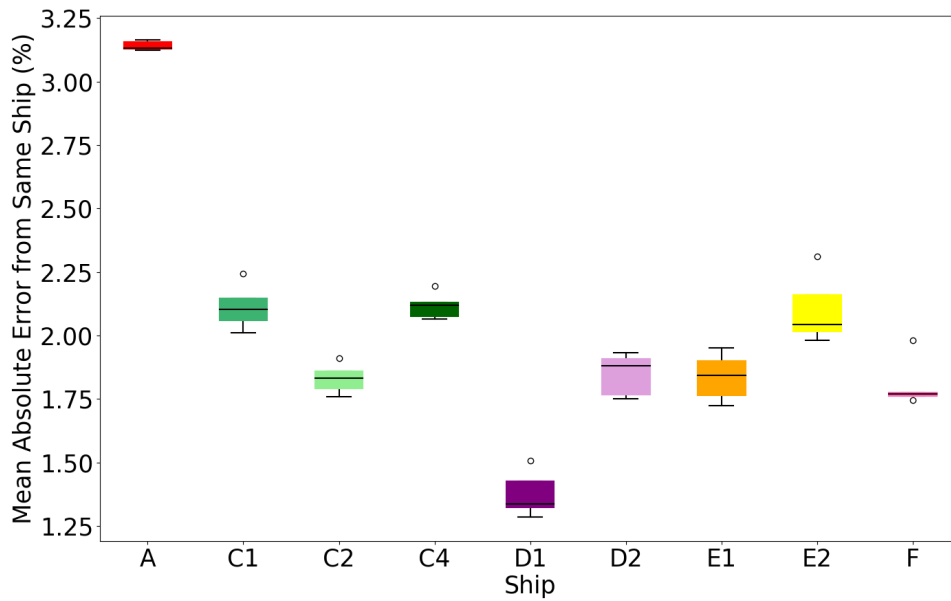


FIGURE 3.24: The distribution of mean absolute relative error from multiple networks of size (3, 300) for individual vessels, showing consistent predictions for individual ships and a maximum difference of 1% between the mean error for different ships. In the box and whisker plots the boxes contain 50% of the distribution and the whiskers contain 90% of the distribution and the circles show outliers.

⁵All simulations from this Section and the rest of the thesis are performed on Iridis Compute Cluster at the University of Southampton.

Within these predictions there is no relationship between error value and ship class, as error values within ship classes vary just as much as between ship classes. This suggests that this variation between ships is likely to be due to factors not relating to hull form or ship parameters. These factors may be a specific sensor error, the vessel conditions experienced, or differences in piloting and operation. The next section documents the use of a network trained on a fusion of data from all of the ships in the fleet apart from one, which the network is tested on.

3.5.3 Prediction for Ship Without Data

To emulate a more realistic situation, where data is available from some but not all vessels in a fleet; this section uses a form of k-fold cross validation where data from all of the vessels except for one, and uses this network to predict powering for the unused vessel. This increases the size of dataset used to train the networks from around 700,000 to around 5,600,000.

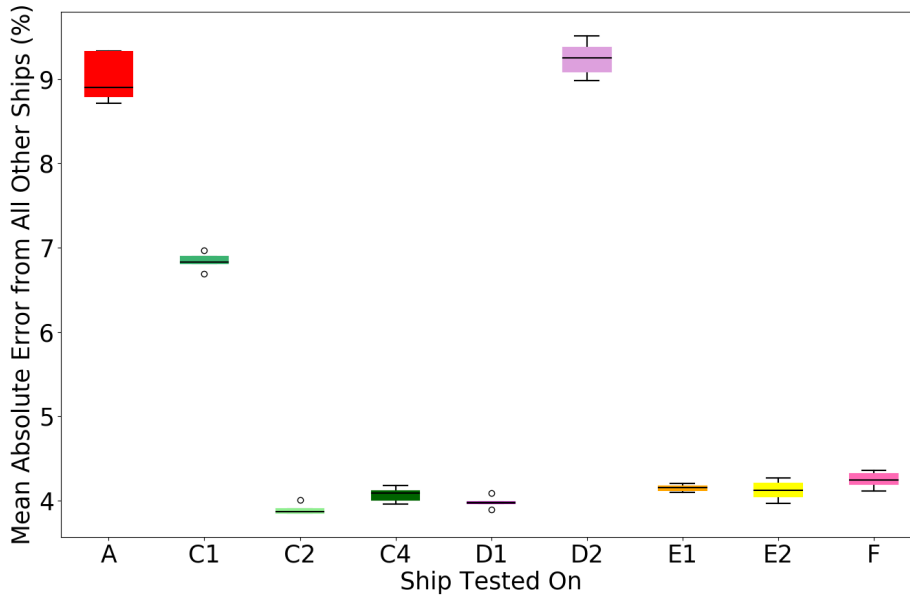


FIGURE 3.25: The distribution of mean absolute relative errors from multiple networks trained on all of the vessels apart from the ship tested on.

Errors in the range $(4 \pm 0.25)\%$ are observed for 6 of the 9 vessels tested, Figure 3.25. The three vessels which have a higher error are ships C1 with $(6.83 \pm 0.14)\%$ error in prediction and ships A and D2 with $(8.89 \pm 0.44)\%$ error and $(9.26 \pm 0.27)\%$. Ships C1 and D2 have sister ships with errors in line with $(4 \pm 0.25)\%$, which shows that the dimensions of the vessel are not causing the high errors for these ships. Therefore, including parameters like vessel length and hull form as an input would not improve prediction accuracies for this fleet.

It is suggested that the difference in power-speed curves, in Section 3.5.1, explains why the errors in prediction for ship A are high, as the speed-power relationship in the other 8 datasets are 20% lower than ship A, Figure 3.21a. This is due to the difference in propulsion systems and age of vessel as ship A is the only steam powered vessel in the fleet.

Ship C1 shows an error 2% lower than the other anomalous ships, A and D2. Ship C1 is one of the four ships to operate at low draughts, Figure 3.23b. All of the power-draught relationships in this region show different relationships, due to the low quantity of data. However, ship C1 is the only vessel to show an increasing trend in this region, with an increase in power for an increase in draught. When the region of draught below 7m is removed from all of the datasets, the powering for ship C1 is predicted with a 4.6% error from a network trained on all other vessels. The difference in low draught behaviour is likely to be the reason for the difference in error, as when appropriate input variable ranges are selected the error decreases to within 0.5% of all other ships. This illustrates that the networks have not modelled the ground truth input-output relationships within the dataset, as they do not perform in areas of sparse data.

Ship D2 has errors over double that of its sister ship D1, Figure 3.25. This suggests that an area of the input variable space for this vessel is not covered by the training dataset, so the network is forced to extrapolate. The operating conditions experienced by the vessel differs from the rest as it is the only to experience 60+knot wind speeds, Figure 3.22a. This explains why the errors for ship D2 are high; no other ship experiences the same extreme conditions so a network trained on all other ships which does not model the relationship between wind speed and shaft power will not be able to predict in this region. It is confirmed that when the area of extreme weather, 60+knots, and unusually low draughts were removed from the dataset, the errors for ship D2 reduce to 3.6%, which is in line with the other vessels.

This illustrates that the same situation occurs as in Section 3.3. Where low interpolation errors are reported, predictions within regions of dense data are accurate, but these error measures do not capture the fundamental relationship between wind speed and powering requirements, as the network fails to extrapolate the increasing relationship between wind speed and power which is shown in the majority of vessels, Figure 3.22a.

3.6 Discussion

A common limitation of neural networks is poor extrapolation abilities. The results of the larger networks used in Section 3.3 produce poor extrapolation despite producing higher accuracy, even though they approximate the traditional regression curves for the majority of ship speeds, which show much more logical extrapolation. These extrapolation errors suggest that the networks are

not modelling the underlying relationships in the data, even those with the strongest effects, and predicted relationships are being skewed by small quantities of extreme data.

Section 3.3 investigates repeatability of the networks with saliency analysis and further explores the idea of networks prediction powers, demonstrating that predictions in areas with sparse data are worse. This principle is illustrated in the low consistency of predictions in these regions compared to other areas with a high density of data. The network is acting as a ‘black-box’ and accurately predicting in areas with dense data but not in areas with sparse data as it has not approximated the relationships between variables. Producing low error in areas of high density data does not require the network to be able to approximate the relationships between variables, as the data tunes the regions predictions, but when there is less data, the networks struggle to produce realistic results.

This poor prediction in areas of sparse data is noted in Section 3.5, where networks are trained on data from a fleet of vessels and predict powering for a different ship. A network trained on data from all vessels apart from ship D2, cannot extrapolate a linear increasing curve for increasing wind speed beyond values seen in the training data. The Mean Absolute Relative Error for this network when predicting within regions of previously ‘experienced’ wind speed, is 3.6%. This increases to an average of 9.3% across the entire range of wind speeds when including a region for prediction which has not been seen by the network before. The networks are not modelling relationships between input variables and the output, even the simplest one, regardless of the low Mean Absolute Relative Errors observed.

It is suggested that previous literature does not encounter the problem of predicting in regions of sparse data as it is not investigated. No other study presents the isolated input-output relationships learnt by the networks, nor discusses predictions in sparse or extrapolated regions. It is possible that the extensive data trimming that is often performed, reduces the variation in a dataset and removing these areas of sparse data, even if prediction in these areas is of interest. The datasets used in this chapter have more variance than the data used in other neural network ship powering applications, Table 3.4, which supports this hypothesis.

TABLE 3.4: The coefficients of variation for ship speed and powering and quantity of data used in this chapter compared to other datasets used for neural network applications

Study	Coefficient of Variation	
	Ship Speed	Power
Section 3.4 (Parkes et al. 2018)	0.3	0.59
Section 3.5	0.16-0.36	0.41-0.58
(Pedersen and Larsen 2009)	0.006	0.001
(Petersen et al. 2012)	0.17	0.22
(Bal Beşikçi et al. 2016)		0.26
(Le et al. 2020)	0.199	

Despite the poor extrapolation of networks trained in this chapter, most produce Mean Absolute Relative Errors in line with, or lower than, results in the literature. Visualising the predicted isolated input-output relationships suggests that arbitrary curves can be approximated and still produce low error. It is suggested that to be able to produce a network with the ability to model relationships and hence have better extrapolation powers, error measures which measure how well the networks approximate the underlying relationships in the data rather than the accuracy of predictions for single datapoints will be essential. No such error measure exists and all papers reviewed in the literature review use error measures based on point to point accuracy between predicted values and targets, or the correlation of predictions for model evaluation. Once a error measure to assess this exists, it can be used to determine which types of network approximate the physical relationships in a system and can then be used as a tool for other problems.

3.7 Summary

This chapter details the methodology of applying a regression neural network to a ship powering application from data analysis and refinement to benchmarking of results with a focus on the input-output relationships modelled by the networks. This application is selected as the physical relationship between input and output variables are not fully understood and there is a requirement for regression methods to accurately approximate the ground truth input-output relationships.

The input-output relationships modelled by neural networks predicting vessel powering have not been analysed before in the literature. This section shows that, although producing error values which approximate the measurement error of the dataset, and which are similar to error values

from other applications to predict vessel powering, the networks do not model the underlying relationships in the data. They fail to extrapolate accurately, and produce low repeatability in regions of less dense data. It is concluded that the reason for the poor extrapolation is that the accuracy measures used to assess network performance do not measure how well the network approximates the ground truth functions in the dataset, and to improve a networks abilities to model this new error measures are required.

Power prediction for a ship which does not gather operational data, from a network trained on a fleet of different vessels is performed. This has not been studied before, and the accuracy of prediction for an ‘unseen’ ship is around 4%, low enough for use on energy saving devices. The fleet predictions also show that extrapolated predictions have poor accuracy, and attribute this to the networks not approximating the ground truth relationships of the dataset.

Chapter 4

New ‘Fit to Median’ Measure

As demonstrated in Chapter 3, neural networks reporting low Mean Absolute Relative Error, or other Minkowski-r error metrics, do not necessarily model the isolated input-output relationships accurately. This chapter therefore derives a new error measure to assess how accurately a neural network approximates the ground truth of a dataset. This error measure will be able to identify if a trained network approximates the conditional averages of a dataset, hence this error measure will not be useful when the relationships between inputs and outputs are not functions. To validate the new error measures abilities, artificial datasets are required, to ensure the ground truth relationships are known. The novel error measures are then applied to the ship power prediction dataset and produce networks with improved predictions.

4.1 Regression Metrics

Common regression measures can be classified into two main types: point based and correlation based, Figure 4.1. Point-based measures calculate a distance measure between two points—the predicted value, and most often the target value—aggregated over the whole set.

Correlation based measures assess the relationship between the set of test target values and the set of predictions of these values irrespective of which prediction is for which target. They include the R^2 or Pearson Correlation Coefficient between model outputs and target values. Correlation based measures can be misleading for non-standard distributions of data. Additionally, interpreting model performance from correlation measures is more difficult than interpreting point-based error measures (Taylor 1990), as they provide limited intuition about model behaviour for a single prediction or use in ‘real-life’ scenarios.

Nearly all common regression error measures are point based and can be defined as in expression 4.1 (Botchkarev 2018),

$$\mathbb{G}_{i=1,\dots,n}(\mathbb{N}[\mathbb{D}(a_i, b_i)]), \quad (4.1)$$

where \mathbb{G} is some aggregation method for a test set of size n , \mathbb{N} is some normalisation method, and \mathbb{D} is a distance measure between the two points a and b .

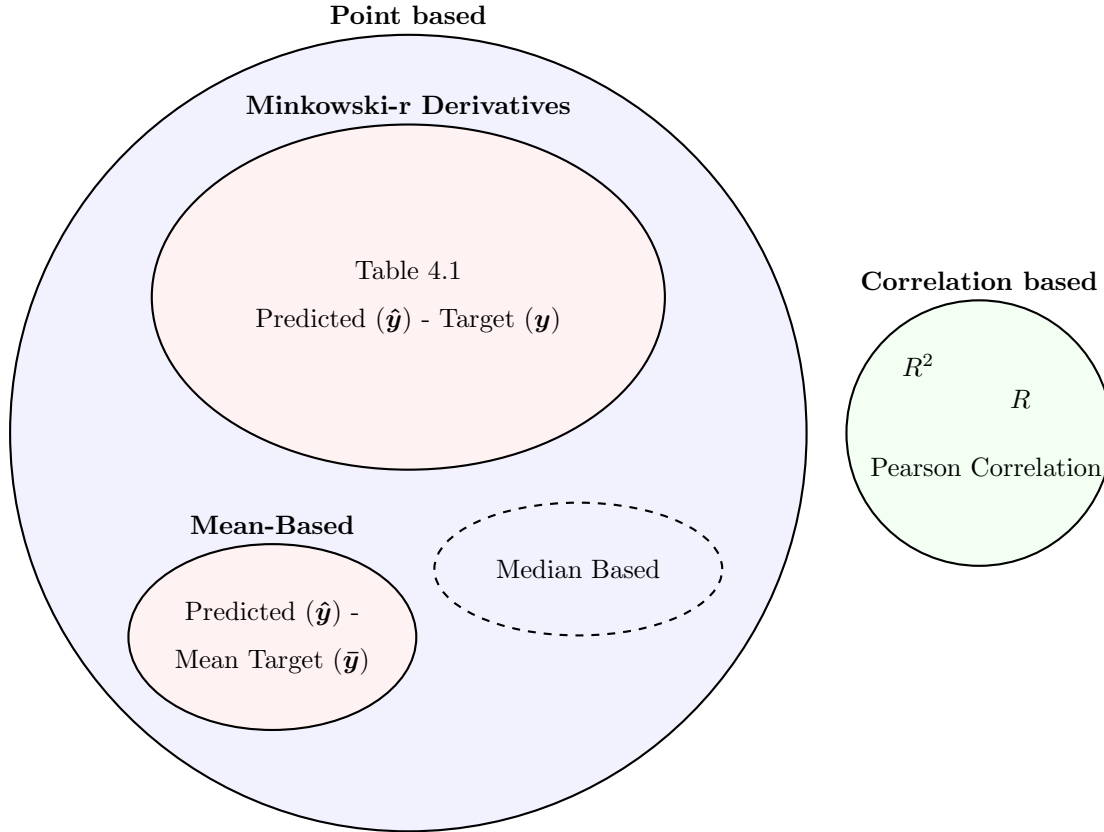


FIGURE 4.1: Euler Diagram of Regression Error Measures.

4.1.1 Minkowski-r Derivatives

The majority of these measures calculate the distance between the predicted target variable(s) \hat{y} and the actual target value from the measured dataset y . A selection of error measures of this form are summarised in Table 4.1, illustrating that the distance between the predicted and actual target values is integral to the common regression performance measures. Table 4.1 does not present an exhaustive list of error measures, instead it introduces a taxonomy of commonly used regression error measures. Other common error measures such as Kullback Leibler (Kullback and Leibler 1951), fit the taxonomy with a distance measure of the log quotient and normalisation from the target value but are not included in the table for compactness.

Similarly the Mahalanobis distance (Mahalanobis 1936) reduces to a scaled Euclidean distance, for diagonal covariance matrices, so fits the same taxonomy.

TABLE 4.1: Point based error measures, derivatives of Minkowski-r Metrics

D Distance Measure	N Normalisation				G Aggregation
	None	Target Value	Range of Target Values	Sum of Target and Predicted	
Euclidean Distance $y - \hat{y}$	Mean Error	Mean Percentage Error, Mean Normalised Bias		Fractional Bias	Mean
					Median
	Manhattan Distance				Sum
Absolute Euclidean Distance $ y - \hat{y} $	Mean Absolute Error	Mean Absolute Percentage Error	Mean Absolute Relative Error	Symmetric Mean Absolute Percentage Error, Fractional Absolute Error	Mean
	Median Absolute Error	Median Absolute Percentage Error	Median Absolute Relative Error	Symmetric Median Absolute Percentage Error	Median
	Sum of Absolute Differences		Relative Absolute Error	Canberra Metric	Sum
Squared Euclidean Distance $(y - \hat{y})^2$	Mean Squared Error, Root Mean Squared Error	Mean Squared Percentage Error, Root Mean Squared Percentage Error			Mean
		Median Squared Percentage Error			Median
	Sum of Squared Errors	Neyman Chi- Squared Distance	Relative Squared Error	Square Distance Divergence	Sum

Derivatives of Minkowski-r metrics are produced by varying the normalisation or aggregation method. These measures do not meet the original description of Minkowski-r measures, but rely on the same basic principle of the Euclidean distance between predicted and target value (De Gooijer and Hyndman 2006). For example, the Mean Squared Error is a standard Minkowski-r measure but Root Mean Squared Error is not, equations 4.2 and 4.3. Assuming target distributions can be approximated by normal distributions, Mean Absolute Relative Error and Mean Absolute Percentage Error should be comparable, Mean Absolute Relative Error normalises error values relative to the range of the variable whereas Mean Absolute Percentage Error normalises error values relative to the value.

Mean Absolute Percentage Error is produced using a normalisation with the target variable value, equation 4.6. This provides intuition about how poor a prediction is relative to the target value, reducing the effect of bias towards larger target values. Normalising by the range of target values creates a ‘relative’ error, Table 4.1, where errors are relative to the range of the data. These two measures will approach each other as the distribution of the target variables approach a normal distribution;

$$\text{Mean Squared Error} = \frac{1}{n} \sum_i (|y_i - \hat{y}_i|)^2, \quad (4.2)$$

$$\text{Root Mean Squared Error} = \sqrt{\frac{1}{n} \sum_i (|y_i - \hat{y}_i|)^2}, \quad (4.3)$$

$$\text{Mean Absolute Percentage Error} = \frac{100}{n} \sum_i \frac{|y_i - \hat{y}_i|}{y_i}. \quad (4.4)$$

New error measures are regularly defined and fall into two categories: introducing domain knowledge (Grigsby et al. 2018) (Bratu 2013) (Kim and Kim 2016), or combining existing measures (Kyriakidis et al. 2015). The former of these categories is synonymous with physics-based machine learning methods which alter the loss function to direct a method towards known relationships. This approach produces improved extrapolation prediction and more reliable results, but is only possible if sufficient domain knowledge exists, as discussed in Section 2.7. The latter fits into the framework discussed here, and Table 4.1 could be expanded to include a larger range of more obscure measures of this form. As these measures are based off the same set of point-based principles, they do not produce any improved performance above incremental dataset or application specific improvements.

The difference in scaling produced by different aggregations and normalisations may mean one error measure is preferable over the other for certain applications as the error surfaces produced by these error measures will differ slightly in shape (Cha 2007). This can change the performance

of a regression tool used to minimise these errors. However, all Minkowski-r derivatives will have minimum and maximum values from the same arguments.

4.1.2 Mean-Based Measures

To ensure predictions are close to conditional averages, ‘Mean-Based’ error measures have been developed. For intermittent demand forecasting, where time series with occasional extreme spikes are modelled, the mean target value \bar{y} is used in place of the actual target value y producing more reliable forecast demands (Prestwich et al. 2014). In this study Mean Absolute, Mean Absolute Percentage, Median Absolute, and Mean Squared Errors are adapted to be ‘Mean-Based’ and the ability to adapt any standard point based error measure to be ‘Mean-Based’ is stressed. Improved behaviour results from single exponential smoothers using ‘Mean-Based’ error measures compared to the Mean Absolute Relative Error which is the most common error measure for intermittent demand forecasting (Hyndman and Koehler 2006).

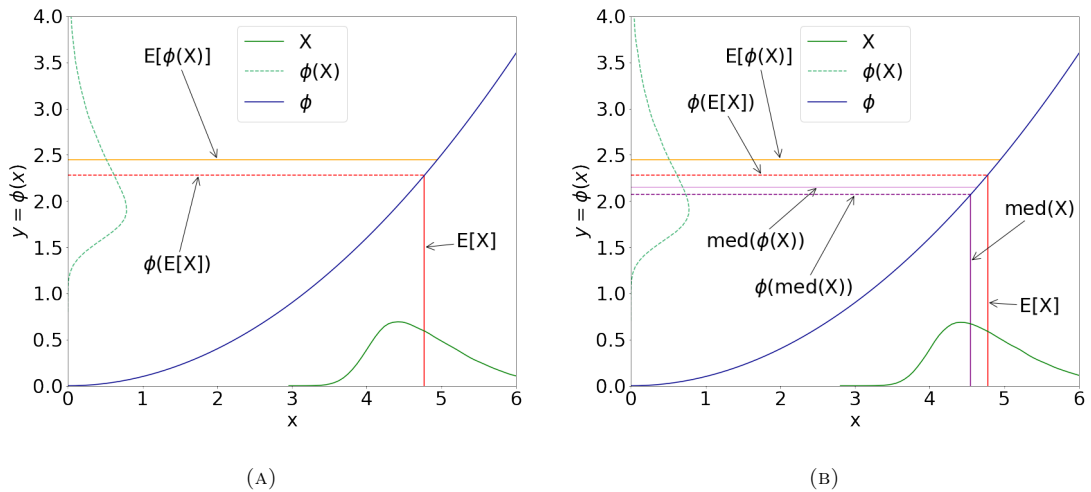


FIGURE 4.2: Illustrations of Jensen's inequality $\phi(E[X]) \leq E[\phi(X)]$ for convex function ϕ where A) the transformed mean of a distribution $\phi(E[X])$ is less than the mean of the transformed distribution $E[\phi(X)]$ (Jensen 1906) and B) where the gap caused by this inequality is smaller for the median of the distributions than for the mean. (Merkle 2005).

For applications modelling physical causality in a system a conditional average is likely to approximate the ground truth of the dataset, although depending on the characteristics of input-output relationship and any noise the choice of conditional average may be important. By synthesising results from (Jensen 1906) and (MacGillivray 1981) we have that, for datasets where the input-output function has higher curvature and larger standard deviations of noise, the conditional median will approximate the ground truth closer than the conditional mean compared to datasets with a lower curvature and smaller standard deviation of noise, where the mean and median will tend to the same value as the skew of the conditional distributions decreases.

However, the conditional average does not necessarily approximate the ground truth for all datasets. For example if the relationship between inputs and outputs are a relation and not a function, where a function is a relationship where all input values are mapped to at most one output value. In the case the relationship between inputs and outputs are not a function there are several valid output values for an input value. This means the conditional average of the dataset can hold no useful information about the data at all, as the average of several solutions is not necessarily itself a solution.

4.2 Derivation of the New Error Measure

To derive an error measure which quantifies how accurately a regression method models the ground truth in a dataset, an approximation of the isolated input-output relationships is required. Therefore, the proposed new error measure calculates a proxy for the relationship between each input variable and the output, the conditional average independent of other input variables. It measures the distance between the networks approximation of these relationships and the proxy curves.

First, these relationships are formalised for any dataset; for clarity the principles of the new error measure are explained using a two dimensional example, where input x is used to predict output y , and this is expanded to include scenarios with multi-dimensional inputs for the formal statement of the error measure. In this initial reduced set-up, let ϕ be the relationship between input variable x and output variable y , such that $y = \phi(x)$. For simplicity, the only curves ϕ discussed are either convex or concave across their entire domain, although it is suggested that the arguments extend to a piecewise curve.

From this dataset we have the distributions X and $Y(= \phi(X))$ shown on the axis of Figure 4.3a and $E[X]$ and $E[Y](= E[\phi(X)])$ which are defined as the expectation, or mean, of the distributions. Assuming these distributions are sufficiently continuous, a naive proxy for ϕ can be derived by estimating the conditional averages of the dataset. To produce this the X domain must first be divided into n equally spaced bins (X_1, X_2, \dots, X_n) , Figure 4.3b. Then $Y(= \phi(X))$ is partitioned based on the datapoints’ corresponding x values, producing (Y_1, Y_2, \dots, Y_n) . Y_i is therefore the set of all the observed output values for the input values in X_i . Assuming a sufficiently large set of partitions n , the mean of Y_i and the midpoints of X_i create an initial proxy for the conditional average: $(\text{mid}(X_i), E[Y_i])$ for $i = 1, 2, \dots, n$.

However, from Jensen’s inequality we have that $\phi(E[X]) \leq E[\phi(X)]$ for convex¹ ϕ (Jensen 1906). This is illustrated in Figure 4.2a where the function value at the average of the input variable distribution ($\phi(E[X])$) is less than the average of the distribution of the function values

¹And $\phi(E[X]) \geq E[\phi(X)]$ for concave ϕ .

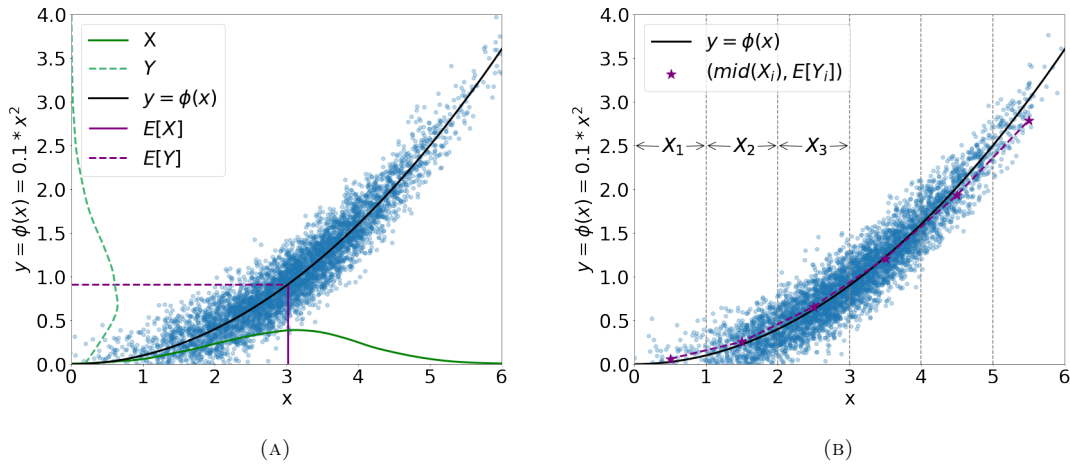


FIGURE 4.3: Illustration of (A) expectation of X and Y distribution, (B) partitioning X distribution to create proxy curve.

of the inputs ($E[\phi(X)]$). This means that for any ϕ which is not a straight line, a gap between $\phi(E[X])$ and $E[\phi(X)]$ will exist and hence the mean of Y_i will not approximate $\phi(X_i)$ for these applications. Formal bounds on this inequality exist (Gao et al. 2018), but cannot be applied without prior knowledge of ϕ .

We know that $\text{med}(\phi(X)) \leq E[\phi(X)]$ for unimodal right-skewed distributions from basic statistics (MacGillivray 1981), so the median of each Y_i (denoted $\text{med}(\phi(Y_i))$) could produce better estimates of ϕ under certain constraints. If ϕ is non-linear then X_i and Y_i will have a skew and, as continuity is already assumed, these subsets of the distributions will be unimodal. Jensen's inequality also holds for medians; $\phi(\text{med}(X)) \leq \text{med}(\phi(X))$ for convex ϕ (Merkle 2005), Figure 4.2b. So although there will also be a gap between $\text{med}(\phi(X))$ and the true curve ϕ , $\text{med}(Y_i)$ will produce a closer approximation than $E[Y_i]$. The proposed measure is therefore the ‘Mean Fit to Median’ which is the mean of the distance between the proxy points for the conditional median and the isolated input-output relationships predicted by a machine learning regression method.

Here the notation is expanded to include multiple input variables, so that there are m independent input variables x^j such that $y = \phi(x^1, x^2, \dots, x^m)$. The distribution of each input variable is partitioned as above, formalised this produces

$$\text{proxy curves} = \{\text{med}(Y_i^j) | i = 1, 2, \dots, n, j = 1, 2, \dots, m\},$$

where Y_i^j is the set of y values corresponding to the set X_i^j .

The distance between these proxy curves and the input-output relationships learnt during training is calculated to produce the Mean Fit to Median Error value. The input-output relationships

learnt by a trained regression method, are produced by probing a trained method to predict output values for each input variable in turn by using inputs, cycling from the minimum to maximum observed values of each input, with all other inputs at the median value. Explicitly, if P denotes the representations learnt from a model such that $P(x^1, x^2, \dots, x^m)$ is the predicted output value for inputs x^1, x^2, \dots, x^m . Then for input variable j , the predicted input-output relationship is approximated by the set

$$\text{learnt input-output curves} = \{p_{X_i}^j | i = 1, 2, \dots, n, j = 1, 2, \dots, m\},$$

where $p_{X_i}^j = P(\text{mid}(X_i^j), \text{med}(X^k) \text{ for } k \neq j)$ with $\text{mid}(X)$ denoting the midpoint of the set X . This allows the Mean Fit to Median Error measure to be written as

$$\text{Mean Fit to Median} = \frac{1}{m} \sum_{j=1}^m \frac{1}{n} \sum_{i=1}^n \left| \text{med}(Y_i^j) - p_{X_i}^j \right|. \quad (4.5)$$

To summarise, the Mean Fit to Median is the average Euclidean distance from the learnt input-output relationships to the median output value conditioned on the input for each input in turn. It is only possible to calculate this error measure after normal training has terminated, as the input-output relationships learnt by the method are required to perform the calculation. This means that currently it is only possible to use the ‘Fit to Median’ measure as a post-training evaluation method. Approaches to use the measure to train networks are discussed in Section 5. However, it can be used to identify the method with the best ground truth approximation from a set of trained methods. Due to the extension of Jensen’s inequality to medians (Merkle 2005) the Fit to Median does not generalise to solve inverse problems, as illustrated in Figure 2.3. The conditional median is chosen as the ‘proxy’ to measure from, as from analysis in Section 4.1 the median will always have the least shift away from the ground truth relationship than the mean in situations with high relationship curvature and large variance.

4.3 Artificial Datasets

For many real regression applications a full understanding of the system and its uncertainties is not available. This means that benchmarking to what extent a trained method replicates the true relationships between inputs and outputs is not possible. To allow this to be accurately measured artificial datasets are used, with a range of fully defined variable relationships and differing levels of uncertainty, with the aim of approximating the complexity and characteristics of common regression problems. The data is also generated with the intention of violating assumptions (ii)-(iv), discussed in Section 2.2.

TABLE 4.2: Varieties of Dataset Generated for Trialling Error Measure, illustrated in Figure 4.4.

		f		
		\mathbb{P}_{0-1}	\mathbb{P}_{2-3}	\mathbb{P}_{4-5}
σ	0.1	a	b	c
	1	d	e	f
	10	g	h	i
	100	j	k	l

This data generation process is formalised as follows: the input variables are generated such that $\mathbf{X} \sim N_6(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, with $\boldsymbol{\mu} \in \{(\mu_0, \dots, \mu_5) | \mu \in \mathbb{R}\}$ and for initial benchmarking $\boldsymbol{\Sigma}$, the covariance matrix, is diagonal with a view to investigate scenarios where input variables are not independent in a further study. In practice μ_i is randomly generated as $\mu_i \sim \text{unif}(0, 10)$ from NumPy random number modules (Harris et al. 2020) in Python and the number of observations generated is 1,000,000. To generate one output y the following identity is used;

$$y = \phi(x_0, x_1, \dots, x_5) = \phi(\mathbf{X}) = \mathbf{W} \cdot \mathbf{F}(\mathbf{X}),$$

where $\mathbf{W} \in \{(w_0, \dots, w_5) | w_i \in [0, 1], \sum w_i = 1\}$ and,

$$\mathbf{F} = (\tilde{f}_0, \dots, \tilde{f}_5).$$

Where noise in the y direction is introduced as

$$\tilde{f}(x) = f(x) + \epsilon,$$

with

$$\epsilon \sim N(0, \sigma^2),$$

and $\sigma \in \mathbb{R}$ not dependent on x . The isolated input-output functions, f_i are represented as

$$f \in \mathbb{P}_5[x],$$

where $\mathbb{P}_n[x] = \{p_0 + p_1x + p_2x^2 + \dots + p_nx^n | p_i \in \mathbb{R}\}$. For each dataset the degree of the polynomials f_i are randomly chosen from the range specified, for example \mathbb{P}_{a-b} indicates the degrees of f_i are chosen integers from the range from $a - b$ inclusive. Then the coefficients are randomly selected uniformly from the interval $(-10, 10)$.

This produces a dataset for a regression where the known relationships between input and output variables are $\mathbf{W} \cdot \mathbf{F}$ or $w_i \cdot f_i \forall i$, which are all functions and will be denoted f_i , and is the isolated relationship between the input i and the output. A range of curvatures of f_i ; noise in the y direction; and covariance matrices for generating X are tested, Table 4.2, to identify the types of data where the error measure is effective.

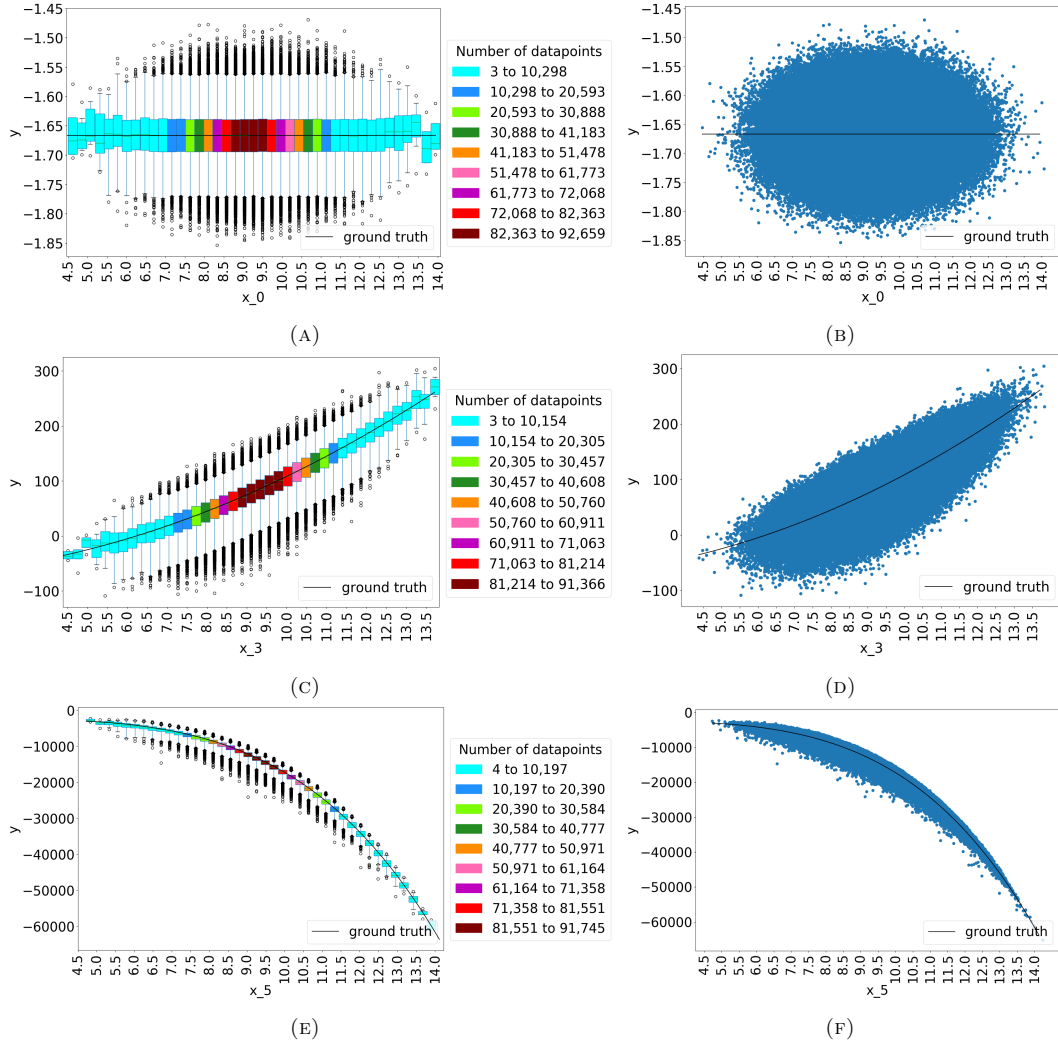


FIGURE 4.4: Illustrations of different datasets generated for this study: A) boxplots showing the spread of y values for each interval of x_0 along with f_0 from a dataset of **type a** from Table 4.2, shading of each boxplot shows the quantity of datapoints in any given interval, illustrating the sparse areas of data created at the tails of the x_1 distribution; B) scatter plot of x_0 to y for a dataset of **type a** showing f_0 ; C) boxplots showing the spread of y values for each interval of x_3 along with f_3 from a dataset of **type h**; D) scatter plot of x_3 to y for a dataset of **type h** showing f_3 ; E) boxplots showing the spread of y values for each interval of x_5 along with f_5 from a dataset of **type l**; F) scatter plot of x_5 to y for a dataset of **type l** showing f_5 .

The data generated for this study aims to replicate a scenario where a single target variable is dependent on 6 independent input variables, with their own specific relationship to the target. Each input variable has an arbitrarily generated polynomial relationship, f_i , with noise

independent of the other input variables, the target, y , is the weighted average of each f_i .

For datasets with non-linear polynomials this creates non-Gaussian noise in the y distribution, violating assumption (ii). The reason for violating this assumption is that many applications require modelling of systems with non-Gaussian noise in the target variable. Although the noise introduced in these artificial datasets has the same Gaussian distribution for each input variable, their introduction at the input variable level is sufficient to provide non-Gaussian noise in the output variable. The interaction between input-level noise also produces heteroscedastic datasets, which violates assumption (iii).

The noise distributions are illustrated by the scatter plot in Figure 4.4e and 4.4f, the combination of noise from the other 5 inputs creates an increased spread below the concave $x_5 - y$ relationship, explained by Jensen’s inequality. This skewed conditional distribution of the output variable means that often the conditional median approximates the ground truth closer than the conditional mean.

Input variables are generated as normal distributions around an arbitrary mean, so there is not sufficient data across the entirety of each x_i domain, this violates assumption (iv). The shading of boxplots in Figure 4.4a, 4.4c and 4.4e illustrates the normal distribution of x_i values in one of the generated datasets, where a mean of 9.25 and standard deviation of 1, has created ranges of sparse data for $x_i < 6$ and $x_i > 13$.

This methodology produces a dataset which is not equivalent to any specific dataset, but allows validation of the different regression error measures on a dataset that can be controlled whilst replicating the main features from real data. However, the artificial input values all have a normal distribution with equal mean and median values whereas many real-life situations include variables which show more complex distributions. Arbitrary regression datasets from the UCI Machine Learning Repository (Dua and Graff 2017), as well as privately available datasets, were analysed to contextualise the artificial datasets used in this study. The input-output relationships in real datasets, as diverse as wind turbine power, fish toxicity and housing price regression data, are seen to be similar to those used to generate the artificial datasets in terms of relationship gradients, noise levels and continuity.

4.4 Assess New Metric

4.4.1 Effectiveness on a Range of Datasets

Three separate datasets are tested for each type of dataset in Table 4.2, with different randomly generated parameters. For each of these datasets 1,500 neural networks are trained using normal

TABLE 4.3: Selected hyperparameters for assessing new error measure

Hyperparameter	Value or set
Number of hidden layers	[1,3]
Number of neurons in each hidden layer	[1,1000]
Number of epochs	20
Early Stopping Patience	5
Early Stopping Tolerance	0
Loss function	Mean Absolute Relative Error
Performance Measures	Mean Absolute Relative Error, Mean Fit to Median and Mean Fit to Mean
Optimiser	AdaMax (Kingma and Ba 2014)
Learning rate, β_1 , β_2 , ϵ	0.001, 0.9, 0.999, 1^{-7}
Activation Function	ReLU
Regulariser	None
Dropout	None
Initialiser	Random Normal ($\mu = 0, \sigma = 0.1$)

backpropagation² of random sizes ranging from (1,1) to (3,1000) and a loss function based on the Mean Absolute Relative Error, full hyperparameters are provided in Table 3.1. After training, each network is tested with the newly derived Mean Fit to Median error measure, as well as Mean Absolute Relative Error and Mean Squared Error as these are the most popular error measures used for regression. The Mean Absolute Relative Error is used as the Minkowski-r metric to compare to because the curvature of the input-output relationships means the conditional median is closer to the ground truth than the conditional mean (Bishop 1995), although the Mean Square Error was also trialled due to its prolific use in the field and showed similar behaviour to the Mean Absolute Error.

Since the datasets are all artificial, the underlying relationships within them are known explicitly, therefore it is possible to assess how well each network fits the ground truth of the dataset. This measure is called ‘Mean Fit to the Ground Truth’ which is derived similarly to the Mean Fit To Median with the difference that the input-output function ϕ is used to generate the dataset,

$$\text{Mean Fit to Ground Truth} = \frac{1}{m} \sum_{j=1}^m \frac{1}{n} \sum_{i=1}^n \left| \phi \left(\text{mid}(X_i^j), \text{med}(X^k) \text{ for } k \neq j \right) - p_{X_i}^j \right|. \quad (4.6)$$

This involves cycling from the minimum to maximum observed values, taking the true value of ϕ for the midpoint of each partition in the input domain, with all of the other inputs held at the median value. It is only possible to calculate this measure for artificial datasets.

²All computation in this Chapter is performed on the Iridis Compute Cluster at the University of Southampton.

The errors from 1,500 separate network runs are calculated and each error measure is normalised independently as the datasets are artificial, therefore the absolute magnitude of an error measure has limited meaning. The error measures are compared to assess if the Mean Fit to Median error measure approximates the Mean Fit to the Ground Truth of the dataset better than the Mean Absolute Relative Error or Mean Squared Error. If there is a correlation between the Mean Fit to the Median and the Mean Fit to the Ground Truth then the Mean Fit to the Median can be used as a error measure that acts as a proxy for how well a network approximates the ground truth of these datasets.

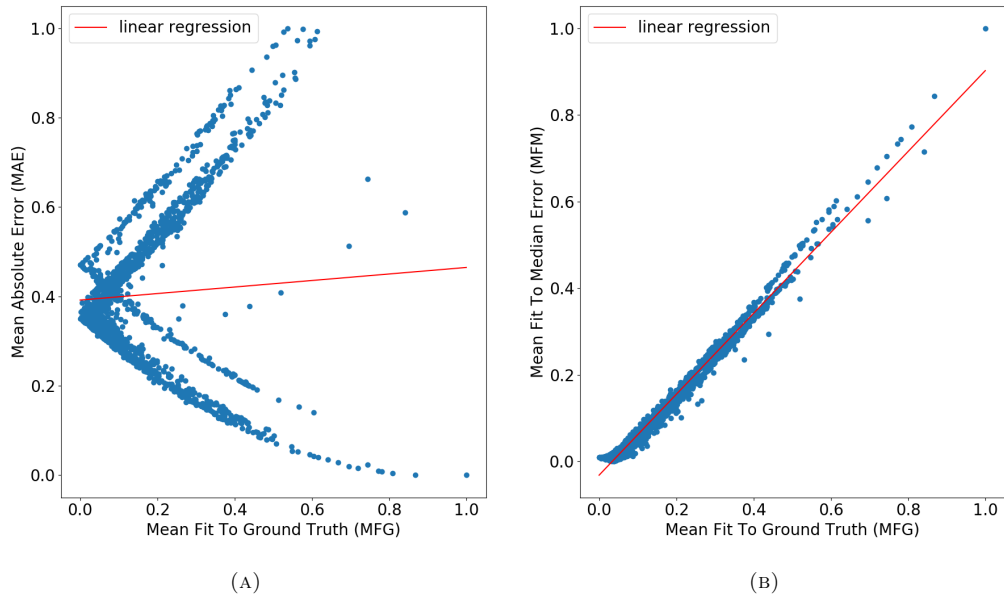


FIGURE 4.5: 1,500 networks assessed on the dataset of **type d**, where the data is fit with a linear regression a) the Mean Absolute Relative Error against Mean Fit To Ground Truth with R^2 of 0.0036 and b) the Mean Fit To Median Error against Mean Fit To Ground Truth with R^2 of 0.9801.

The relationships between the Mean Absolute Relative Error and the Mean Fit to the Ground Truth show a high variation across the different datasets, nearly all of the relationships involve bifurcations, although this becomes less prominent as the degree of the polynomials in the dataset decreases. Figure 4.5a shows an almost perpendicular bifurcation at 0.4 Mean Absolute Relative Error whereas Figure 4.6a and 4.7a have a more irregular pattern but still show prominent bifurcations. Contrastingly, the Mean Fit to the Median and the Mean Fit to the Ground Truth clearly approach more of a one-to-one relationship. This is quantified using R^2 from a linear regression performed on the normalised data, the lines are shown on Figures 4.5, 4.6 and 4.7.

For a dataset with entirely linear f_i and a low standard deviation of noise, there is almost total agreement between the Mean Fit to the Median and Mean Fit to the Ground Truth, Figure 4.5b. This correlation lessens as the standard deviation of noise increases and as the f_i increases in degree, or curvature. Figure 4.6b shows results from a dataset with increased curvature but low noise, compared to Figure 4.5b, and although the correlation is still prominent, the R^2 has

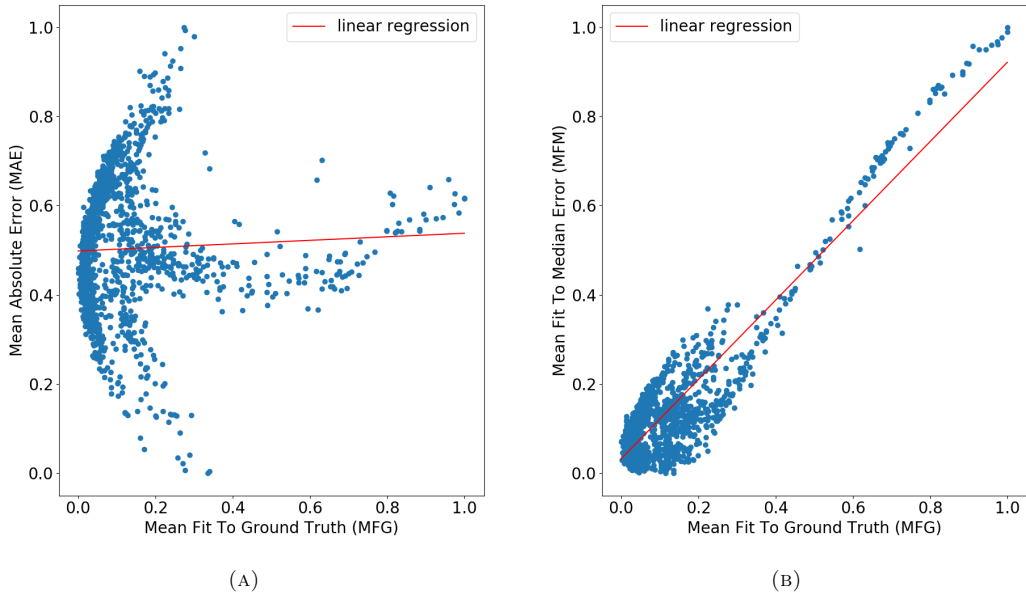


FIGURE 4.6: 1,500 networks assessed on the dataset of **type b**, where the data is fit with a linear regression a) the Mean Absolute Relative Error against Mean Fit To Ground Truth with R^2 of 0.0016 and b) the Mean Fit To Median against Mean Fit To Ground Truth with R^2 of 0.8464.

decreased from 0.98 to 0.85. The datasets with polynomial f_i of degrees 4-5 and high noise levels are hard to model, requiring at least three hidden layers in a neural network to approximate accurately. The results from this dataset show the lowest correlation between Mean Fit to the Median and Mean Fit to the Ground Truth with an R^2 of 0.52 Figure 4.7, but this is still significantly higher than the correlation between Mean Absolute Relative Error and Mean Fit to the Ground Truth.

This decrease in correlation between Mean Fit to the Median and Mean Fit to the Ground Truth as more noise and convexity is introduced to the datasets can be explained by Jensen’s inequality. The increase in noise and convexity of f_i create a larger skew in the conditional distributions of the output variable, this skew causes the mean and median of the conditional distribution to move away from the ground truth or $\phi(X)$. If this median is not an accurate representation of the ground truth then the Mean Fit to the Median will not be as effective as it directly measures the distance of a prediction from the conditional median. However, the Mean Fit to the Median approximates the Mean Fit to the Ground Truth better than the Mean Absolute Relative Error for all analysed datasets. Networks with low Mean Fit to the Median therefore replicate the ground truth better than networks with low Mean Absolute Relative Error.

The R^2 values from all of the regressions on the 36 datasets are collated and this is illustrated in Figure 4.8a and 4.8b. Comparing the average R^2 value of the linear regressions performed for each different type of dataset, shows where the Mean Fit to the Median is most effective. All of the R^2 values for the Mean Absolute Relative Error are lower, 0.02-0.57, than their Mean

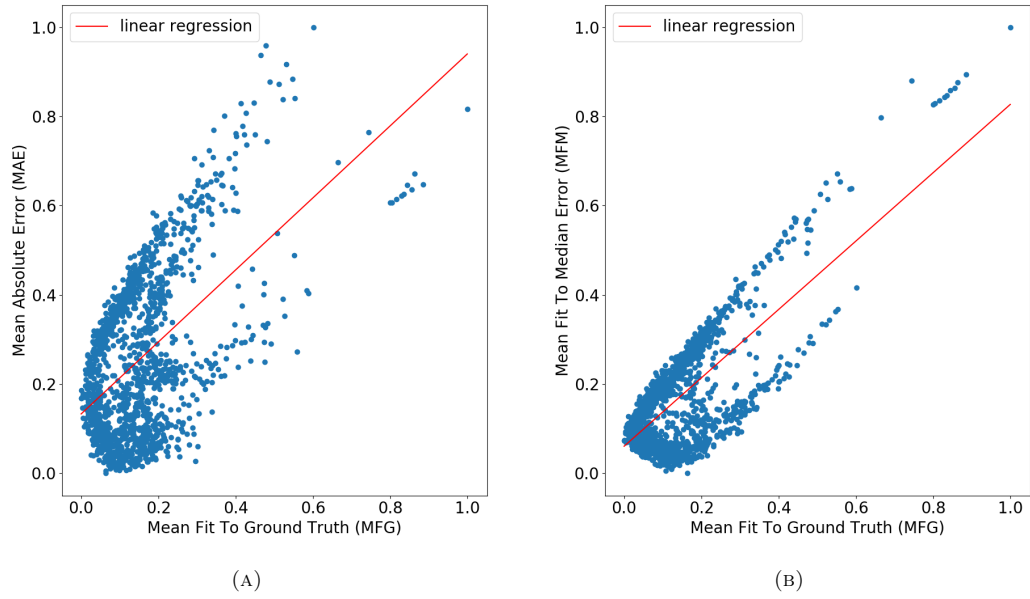


FIGURE 4.7: 1,500 networks assessed on the dataset of **type 1**, where the data is fit with a linear regression a) the Mean Absolute Relative Error against Mean Fit To Ground Truth with R^2 of 0.2704 and b) the Mean Fit To Median against Mean Fit To Ground Truth with R^2 of 0.5184.

Fit to the Median counterparts, 0.39-0.99, from Figure 4.8a, where a higher R^2 indicates a tighter positive correlation between the error measure and the Mean Fit to the Ground Truth of the dataset. For datasets with lower degree polynomial relationships the R^2 values for the Mean Fit to the Median are particularly high, approaching 1, Figure 4.8b. Datasets with these characteristics should be simple enough to analyse that advanced machine learning techniques, such as neural networks, do not produce any further insight than simpler analysis techniques.

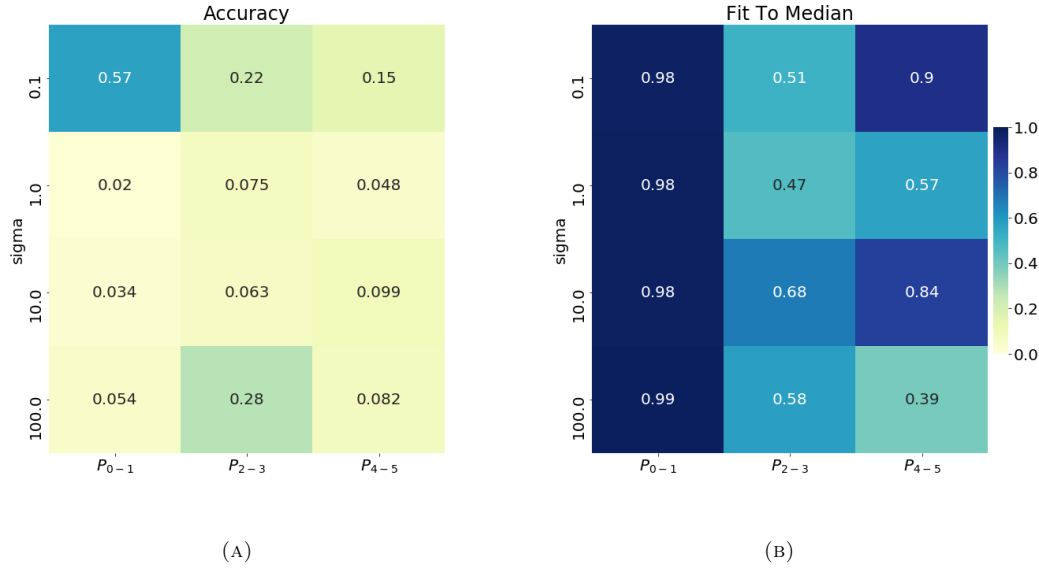


FIGURE 4.8: Heatmaps illustrating the average R^2 of a error measure to the Mean Fit To Ground Truth from the different types of dataset, stipulated in Table 4.2 a) the average R^2 of the traditional, Mean Absolute Relative Error, error measure against the Mean Fit To Ground Truth and b) the average R^2 of the proposed, Mean Fit To Median, error measure against the Mean Fit To Ground Truth.

All of the R^2 values for the Mean Fit to the Median are higher than the Mean Absolute Relative Error values, meaning the Fit to Median is better at assessing how well a network models the ground truth than the Mean Absolute Relative Error. This increase in R^2 ranges from 0.29-0.96, with an average of 0.6, across all 36 datasets. Choosing networks with a low Mean Fit to the Median thus ensures a better modelling of the underlying relationships within a dataset, when compared to relying on most commonly used error measures.

4.4.2 Investigation into the Divergence

The causes of the bifurcations in the relationship between Mean Absolute Relative Error and the Mean Fit to the Ground Truth are investigated. A major factor affecting the error of a trained network is its size and shape. The networks trained had a random number of layers and neurons, with a maximum size of (3,1000) and therefore the bifurcations imply that some networks trained with Mean Absolute Relative Error have a bias to the ground truth and that some have a bias to a pattern that is not the ground truth, negatively affecting their ability to replicate this relationship. One of the assumptions required for minimum Mean Absolute Relative Error to model the conditional median of a dataset is the use of a sufficiently large neural network. As randomly sized networks are used in this study this assumption is not necessarily satisfied. Therefore, to evaluate the effect of the size of a network on the error, the complexity, or number of connections, are indicated by colour in Figure 4.9.

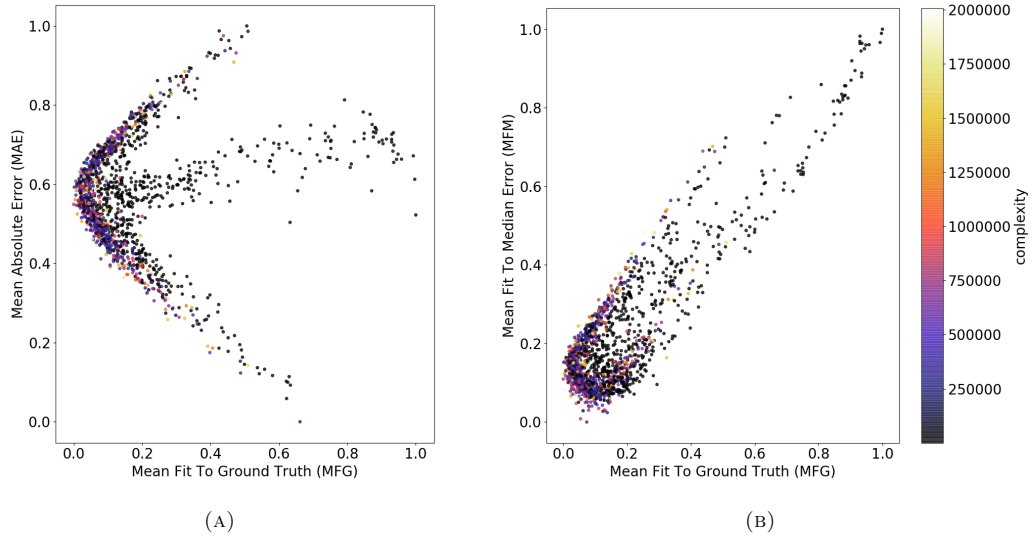


FIGURE 4.9: 1,500 networks assessed on the dataset of **type e** with a) Mean Absolute Relative Error against Mean Fit To Ground Truth and b) Mean Fit To Median against Mean Fit To Ground Truth. Where complexity, or number of connections in each network, is indicated by the colour of each point and shows no pattern for either plot.

There is no clear relationship between the size of the network and error for either the Mean Absolute Relative Error, Figure 4.9a, the Mean Fit to the Median or the Mean Fit to the Ground Truth, Figure 4.9b. This suggests that the bifurcations are not caused by over or under-fitting of the networks. Although some relationship between the network size and error would be expected when using a large range of network sizes, as all of the networks employ early stopping it is suggested this procedure decreases the chances of networks overfitting.

It is confirmed that, on the whole, networks with low Mean Fit To Median also have a low Mean Absolute Relative Error but not the lowest observed Mean Absolute Relative Error. Hence, fitting to the ground truth of a dataset does not produce the lowest point-to-point accuracy but does provide an acceptable level. For all 36 trialled datasets, 70% show lower Mean Absolute Relative Error as Mean Fit To Median lowers, this means the Mean Fit To Median error measure can be used alongside Mean Absolute Relative Error to produce networks which do not compromise the point accuracy of prediction to target significantly, but do increase the fit to the ground truth of the dataset.

The characteristics of the datasets, relating to their distributions and underlying relationships, are investigated as possible causes of the bifurcations. The normalised difference between the Mean Fit to the Median and ‘Fit to Mean’ error measures are indicated by datapoint colour in Figure 4.10. The ‘Fit to Mean’ error measure is derived similarly to the Mean Fit to the Median except the proxy curve is $E[\phi(X)]$ as opposed to $med(\phi(X))$ for the median.

The networks with the lowest Mean Absolute Relative Error approximate relationships which are closer to the mean of the data than the median of the data and hence have a positive difference

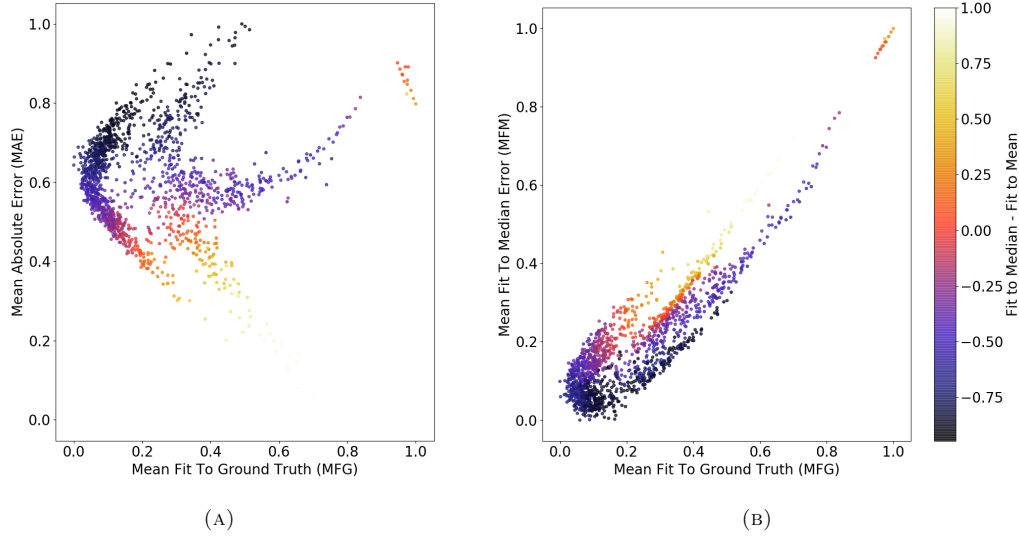


FIGURE 4.10: 1,500 networks assessed on the dataset of **type 1** with a) Mean Absolute Relative Error against Mean Fit To Ground Truth and b) Mean Fit To Median against Mean Fit To Ground Truth. The difference between the Mean Fit To Median and ‘Fit to Mean’ of each network is indicated by their colour, with networks where the ‘Fit to Mean’ is lower coloured lighter and networks where the Mean Fit To Median is lower coloured darker.

between Mean Fit to the Median and ‘Fit to Mean’. These lighter points in Figure 4.10a form the bottom tail of the bifurcation and have a higher Mean Fit to the Ground Truth error than the main body. This suggests that as networks fit the conditional mean of a dataset more, this approximation departs from fitting the ground truth. As the difference between Mean Fit to the Median and ‘Fit to Mean’ becomes negative the networks fit closer to the conditional median of the dataset than the mean, indicated by the darker datapoints. The networks which have the lowest Mean Fit to the Median, comparative to the ‘Fit to Mean’, diverge away from the main body as the upper part of the bifurcation as the Mean Absolute Relative Error from these networks is high.

Therefore, the bifurcations noted for these datasets may be caused by the bias created by training with the Mean Absolute Relative Error. The minimum Mean Absolute Relative Error does not guarantee a fit of the conditional median as shown in previous sections, in this particular dataset the networks with lowest Mean Absolute Relative Error are closer to the conditional mean of the dataset. As also discussed above, for the datasets used in this study the conditional median is a better approximation of the ground truth than the conditional mean, hence the lower tail in Figure 4.10a which contains networks which fit closely to the conditional mean but do not fit the ground truth well. This shows an unhelpful inductive bias is created by the use of the Mean Absolute Relative Error.

This bias produces networks with inconsistent results, which can be noted in the multi-modal distributions of Mean Absolute Relative Errors, Figure 4.11. For 34 of the 36 datasets a chi-squared test that the distribution is a normal distribution is rejected at the 99.99% confidence

level. A multi-modal distribution of Mean Absolute Relative Error suggests that there are multiple local minima on the Mean Absolute Relative Error surface that the gradient descent method gets ‘stuck’ in. The local minima are represented by the multiple modes of the distributions, none of which are the global minimum on the Mean Absolute Relative Error surface, if it has been found, which is represented by the lowest observed Mean Absolute Error.

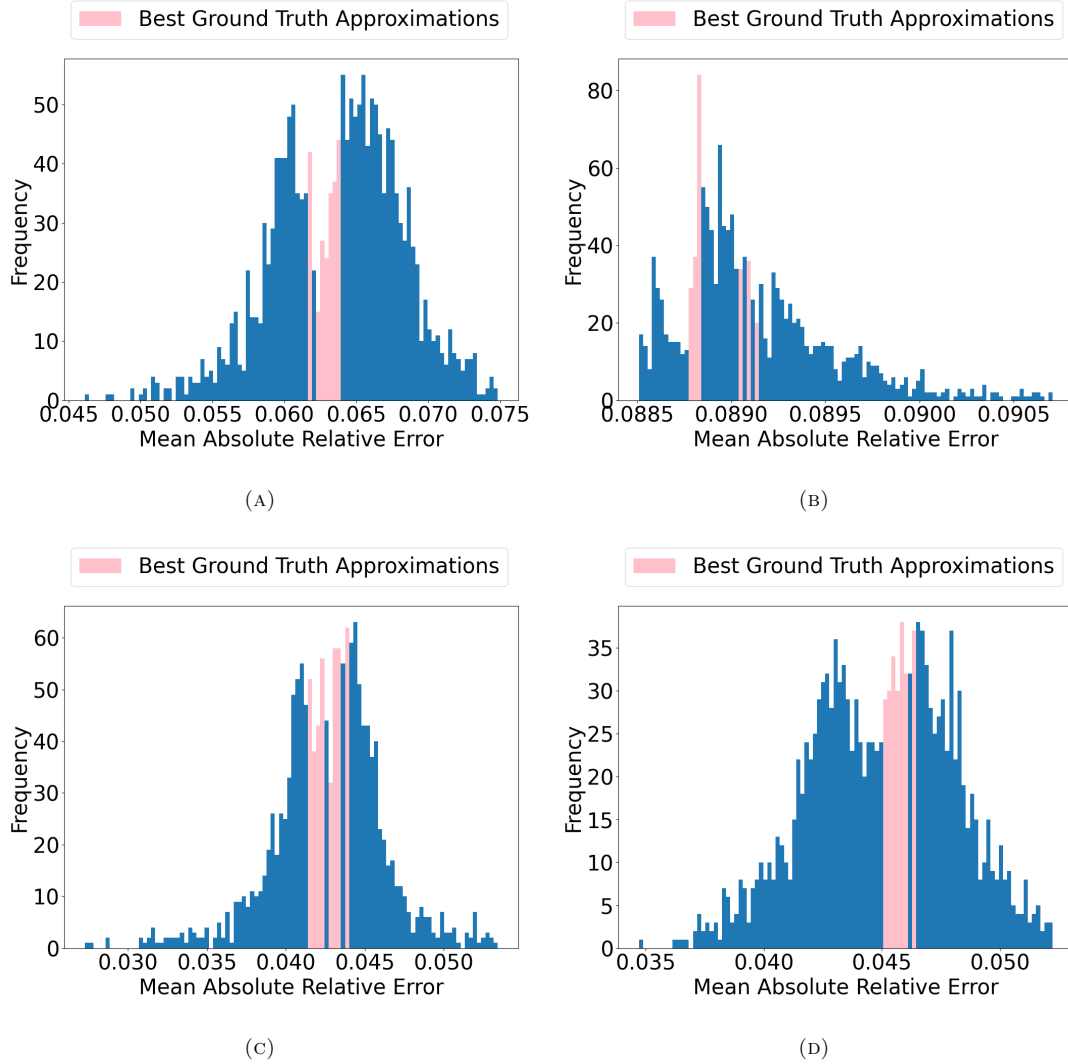


FIGURE 4.11: Histogram illustrating the multi-modal distributions of Mean Absolute Relative Errors from 1,500 randomly sized neural networks for 4 of the 36 datasets. The bins which contain the 10 networks which best approximate the ground truth are coloured pink, none of which produce the lowest observed Mean Absolute Relative Errors. A) dataset of **type b** has a p-value of 8^{-8} , B) dataset of **type a** has a p-value of 1^{-61} , C) dataset of **type l** has a p-value of 3^{-24} , and D) dataset of **type h** has a p-value of 2^{-4} .

As illustrated in Figure 4.10a the networks producing the lowest Mean Absolute Relative Errors are not producing the best approximation of the ground truth. The Mean Absolute Relative Error from the 10 networks which approximate the ground truth the best are highlighted in the histograms. Across all 36 datasets, the networks with the best approximation of the ground truth never produce the best Mean Absolute Relative Error and rarely produce the most common

Mean Absolute Relative Error. The exception is one of the datasets of **type a**, illustrated in Figure 4.11b, where the other 9 networks which best approximate the ground truth are neither the most common or lowest observed Mean Absolute Relative Error. These error distributions suggest that there are multiple possible biases from the Mean Absolute Relative Error, or local minima on the error surface, but none of these biases relate to a good approximation of the ground truth of the dataset.

It is therefore improbable that any given network, trained with a Mean Absolute Relative loss function, will provide a good approximation of the ground truth. The biases from using a Minkowski-r metric do not relate to ground truth approximation. As neural networks display low neuroplasticity, these biases will be decided early on in the training process, which makes it difficult to overcome using conventional error measures. This suggests that the convergence to one local minima originates from inherent stochasticity in the training of a neural network. Explicitly this is a combination of the initialisation and the order the dataset is shown to the network in the first epoch, as well as any stochastic elements introduced by learning rules.

4.5 Ship Power Prediction

The Fit to Median Error is tested on the ship power prediction problem discussed in Chapter 3, to illustrate its use on a dataset where the ground truth is not known. For the single ship prediction dataset used in Section 3.4 1,000 neural networks of size (3,300) are trained separately using the Mean Absolute Relative Error as a loss function and other hyperparameters specified in Table 3.1. Networks with the lowest observed Fit to Median Errors produce input-output variable curves which approximate the conditional average of the dataset closer and more consistently than the networks with the lowest Mean Absolute Relative Errors.

4.5.1 Similarity to Artificial Data

The distribution of Mean Absolute Relative Errors from the 1,000 network runs and the correlation to the Mean Fit to Median Error illustrate the same properties as the observed from the artificial datasets in Section 3.2.2. There is no relationship between Mean Fit to Median Error and Mean Absolute Relative Error, Figure 4.12a, with an R^2 value of 3^{-32} . However, Figure 4.12a does illustrate the bimodal normal distribution of the Mean Absolute Relative Error values intersecting with the unimodal normal distribution of the Mean Fit to Median Error values.

For over 75% of the artificial datasets the R^2 between Mean Fit to Median and Mean Absolute Relative Error is less than 0.3. It is suggested this decrease in relation between error measures for the real datasets is due to the intentionally simplistic relationships and noise profiles employed

in the artificial data, which produces a smoother Mean Absolute Relative Error surface and therefore less local minima.

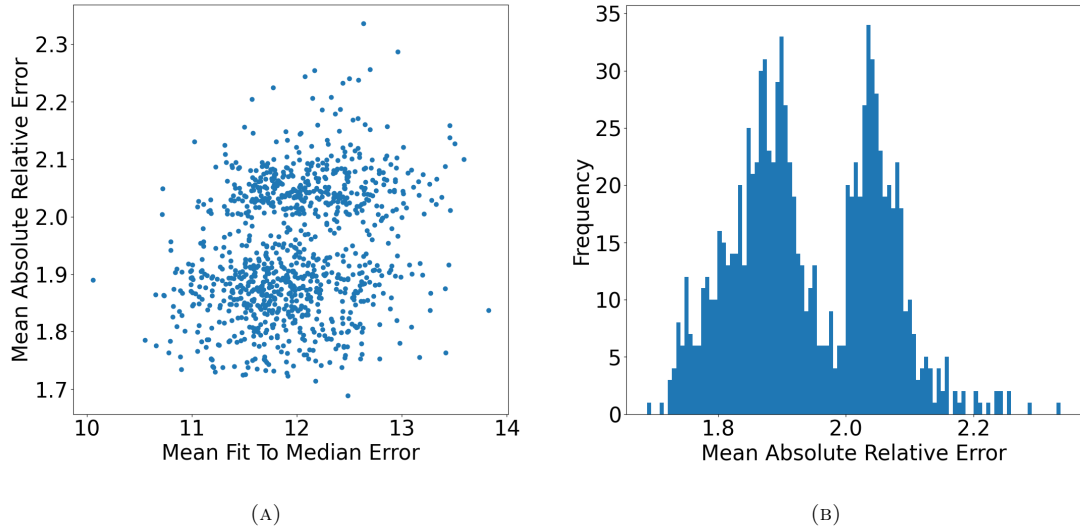


FIGURE 4.12: (A) Mean Fit to Median Error and Mean Absolute Relative Error of the 1,000 neural networks to predict ship powering, with an R^2 of 3^{-32} , (B) the bimodal distribution of Mean Absolute Relative Error.

The multiple biases noted on the Mean Absolute Relative Error surface from the artificial datasets are more apparent on the ship dataset. The bimodality of Mean Absolute Relative Error is noted in both Figure 4.12a and 4.12b. This illustrates that the artificial datasets are representative of real regression problems, in so far as the distribution of traditional error measure values and their relation to the new Fit to Median Error values.

4.5.2 Potential for Reduced Compute Requirements

As all networks used are the same size, the training time is equivalent to the number of epochs executed before early stopping terminates training. The relationship between training time, Mean Absolute Relative Error and Mean Fit to Median Error shows that networks trained for longer consistently have lower Mean Absolute Relative Errors, Figure 4.13. However there is no relation between training time and Mean Fit to Median Error, suggesting that networks which model the conditional averages accurately can be found with less computational time than those with low Mean Absolute Relative Error.

The set of networks with Mean Absolute Relative Error values above 2%, which form the right peak in Figure 4.12b, all took less than 7 minutes to train. This illustrates that the local minima producing around 2.05% Mean Absolute Relative Error is identified by around half the networks and cannot be overcome, so training terminates once it is found. Those networks which do not identify it take more epochs to locate the other dominant local minima producing Mean

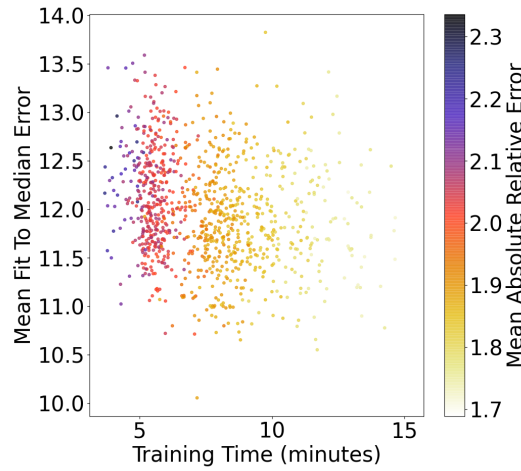


FIGURE 4.13: Mean Fit to Median Error against Training Time with Mean Absolute Relative Error illustrated by colour of point.

Absolute Relative Errors around 1.9%. Both of the local minima for Mean Absolute Relative Error produce a range of Mean Fit to Median Error values. This shows the flexibility for a network to have a low Mean Absolute Relative Error as well as a low Mean Fit to Median Error, but also demonstrates the requirement for the new error measure, as there is no certainty that networks with low Mean Absolute Relative Errors also have low Mean Fit to Median Errors.

4.5.3 Improved Consistency of Input-Output Curves

The 5 networks with the lowest Mean Absolute Relative Error and the 5 with the lowest Mean Fit to Median Error are selected and the isolated input-output relationships learnt are visualised Figure 4.14 and 4.15, figures for the other input variables are found in Appendix B. For all input variables the input-output curves are more consistent for the networks with the lowest Mean Fit to Median Error and this is most notable in areas of sparse data which is indicated by the lightest coloured boxplots.

The decreased spread in predicted relationships for networks producing the lowest Mean Fit to Median is more notable for the input variables less correlated to shaft power. For networks with the lowest Mean Absolute Relative Error the most correlated variable, ship speed Figure 4.14a, shows more consistent relationships around 16knots, where the data is most dense. However, these curves approximate a relationships below the interquartile range of the data. This implies that the local minimum on the Mean Absolute Relative Error surface which is approximated does not correspond to a close modelling of the conditional averages.

The improved consistency of networks with the lowest Mean Fit to Median Errors is most notable for the areas of sparse data in the input domain. For the wave height variable, waves above 5m

have poor predictions from the networks with lowest Mean Absolute Relative Error Figure 4.15a and predictions significantly closer to the conditional averages from the networks with lowest Mean Fit to Median Error, Figure 4.15b. The lightest coloured boxplots in these figures contain less than 0.4% of the dataset, highlighting the inability for networks producing low traditional error measures to perform in sparse areas of data.

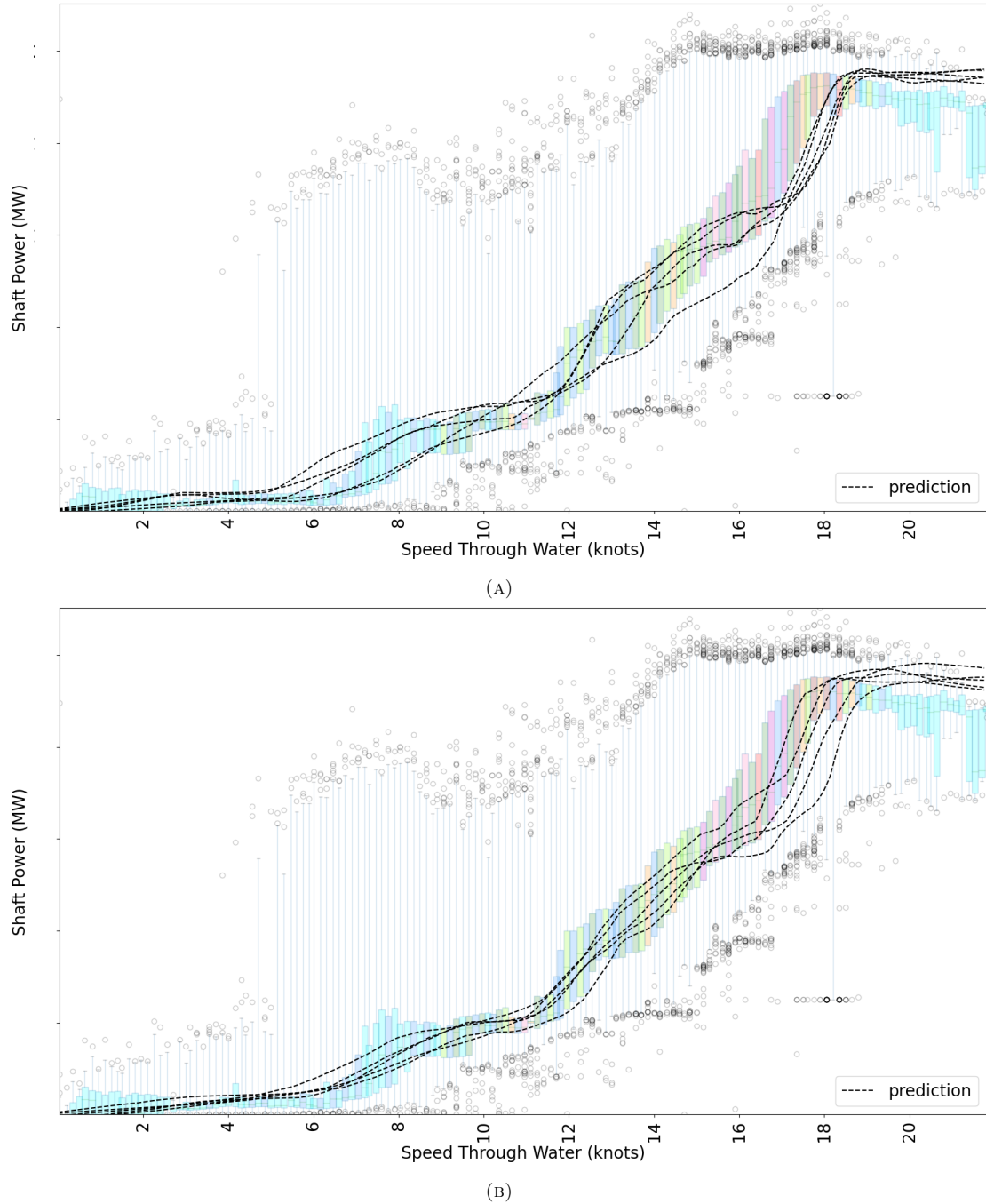


FIGURE 4.14: Visualisations of isolated relationships learnt, between ship speed and power. (A) from the 5 networks out of 1,000 with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.

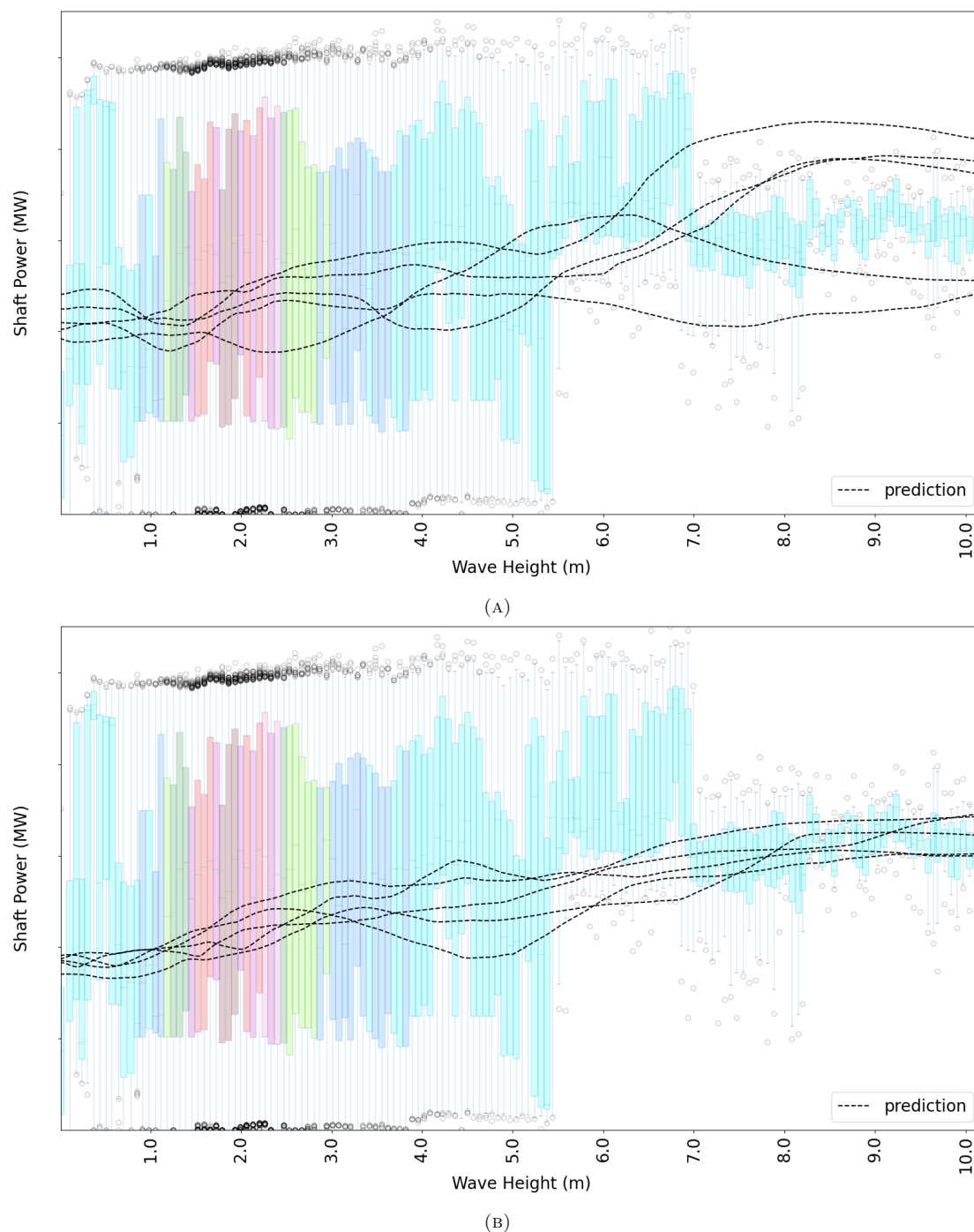


FIGURE 4.15: Visualisations of isolated relationships learnt, between wave height and power. (A) from the 5 networks out of 1,000 with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.

The spread of predictions for each variable is quantified by taking the Euclidean distance between the highest and lowest prediction values at the midpoint of every boxplot along the input domain. The 1,000 networks are sorted, first based on their Mean Absolute Relative Error value and this measure is calculated for increasing numbers of network predictions. The networks are then sorted based on their Mean Fit to Median Error value and the same process is performed. This

creates two curves illustrating the effect of the error measures on the spread of predicted input-output relationships, Figure 4.16.

Since the same 1,000 networks are utilised for both curves, the final average spread is the same for both error measures. The gradient of the curve however, depends on the efficacy of each error measure at producing consistent predictions. Illustrated here is the average over all input variables, Figure 4.16. It is confirmed that all input variables show the same trend; that along the majority of the range of increasing numbers of networks the networks ranked by Mean Fit to Median Error have less of a spread. These figures are provided in Appendix B.

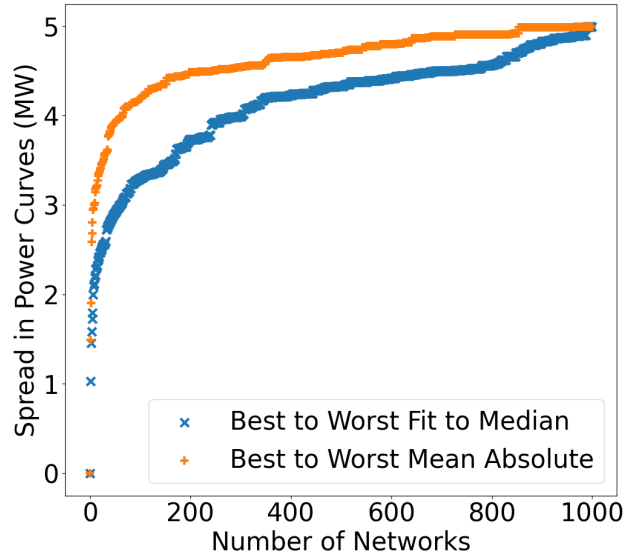


FIGURE 4.16: Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error.

Of the 5 networks with the lowest Mean Fit to Median Error, four have Mean Absolute Relative Errors in the local minima with errors around 1.9% and one has a Mean Absolute Relative Error in the minima with higher errors of over 2%. This suggests there is no need to sacrifice other error measures to produce low Fit to Median errors. However it is noted that networks with the lowest Mean Absolute Relative Errors produce less consistent input-output curves than networks with the highest Mean Absolute Relative Errors. Since all networks are trained to minimise the Mean Absolute Relative Error and the total range in observed Mean Absolute Relative Errors across the 1,000 networks is 0.65%, these differences are negligible.

4.6 Unseen Vessel Power Prediction

The fleet dataset comprised of 9 different vessels, used to replicate a scenario where power prediction is required for a ship which does not record data in Section 3.5, is used to further

illustrate the potential of the Mean Fit to Median Error measure. This problem is chosen as it allows for ‘true’ extrapolation capabilities to be assessed. So far in this thesis ‘extrapolation’ has been used to refer to areas of sparse data, as well as areas outside of the training domain. Since all 9 ships encounter different ranges of input variables, predicting for an unseen ship provides an opportunity to analyse network performance in regions of input domain not experienced before. The input-output relationships learnt by the trained networks are visualised, across the full range of observed input variable values for all 9 ships, regardless of whether the network was exposed to data across this full range.

For each dataset containing data from 8 of the 9 available ships, 100 neural networks of size (3,300) are trained separately using the Mean Absolute Relative Error as a loss function and other hyperparameters specified in Table 3.1. Networks with the lowest observed Fit to Median Errors produce input-output variable curves which extrapolate more accurately and more consistently than the networks with the lowest Mean Absolute Relative Errors.

The same procedure used in Section 3.5 is followed, with the exception of running 10 times the number of repeats to allow sorting of the trained networks based on the Fit to Median Error measure. To replicate the scenario where power prediction is required for a ship with no recorded data, errors are calculated only using the fused dataset of 8 ships. Training is performed using 75% of the fused dataset, to allow Mean Absolute Relative Error to be calculated on a dedicated testing set from this data fusion. The Fit to Median Error is calculated using the conditional median curves from the same fused dataset, again to ensure a realistic setup.

For each fused dataset, the trained networks are sorted based on the error values they produce. The range of the relationships learnt by the 5 networks with lowest Fit to Median Error and the 5 networks with the lowest Mean Absolute Relative Error are illustrated against the range of training curves. That is, the range of the median power conditioned on the input variable, the area between the curves visualised in Figure 3.23, for the 8 ships in the data fusion.

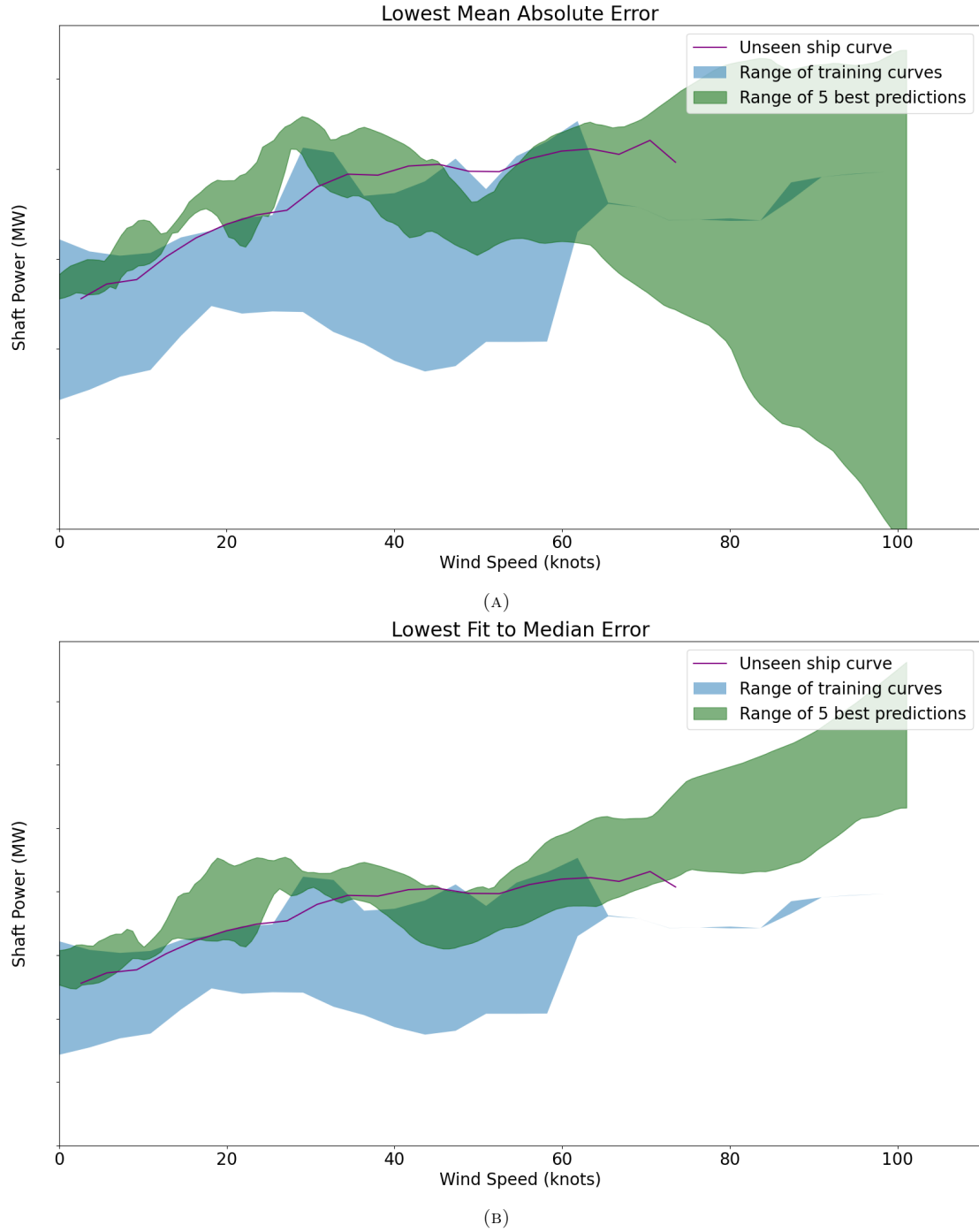


FIGURE 4.17: Visualisations of range of isolated relationships learnt, between wind speed and power for Ship A. (A) from the 5 networks with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.

As well as the networks with lowest observed Mean Fit to Median Error modelling more consistent input-output relationships compared to the networks with the lowest Mean Absolute Relative Errors, Figure 4.17, the networks with the lowest Mean Fit to Median Errors also model more accurate extrapolated predictions. The range of predicted power requirements for wind speeds of 100 knots is the entire range of observed shaft powers for the networks with the lowest Mean

Absolute Relative Error, Figure 4.17a. Whereas, the networks with the lowest Mean Fit to Median Errors predict within a range of 10MW, Figure 4.17b. A more extensive selection of visualised predictions are provided in Appendix C.

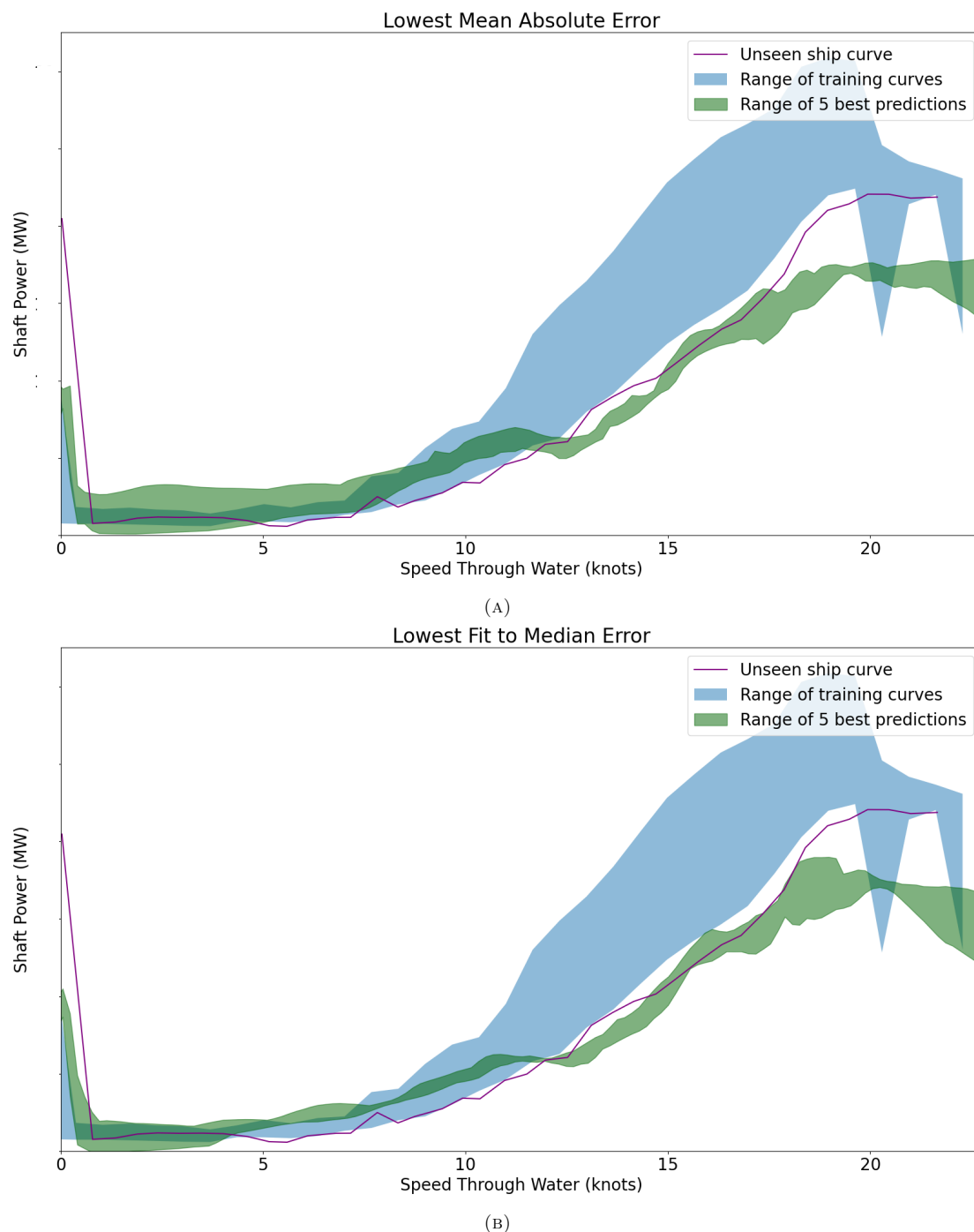


FIGURE 4.18: Visualisations of range of isolated relationships learnt, between ship speed and power for Ship F. (A) from the 5 networks with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.

The difference in consistency of predicted relationships is less notable for the input variables which are more highly correlated to powering, ship speed Figure 4.18. Networks with low Fit to

Median Error and networks with low Mean Absolute Relative Error produce consistent speed-power curves, with less than 2.5MW range. Both under-predict power requirements for ship speeds of above 18 knots, Figures 4.18a and 4.18b. However, this region of input space is akin to extrapolated regions due to the low quantity of datapoints, Figure 3.3 and 3.20. The similarity between the consistency and accuracy of predicted speed-power curves for networks with low Mean Absolute Relative Error and networks with low Mean Fit to Median Error show that input-output relationships from networks with the lowest Mean Fit to Median Error are either better or no worse than relationships from networks with the lowest Mean Absolute Relative Error.

The spread of predictions is visualised in the same way as Figure 4.16: spread in input-output relationship for each variable is quantified by taking the Euclidean distance between the highest and lowest prediction values; networks are sorted and this measure is calculated for increasing numbers of network predictions. When aggregated across all input variables and all ships, there is a notable increase in consistency of predictions for the best to worst Mean Fit to Median Error compared to the best to worst Mean Absolute Relative Error, Figure 4.19. This difference is noted across the entire list of ordered networks, where networks ordered by Mean Fit to Median Error are 2MW more consistent for quartiles from 5-65%. The total average spread of the 900 networks is only attained by the the Mean Fit to Median when including the top 95% of networks.

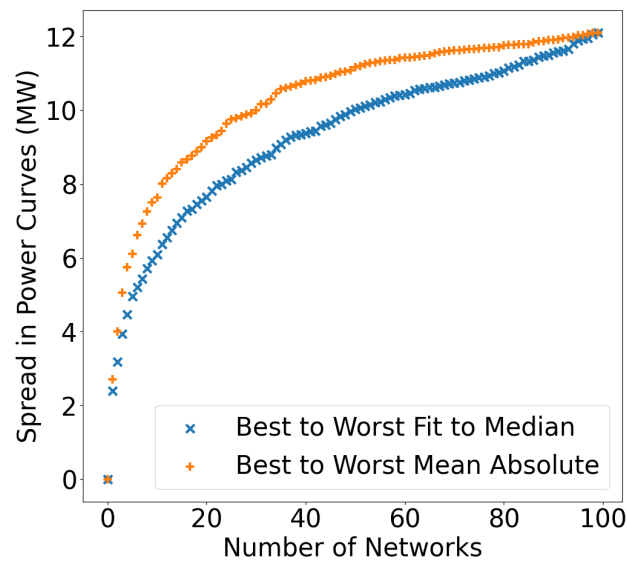


FIGURE 4.19: Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error for unseen ship prediction, averaged over all input-output curves for all ships.

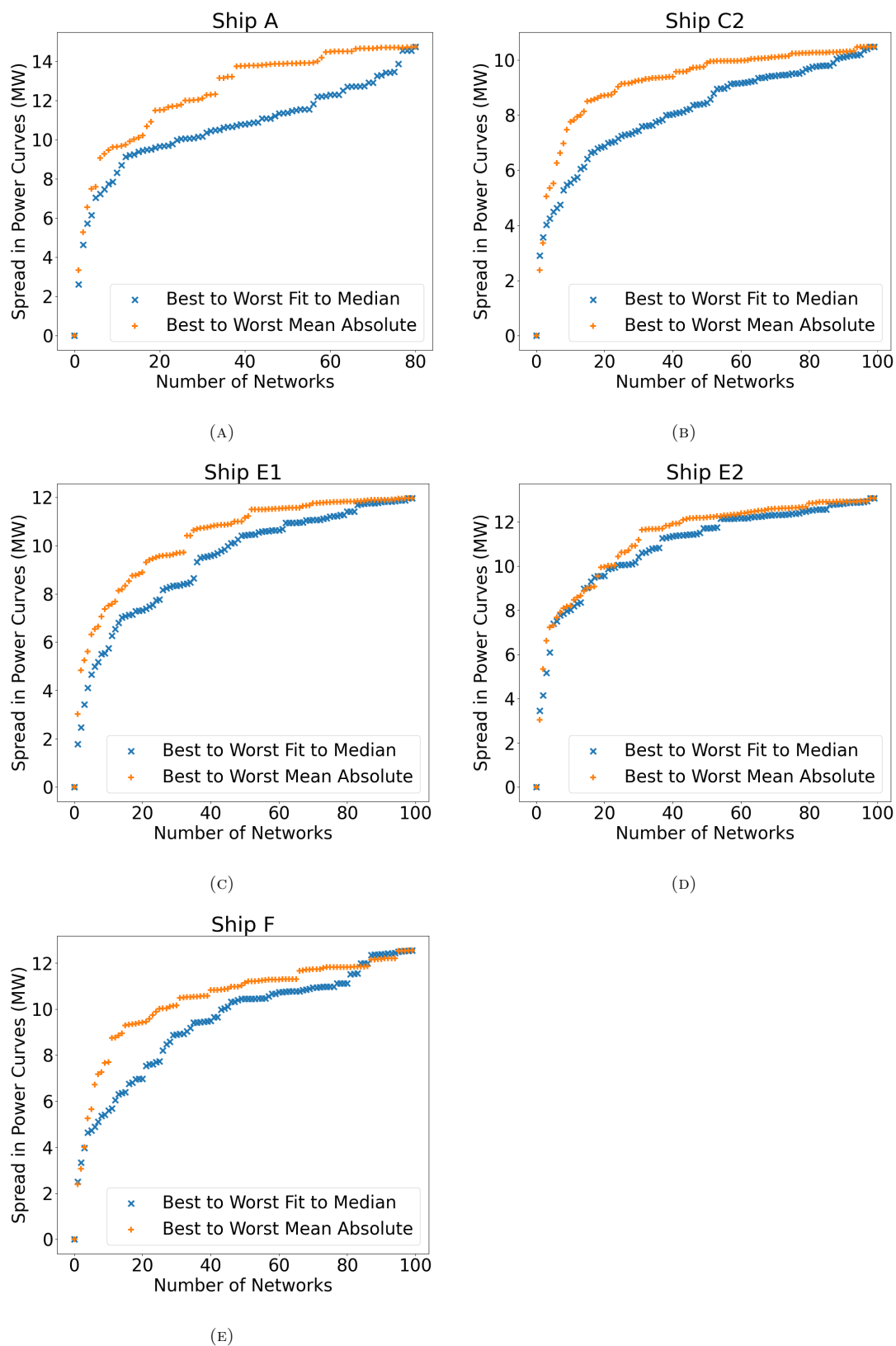


FIGURE 4.20: Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error, averaged over all input-output curves for (A) Ship A, (B) Ship C2, (C) Ship E1, (D) Ship E2, and (E) Ship F.

The average spread of all input variable curves for each ship are also analysed. Only ships A, C2, E1, E2 and F are shown here, the rest are provided in Appendix C. They illustrate that consistency of predictions is always better or no worse for networks ordered by Mean Fit to Median Error compared to Mean Absolute Relative Error, Figure 4.20.

The spread in predictions for Ship E2 shows less than a 1MW improvement for the majority of the curve, Figure 4.20d. Apart from one datapoint at 19 networks, the networks with the lowest Mean Fit to Median Errors are always more consistent than the networks with the lowest Mean Absolute Relative Error. There is no clear reason why the Fit to Median measure does not produce notably more consistent results for Ship E2 in particular; it is a sister ship and its sister, ship E1 Figure 4.20c, shows notably more consistent results with the Fit to Median Error measure.

Some ships show distinct jumps in spread for both Mean Absolute Relative Error and Mean Fit to Median curves, Figures 4.20a, 4.20c and 4.20e. This is suggested to be caused by the introduction of a network which has found a local minima which has not been found by the networks preceding it. This would produce a change in input-output relationship modelled, increasing the spread of predictions for all datapoints proceeding it.

The results from applying the Mean Fit to Median Error measure to the fleet power prediction problem show that more consistent input-output relationships are learnt by networks with low Fit to Median Errors and that these relationships also extrapolate more accurately.

4.7 Summary

As highlighted in previous sections, current error measures for regression can produce networks with low errors on a testing set by mapping arbitrary patterns between the inputs and outputs but how well they fit to the ground truth is unknown. This section derives a new error measure, Mean Fit To Median error, which measures how far from a proxy of the conditional median a network’s predictions are, for applications where no understanding of the input-output relationships exist.

The error measure is compared to the traditional Minkowski-r metrics on 36 different artificial datasets which approximate real datasets and violate the assumptions for a neural networks reporting low Mean Absolute Relative Error to reliably approximate the ground truth of the dataset. The Mean Fit to Median Error is shown to correlate to the Fit to Ground Truth more, with an average increase in R^2 of 0.6 compared to the Mean Absolute Error. Networks with low Mean Fit To Median error model the ground truth of a dataset more reliably than networks with a low Mean Absolute Error; by creating a bias to the ground truth. This allows users to select networks which accurately map the relationships between the inputs and the outputs of a regression problem, even when they do not know what these relationships are.

This method is applied to the ship power prediction application. This verifies that the behaviour on the artificial datasets approximate that seen on the real dataset. It is noted that networks with low Mean Fit to Median Errors approximate the conditional averages closer and produce more consistent input-output relationships. The new error measure is further applied to the problem of power prediction for a vessel without data, this allows more thorough investigation into extrapolation behaviour. The results show that networks with lower Fit to Median Errors extrapolate more accurately and more consistently. There is also minimal requirement to sacrifice Minkowski-r Error values to ensure a good fit to the ground truth, as all networks produced competitive Mean Absolute Relative Error values compared to the literature.

Chapter 5

Discussion

There is an increasing demand for machine learning that can be relied upon for significant decisions, requiring a high level of trust (Voosen 2017). These methods cannot be applied with confidence without a certainty that they have modelled the correct input-output relationships (Hutson 2018). Current methods predict accurately within the ranges of the training data, but extrapolate poorly because they have not modelled the ground truth relationships of the dataset (Willard et al. 2020). A good example of this is the problem of power prediction for large merchant ships in operation. Given the complexity of input-output interactions and the subsequent quantity of domain knowledge known by Naval Architects about vessel propulsion, neural networks are identified as the most promising regression method to solve the problem.

Chapter 3 shows that power can be predicted to 2% error, but the input-output relationships modelled only approximate the ground truth in areas of dense data. It is also demonstrated, for the first time, that prediction for a ship which does not gather data is possible to 4% error. This is achieved through a data fusion of 8 different vessels where the fusion is the combination of the 8 ships, without bias or weighting to any specific ship. Although meaningful feature extraction has not been noted during either single or fleet study, a full investigation into whether it is possible is of interest.

To be usable, predictions from regression methods are required outside of the training ranges. Using ship power prediction as an example, there is no guarantee what conditions a vessel will encounter while in operation. As seen in the neural network results, low error values are achieved by the neural networks but analysis of the input-output relationships modelled show they are inconsistent and illogical. As consistent and correct input-output relationships are required for accurate extrapolation (Sahoo et al. 2018), or prediction in sparse regions of data, this suggests that the current error measures do not assess whether the ground truth is modelled by a regression method.

Chapter 4 develops a new error measure, the Mean Fit to Median, for regression which assesses how close the trained methods' modelled relationships are to the conditional medians of the dataset; the median output value conditioned on each isolated input value in turn. This is used as a proxy for the ground truth relationships, and is shown to be a better approximation than the conditional means. The Mean Fit to Median is validated on 36 different artificial datasets, which resemble simplified versions of regression problems like the ship powering example. Although neural networks are used for the ship powering example, the Fit to Median is developed specifically to be applicable across all regression methods. For example if Gaussian processes were a feasible method to apply to vessel power prediction, the same procedure would be followed: train multiple Gaussian processes, then use the Fit to Median to assess how closely each trained process models the conditional medians of the dataset and choose the process which produces the lowest Fit to Median Error.

In Section 4.4.2 it is noted that a set of networks trained with a Mean Absolute Error loss function produce Mean Absolute Error values in a bimodal distribution. The two modes represent the bifurcations seen in Section 4.4.1, where networks with the same modelling of the ground truth have diverging Mean Absolute Errors. It is difficult to identify the causes of the bifurcations and the bimodal distributions; it is posited that it is to do with the non-Gaussian distribution of the output variable when conditioned on the input. The Mean Absolute Error surface has multiple, similar, minima corresponding to different weight value combinations. If the assumptions discussed in Section 2.2 hold, then the Mean Absolute Error minimum has corresponding weight values which produce the conditional averages for the isolated input-output relationships (Bishop 1995). The violation of the Gaussian noise assumption moves the error minimum away from that which produces the conditional average (Rasmussen and Williams 2006), and introduces more minima which produce predictions with a limited relation to the ground truth. Then the stochastic nature of the neural network training procedure determines which minima is 'chosen' by a network, producing the multi-modal distribution. This is supported by the fact that the bimodal distribution of Mean Absolute Error is more distinct for the ship application, where the output data has a more irregular distribution, and by the fact that the networks with the best ground truth approximation did not correspond to the Mean Absolute Error minima.

Multiple attempts have been made to bias the training of a regression method towards low Fit to Median values without removing the regression procedure entirely but none have proved successful. The first attempt to bias training was to use a combined loss function including both a conventional error measure, such as Mean Absolute Relative, and the Fit to Median, weighted such that the Fit to Median has less effect than the Mean Absolute Relative. The combined loss function consistently produced networks with higher Mean Absolute Relative Error and higher Mean Fit to Median Error. It is suggested this is because the isolated conditional median input-output curves are only relevant when all other inputs are at their overall median value and only

a small quantity of datapoints have all input variables apart from one at their median value, for the ship powering datasets no datapoints have all but one input at its median value. So using the Fit to Median in a loss function biases predictions to arbitrary values, reducing the accuracy of predictions.

The second attempt to bias training was to pre-train the networks on the isolated conditional median input-output relationships (Jaitly et al. 2012), to force the networks towards the local minima which produces low conventional errors and a close modelling of the ground truth. This method produced slightly more consistent Mean Absolute Relative and Mean Fit to Median Errors, but these were not significantly lower than the conventional training approach. The pre-training stage in this approach introduced more parameters to be chosen and it is posited that with more time to fully explore the effect of the parameters and tune them, the results would have shown significant reduction in Mean Fit to Median Errors.

As the use of multi-objective evolutionary computation is state-of-the-art for hyperparameter tuning of neural networks (Yang et al. 2021) (Kumar et al. 2021), it is investigated as a method to find trained networks with low Fit to Median Errors. As the practice is to use the errors from trained networks as the fitness of individuals in genetic algorithms or particle swarm optimisations, the Fit to Median can be used alongside point-based metrics such as Mean Absolute Error as objectives. The preliminary results from this approach for the ship powering application identify a trade-off between the two error values, within the same range of variation of Mean Absolute Error when the Fit to Median is not considered as an objective. This further supports hypotheses from Chapter 4 that low conventional errors do not need to be sacrificed to produce networks with low Fit to Median Errors.

The Fit to Median is used to identify which neural networks have best approximated the ground truth relationships for ship powering prediction in Sections 4.5 and 4.6. It is illustrated that networks producing the lowest Fit to Median Errors model more consistent isolated input-output relationships compared to networks producing the lowest Mean Absolute Relative Errors. This is especially notable in sparse areas of data, and for ‘second order variables’ or inputs with lower correlations to the output. It is suggested this is because the Mean Absolute Relative Error produces networks which map arbitrary relationships; as the input variables which are highly correlated are most important for accurate point prediction the isolated relationship is modelled accurately. Whereas, the relationships for second order variables, which have lower magnitude effect on the output, are not modelled accurately.

For the single ship predictions in Section 4.5, there is no visible difference in accuracy of learnt relationship between the sparse areas of data which are adjacent to dense areas of data and sparse areas which are not adjacent to dense areas. However, for the fleet predictions in Section

4.6, networks show significant difference in predicted relationship accuracy for extrapolated areas further away from the areas of the input domain with data.

5.1 Limitations

It is noted that the Fit to Median Error measure will only work in scenarios where the ground truth is closely approximated by the conditional medians, the output median conditioned on each isolated input variable. The scenarios where this is not the case are, for example:

- (i) inverse problems, where the input-output relationships are not functions;
- (ii) problems with high stochasticity, with high noise or variance, of one or more input variables;
- (iii) and where input-output relationships have large gradients.

The ground truth is not approximated by the conditional medians in the above examples as high stochasticity and large function gradients create a larger ‘Jensen gap’ (Gao et al. 2018), the gap between the ground truth function and the averages of the outputs of the function. The artificial datasets have functions of polynomial degree up to 5 and varying levels of noise. Therefore the Fit to Medians’ effectiveness in a larger range of datasets is of interest. If the ground truth relationship is not a function then the conditional averages will not approximate it.

Another clear drawback in the usage of the Fit to Median Error is the requirement to train a large number—in the order of 100-1,000—of regression methods, significantly increasing the required training time for many methods. Since the Fit to Median Error compares the learnt input-output relationships to a set of static curves which are the averages of the dataset, it is not possible to use this error as a loss function for training a regression method, as it would result in a trained network which produced the averages of the dataset exactly, removing any need for a regression procedure. The use of evolutionary computation bypasses this, if it is already being used to tune network hyperparameters, however for applications where it is not employed training times are increased 1,000-fold.

There is also a need for a full study into the minimum number of training repeats required to produce sufficient spread in predicted relationships that the Fit to Median Error can identify a preferable trained method. This is likely to depend on: the method used, the application, the size of dataset and the stochasticity of the training approach.

5.2 Future Work

For power prediction of a vessel without recorded data the future work includes: extending the method to the state of the art for data fusion techniques, identifying if feature extraction is possible (Addison et al. 2003), and expanding the variety and quantity of vessels in a fleet. Statistical hierarchical methods to encourage feature extraction include utilising cascade networks (Du et al. 2019) or training each layer of a network on a different ship, among others (Addison et al. 2003). The latter would provide flexibility for more diverse fleets. It would allow a modular approach to network layers allowing only the most relevant vessels to be used for prediction, identified by ship specifics. This might improve the methods ability to extrapolate outside the range of observed ship types.

Similar data fusion approaches for wind speed predictions use a clustering of data sources (Tasnim et al. 2018), this could be used to group ships with similar powering relationships. For prediction for a given ship, a weighting is assigned to each cluster in relation to its relevance to the given ship, this could similarly improve prediction for extrapolated ships. However, there is limited gain from using clustering or separately trained layers on the fleet used in Section 3.5, due to its small size and limited variation in ship parameters. Therefore, the use of a larger, more diverse fleet is required to fully explore the potential of this methodology.

For the regression error measure investigation the future work includes: further exploration into the cause of the bifurcations noted in Section 4.4.2, investigation into the effect of the Fit to Median on an increased range of type of dataset, and removing the need to train multiple methods to identify one with a low Fit to Median Error. Verification that the driver of the location of Minkowski-r error minima is the violation of the Gaussian assumption would allow loss functions to be created which produce methods providing better approximations of the ground truth. These methods could be applied during traditional training of a regression method, and therefore would not require excess compute.

An empirical study into the range of types of dataset which benefit from the use of the Fit to Median Error is of interest. The Fit to Median provided improved modelling of the ground truth for all 36 datasets investigated. It is suggested that datasets with high stochasticity and datasets where input-output relationships have large gradients will not benefit from the use of the Fit to Median. Verification of this is important, as well as an understanding of at which level of stochasticity, or gradient, the Fit to Median loses usefulness. Investigation into whether the Fit to Median produces worse predictions for certain types of dataset is also salient, for example the Fit to Median has not been trialled on an inverse problem.

As it is noted that conventional regression error measures produce accurate input-output relationships within areas of dense data, a multiscale method (Bataineh and Marler 2017) is suggested

which merges piecewise approximations and full scale convolutional approximations of the relationships (Wang et al. 2016). The full scale approach is required to identify which areas have sufficiently dense data, with homoscedastic noise, so conventional error measures can be used to model these areas. Then the Mean Fit to Median can be used for the areas of sparse data, with heteroscedastic noise, to increase consistency and accuracy of ground truth estimation in the areas where it is hardest to approximate it. This approach should allow the Mean Fit to Median to be applicable to a larger range of regression problems, as it will only be applied to areas of the training domain where it is required.

For the application of the Fit to Median to the fleet power predictions the future work includes investigation into how the density and distribution of data affects ground truth approximations. For networks reporting low Mean Absolute Errors the spread in predictions increases for increasing distance from the dense area of data but this is not apparent for networks reporting low Fit to Median Error. Exploration into if the use of fat tailed distributions, producing small regions of dense data and large regions of sparse data, improves the prediction accuracy in the sparse regions or if the ‘law of small numbers’ produces misleading predictions is of interest.

Chapter 6

Conclusion

To apply machine learning regression with confidence there needs to be a certainty that it has modelled the correct input-output relationships. Current methods predict accurately within the ranges of the training data, but extrapolate poorly because they have not modelled the ground truth relationships of the dataset. This problem is exacerbated by the use of conventional error measures, from the Minkowski-r family, which only require accurate point estimation to assign low error to a method. Minkowski-r error measures will ensure accurate modelling of the average output conditioned on the inputs, a proxy for the ground truth in many applications, only if restrictive dataset assumptions are met. One of these assumptions is sufficient data across the prediction domain, this assumption will never be satisfied if extrapolated predictions are required.

An example of a scenario where extrapolated predictions, or predictions in sparse regions are of interest is merchant vessel power prediction. This thesis shows that power can be predicted to 2% Mean Absolute Relative Error, and have been applied to over a dozen merchant vessels operated by Shell Shipping and Maritime resulting in significant emissions savings. For the first time it is shown that prediction for a ship which does not gather data is possible to 4% Mean Absolute Relative Error. However, these methods fail to model the isolated input-output relationships consistently which is especially notable in areas of sparse data, or extrapolated regions. This demonstrates the downside of ‘black-box’ machine learning methods, that produce low errors but model arbitrary patterns rather than the ground truth. The lack of consistency in predicted relationships reduces the trust a user has for a regression method.

To overcome this, a novel error measure is derived, called the Mean Fit to Median. This determines how close the input-output relationships modelled by a network are to the conditional averages of the dataset. The error measure is verified on 36 different artificial datasets and has a correlation to the ground truth 60% higher on average than traditional error measures. This error measure is then applied to the ship power prediction problem. Networks with low Mean

Fit to Median errors model input-output relationships closer to the conditional averages; they predict the relationships more consistently and this improvement is noted particularly for sparse areas of data without sacrificing traditional accuracy measures. Networks with low Mean Fit to Median also extrapolate more accurately and more consistently than networks with low Mean Absolute Relative Errors. Overall, the new Mean Fit to Median Error measure identifies more trustworthy and interpretable networks than traditional error measures.

References

- Abramowski, T., T. Cepowski, and P. Zvolensky (2018). “Determination of regression formulas for key design characteristics of container ships at preliminary design stage”. In: *New Trends in Production Engineering* 1.1, pp. 247–257.
- Addison, D., S. Wermter, and G. Arevian (2003). “A comparison of feature extraction and selection techniques”. In: *Proceedings of the International Conference on Artificial Neural Networks*, pp. 212–215.
- Aizerman, M. A. (1964). “Theoretical foundations of the potential function method in pattern recognition learning”. In: *Automation and Remote Control* 25, pp. 821–837.
- Albertelli, R. (2020). “Benchmarking Automated Design Methods for Regression Neural Networks Using Ship Propulsion Data”. In: *Final Year Project, University of Southampton*.
- Aldous, L. et al. (2015). “Uncertainty analysis in ship performance monitoring”. In: *Ocean Engineering* 110, pp. 29–38.
- Alston, J. (2019). “Investigating how machine learning can predict the rate of ship fouling”. MA thesis. University of Southampton.
- Álvarez, M., D. Luengo, and N. D. Lawrence (2009). “Latent force models”. In: *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*. Vol. 5, pp. 9–16.
- Anderson, B., T. S. Hy, and R. Kondor (2019). “Cormorant: covariant molecular neural networks”. In: *Advances in Neural Information Processing Systems*. Vol. 32, pp. 14537–14546.
- Aronszajn, N. (1950). “Theory of reproducing kernels”. In: *Transactions of the American Mathematical Society* 68.3, pp. 337–404.
- Bal Beşikçi, E. et al. (2016). “An artificial neural network based decision support system for energy efficient ship operations”. In: *Computers & Operations Research*, pp. 393–401. ISSN: 03050548. DOI: 10.1016/j.cor.2015.04.004.
- Bataineh, M. and T. Marler (2017). “Neural network for regression problems with reduced training sets”. In: *Neural networks* 95, pp. 1–9.
- Battaglia, P. W. et al. (2018). *Relational inductive biases, deep learning, and graph networks*. arXiv: 1806.01261 [cs.LG].
- Berger, J. O. and J. M. Bernardo (1992). *On the development of reference priors*.

- Bernardo, J. M. (1979). "Reference posterior distributions for Bayesian inference". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 41.2, pp. 113–128.
- Bialystocki, N. and D. Konovessis (2016). "On the estimation of ship's fuel consumption and speed curve: a statistical approach". In: *Journal of Ocean Engineering and Science* 1.2, pp. 157–166.
- Bishop, C. (1995). "Neural Networks for Pattern Recognition". In: Oxford University Press. Chap. 6, pp. 194–225.
- Blomqvist, R. (2016). "Determining power increases of a vessel in weather through the analysis of full scale data". In: *Masters Project*.
- Boser, B. E., I. M. Guyon, and V. N. Vapnik (1992). "A training algorithm for optimal margin classifiers". In: *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pp. 144–152.
- Botchkarev, A. (2018). "Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology". In: *arXiv preprint arXiv:1809.03006*.
- Bratu, M. (2013). "New accuracy measures for point and interval forecasts: A case study for Romania's forecasts of inflation and unemployment rate". In: *Atlantic Review of Economics* 1.
- Broomhead, D. S. and D. Lowe (1988). "Multivariable functional interpolation and adaptive networks". In: *Complex Systems*, pp. 321–355.
- Burges, C. J. et al. (1996). "Simplified support vector decision rules". In: *International Conference on Machine Learning (ICML)*. Vol. 96, pp. 71–77.
- Burges, C. J. (1998). "A tutorial on support vector machines for pattern recognition". In: *Data Mining and Knowledge Discovery* 2.2, pp. 121–167.
- Cha, S. H. (2007). "Comprehensive survey on distance/similarity measures between probability density functions". In: *City* 1.2, p. 1.
- Chen, R. T. et al. (2018). "Neural ordinary differential equations". In: *Advances in Neural Information Processing Systems*, pp. 6571–6583.
- Chollet, F. et al. (2015). *Keras*. <https://keras.io>.
- Coraddu, A. et al. (2017). "Vessels fuel consumption forecast and trim optimisation: a data analytics perspective". In: *Ocean Engineering* 130, pp. 351–370.
- Cortes, C. and V. Vapnik (1995). "Support-vector networks". In: *Machine Learning* 20.3, pp. 273–297.
- Cybenko, G. (1989). "Approximation by superpositions of a sigmoidal function". In: *Mathematics of Control Signals Systems* 2, pp. 303–314. ISSN: 10009221. DOI: 10.1007/BF02836480. URL: <https://link.springer.com/content/pdf/10.1007%5C%2F02836480.pdf>.
- De Gooijer, J. G. and R. J. Hyndman (2006). "25 years of time series forecasting". In: *International Journal of Forecasting* 22.3, pp. 443–473.

- Dozat, T. (2016). “Incorporating nesterov momentum into adam”. In: *International Conference on Learning Representations 2016*.
- Drucker, H. et al. (1996). “Support vector regression machines”. In: *Advances in Neural Information Processing systems* 9, pp. 155–161.
- Du, X., K. Farrahi, and M. Niranjani (2019). “Transfer learning across human activities using a cascade neural network architecture”. In: *Proceedings of the 23rd international symposium on wearable computers*, pp. 35–44.
- Dua, D. and C. Graff (2017). *UCI Machine Learning Repository*. URL: <http://archive.ics.uci.edu/ml>.
- Duchi, J., E. Hazan, and Y. Singer (2011). “Adaptive subgradient methods for online learning and stochastic optimization”. In: *Journal of Machine Learning Research* 12.61, pp. 2121–2159.
- Elmas, C. and Y. Sonmez (2011). “A data fusion framework with novel hybrid algorithm for multi-agent decision support system for forest fire”. In: *Expert Systems with Applications* 38, pp. 9225–9236.
- Fan, Y. and L. Ying (2020). “Solving electrical impedance tomography with deep learning”. In: *Journal of Computational Physics* 404, p. 109119.
- Fildes, R. and P. Goodwin (2007). “Against your better judgment? How organizations can improve their use of management judgment in forecasting”. In: *Interfaces* 37.6, pp. 570–576.
- Forrester, A., A. Sobester, and A. Keane (2008). *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons.
- Gal, Y. and Z. Ghahramani (2016). “Dropout as a bayesian approximation: representing model uncertainty in deep learning”. In: *International Conference on Machine Learning, New York, USA*.
- Gao, X., M. Sitharam, and A. Roitberg (2018). “Bounds on the Jensen gap, and implications for mean-concentrated distributions”. In: *arXiv:1712.05267v4*.
- Gauss, C. F. and C. H. Davis (1857). *Theory of the Motion of the Heavenly Bodies Moving about the Sun in Conic Sections: A Translation of Gauss’s “Theoria Motus” with an Appendix*. Little, Brown.
- Glorot, X. and Y. Bengio (2010). “Understanding the difficulty of training deep feedforward neural networks”. In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings*, pp. 249–256.
- Grabowska, K. and P. Szczuko (2015). “Ship resistance prediction with artificial neural networks”. In: *2015 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, pp. 168–173.
- Grigsby, M. R. et al. (2018). “Novel metrics for growth model selection”. In: *Emerging Themes in Epidemiology* 15.1, p. 4.

- Ai-guo, C. and Y. Jia-wei (2009). “Research on the genetic neural network for the computation of ship resistance”. In: *2009 International Conference on Computational Intelligence and Natural Computing*. Vol. 1, pp. 366–369.
- Hanson, S. and D. Burr (1987). “Minkowski-r back-propagation: learning in connectionist models with non-euclidian error signals.” In: *Neural Information Processing Systems (NIPS 1987)*.
- Harris, C. R., K. J. Millman, and S. J. Van der Walt (2020). “Array programming with NumPy”. In: *Nature* 585.7825, pp. 357–362.
- Hinton, G., N. Srivastava, and K. Swersky (2012). *Lecture 6a: Overview of mini-batch gradient descent*. URL: http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf.
- Hofmann, T., B. Schölkopf, and A. J. Smola (2008). “Kernel methods in machine learning”. In: *The annals of statistics*, pp. 1171–1220.
- Holtrop, J. (1984). “A statistical re-analysis of resistance and propulsion data”. In: *International shipbuilding progress* 31.363, pp. 272–276. ISSN: 0020-868X. DOI: 10.1007/s12011-015-0572-4.
- Holtrop, J. and G. G. Mennen (1982). “An approximate power prediction method”. In: *Int. Shipbuilding Progress* 29.
- Hu, Z. et al. (2019). “Prediction of fuel consumption for enroute ship based on machine learning”. In: *IEEE Access* 7, pp. 119497–119505.
- Hutson, M. (2018). *Artificial intelligence faces reproducibility crisis*.
- Hyndman, R. J. and A. B. Koehler (2006). “Another look at measures of forecast accuracy”. In: *International Journal of Forecasting* 22.4, pp. 679–688.
- International Maritime Organisation (2020). *Fourth IMO Greenhouse Gas Study*.
- International Maritime Organization (2018). *Marine Environment Protection Committee (MEPC), 72st session*.
- ISO 19030 (2016). “International standard: ship and marine technology–measurement of changes in hull and propeller performance (ISO 19030)”. In: 1.
- Jaitly, N. et al. (2012). “Application of pre-trained deep neural networks to large vocabulary speech recognition”. In: *Proceedings of Interspeech 2012*.
- Jensen, J. (1906). “Sur les fonctions convexes et les inégalités entre les valeurs moyennes”. In: *Acta Math.* 30, pp. 175–193.
- Jeon, M. et al. (2018). “Prediction of ship fuel consumption by using an artificial neural network”. In: *Journal of Mechanical Science and Technology* 32.12, pp. 5785–5796.
- Kanagawa, M. et al. (2018). “Gaussian processes and kernel methods: a review on connections and equivalences”. In: *arXiv:1807.02582*.
- Karpatne, A. et al. (2018). “Physics-guided neural networks (PGNN): an application in lake temperature modelling”. In: *arXiv:1710.11431v2*.

- Kim, D., S. Lee, and J. Lee (2020). “Data-Driven prediction of vessel propulsion power using support vector regression with onboard measurement and ocean data”. In: *Sensors* 20.6, p. 1588.
- Kim, S. and H. Kim (2016). “A new metric of absolute percentage error for intermittent demand forecasts”. In: *International Journal of Forecasting* 32.3, pp. 669–679.
- Kim, S. H. and F. Boukouvala (2019). “Machine learning-based surrogate modelling for data-driven optimization: a comparison of subset selection for regression techniques”. In: *Optimization Letters*, pp. 1–22.
- Kingma, D. P. and J. Ba (2014). “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980*.
- Kristensen, H. O. and M. Lützen (2012). “Prediction of resistance and propulsion power of ships”. In: *Clean Shipping Currents* 1.6, pp. 1–52.
- Kullback, S. and R. A. Leibler (1951). “On information and sufficiency”. In: *The Annals of Mathematical Statistics* 22.1, pp. 79–86.
- Kumar, P., S. Batra, and B. Raman (2021). “Deep neural network hyper-parameter tuning through twofold genetic approach”. In: *Soft Computing*, pp. 1–25.
- Kyriakidis, I. et al. (2015). “New statistical indices for evaluating model forecasting performance”. In: *Skiathos Island, Greece*.
- Lakshminarayanan, P. and D. A. Hudson (2017). “Estimating added power in waves for ships through analysis of operational data”. In: *2nd Hull Performance and Insight Conference: HullPIC’17*.
- Lampinen, J. and A. Vehtari (2001). “Bayesian approach for neural networks—review and case studies”. In: *Neural networks* 14.3, pp. 257–274.
- Le, L. et al. (2020). “Neural network-based fuel consumption estimation for container ships in Korea”. In: *Maritime Policy & Management*, pp. 1–18.
- Lee, J. et al. (2017). “Deep neural networks as Gaussian processes”. In: *arXiv:1711.00165*.
- Lee, K. and K. T. Carlberg (2020). “Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders”. In: *Journal of Computational Physics* 404, p. 108973.
- Legendre, A. M. (1805). *Nouvelles méthodes pour la détermination des orbites des comètes*. Firmin Didot, Paris.
- Leifsson, L. et al. (2008). “Grey-box modeling of an ocean vessel for operational optimization”. In: *Simulation Modelling Practice and Theory* 16.8, pp. 923–932.
- Leshno, M. et al. (1993). “Multilayer feedforward networks with a nonpolynomial activation function can approximate any function”. In: *Neural Networks* 6.6, pp. 861–867. ISSN: 08936080. DOI: 10.1016/S0893-6080(05)80131-5. URL: <https://pdfs.semanticscholar.org/ae37/354fc5138bf6ae683cf3fba014571d4540c3.pdf>.

- Liang, Q., H. A. Tvette, and H. W. Brinks (2019). “Prediction of vessel propulsion power using machine learning on AIS data, ship performance measurements and weather data”. In: *Journal of Physics: Conference Series*. Vol. 1357. 1, p. 012038.
- Liu, D. and Y. Wang (2019). “Multi-fidelity physics-constrained neural network and its application in materials modeling”. In: *Journal of Mechanical Design* 141.12.
- Lucia, D. J., P. S. Beran, and W. A. Silva (2004). “Reduced-order modeling: new approaches for computational physics”. In: *Progress in Aerospace Sciences* 40.1-2, pp. 51–117.
- MacGillivray, H. (1981). “The mean, median, mode inequality and skewness for a class of densities”. In: *Australian Journal of Statistics* 23, pp. 247–250.
- MacKay, D. J. (1992). “A practical Bayesian framework for backpropagation networks”. In: *Neural Computation* 4.3, pp. 448–472.
- Mahalanobis, P. C. (1936). “On the generalised distance in statistics”. In: *Proceedings of the National Institute of Science of India* 12, pp. 49–55.
- Maier, H. R. and G. C. Dandy (1998). *The effect of internal parameters and geometry on the performance of back-propagation neural networks: An empirical study*. DOI: 10.1016/S1364-8152(98)00020-6. URL: <http://www.neuralware.com>.
- Mardt, A. et al. (2018). “VAMPnets for deep learning of molecular kinetics”. In: *Nature Communications* 9.1, pp. 1–11.
- Marquardt, D. W. (1970). “Generalized inverses, ridge regression, biased linear estimation, and nonlinear estimation”. In: *Technometrics* 12.3, pp. 591–612.
- Matthews, A. G. et al. (2018). “Gaussian process behaviour in wide deep neural networks”. In: *arXiv preprint arXiv:1804.11271*.
- McCarthy, T. M et al. (2006). “The evolution of sales forecasting management: a 20-year longitudinal study of forecasting practices”. In: *Journal of Forecasting* 25.5, pp. 303–324.
- McCulloch, W. S. and W. Pitts (1943). “A logical calculus of the ideas immanent in nervous activity”. In: *The bulletin of mathematical biophysics* 5.4, pp. 115–133.
- Melendez-Pastor, C., R. Ruis-Gonzalez, and J. Gomez-Gil (2017). “A data fusion system of GNSS data and on-vehicle sensors data for improving car positioning precision in urban environments”. In: *Expert Systems with Applications* 80, pp. 28–38.
- Merkle, M. (2005). “Jensen’s inequality for medians”. In: *Statistics and Probability Letters* 71, pp. 277–281.
- Merwe, R. van der et al. (2007). “Fast neural network surrogates for very high dimensional physics-based models in computational oceanography”. In: *Neural Networks* 20.4, pp. 462–478. ISSN: 08936080. DOI: 10.1016/j.neunet.2007.04.023. URL: www.elsevier.com/locate/neunet.
- Minsky, M. and S. Papert (1969). “An introduction to computational geometry”. In: *Cambridge, HIT*.

- Molland, A. F., S. R. Turnock, and D. A. Hudson (2011). *Ship resistance and propulsion*. ISBN: 9780511974113. DOI: 10.1017/CB09780511974113. URL: <http://ebooks.cambridge.org/ref/id/CB09780511974113>.
- Møller, M. F. (1993). “A scaled conjugate gradient algorithm for fast supervised learning”. In: *Neural networks* 6.4, pp. 525–533.
- Mulgrew, B. (1996). “Applying radial basis functions”. In: *IEEE Signal Processing Magazine* 13.2, pp. 50–65.
- Nelder, J. A. and R. W. M. Wedderburn (1972). “Generalized linear models”. In: *Journal of the Royal Statistical Society. Series A (General)* 135.3, pp. 370–384. ISSN: 00359238.
- NOAA (2017). *NOAA Adds Wave, Visibility Data to PORTS® Navigational Data System*. URL: <https://oceanservice.noaa.gov/news/weeklynews/nov10/ports-news.html> (visited on 03/07/2019).
- Orloff, J. and J. Bloom (2014). “Comparison of frequentist and Bayesian inference”. In: *Massachusetts Institute of Technology*.
- Panapakidis, I., V. M. Sourtzi, and A. Dagoumas (2020). “Forecasting the fuel consumption of passenger ships with a combination of shallow and deep learning”. In: *Electronics* 9.5, p. 776.
- Park, J. and J. Park (2019). “Physics-induced graph neural network: an application to wind-farm power estimation”. In: *Energy* 187, p. 115883.
- Parkes, A. I., A. J. Sobey, and D. A. Hudson (2018). “Physics-based shaft power prediction for large merchant ships using neural networks”. In: *Ocean Engineering* 166, pp. 92–104.
- Parkes, A. I. et al. (2019). “Efficient vessel power prediction in operational conditions using machine learning.” In: *Practical Design of Ships and Other Floating Structures (PRADS), September 2019, Yokohama, Japan*.
- Parri, M. (2017). “Physics based learning for wind turbines”. MA thesis. University of Southampton and University of Perugia.
- Pedersen, B. P. and J. Larsen (2009). “Prediction of full-scale propulsion power using artificial neural networks”. In: *Proceedings of the 8th international conference on computer and IT applications in the maritime industries (COMPIT’09), Budapest, Hungary May*, pp. 10–12.
- Petersen, J. P., D. J. Jacobsen, and O. Winther (2012). “Statistical modelling for ship propulsion efficiency”. In: *Journal of marine science and technology* 17.1, pp. 30–39.
- Prestwich, S. et al. (2014). “Mean-based error measures for intermittent demand forecasting”. In: *International Journal of Production Research* 52.22, pp. 6782–6791.
- Price, R. (1763). “An essay towards solving a problem in the doctrine of chances”. In: *Philosophical Transactions of the Royal Society* 53, pp. 370–418.
- Pukrittayakamee, A. et al. (2009). “Simultaneous fitting of a potential-energy surface and its corresponding force fields using feedforward neural networks”. In: *The Journal of Chemical Physics* 130.13, p. 134101.

- Radonjic, A. and K. Vukadinovic (2015). “Application of ensemble neural networks to prediction of towboat shaft power”. In: *Journal of Marine Science and Technology (Japan)* 20.1, pp. 64–80. ISSN: 09484280. DOI: 10.1007/s00773-014-0273-2.
- Raissi M. and Wang, Z., M. S. Triantafyllou, and G. E. Karniadakis (2019). “Deep learning of vortex-induced vibrations”. In: *Journal of Fluid Mechanics* 861, pp. 119–137.
- Rajakarunakaran, S. et al. (2008). “Artificial neural network approach for fault detection in rotary system”. In: *Applied Soft Computing* 8.1, pp. 740–748. ISSN: 15684946. DOI: 10.1016/j.asoc.2007.06.002. URL: <http://linkinghub.elsevier.com/retrieve/pii/S1568494607000580>.
- Ramcharan, R. (2006). “Regressions: Why Are Economists Obsessed with Them?” In: *International Monetary Fund* 43 (1).
- Rasmussen, C. E. and C. K. I. Williams (2006). *Gaussian Processes for machine learning*. The MIT Press.
- Read, J. S. et al. (2019). “Process-guided deep learning predictions of lake water temperature”. In: *Water Resources Research* 55.11, pp. 9173–9190.
- Reddi, S. J., S. Kale, and S. Kumar (2019). “On the convergence of adam and beyond”. In: *arXiv preprint arXiv:1904.09237*.
- Rosenblatt, F. (1957). *The perceptron, a perceiving and recognizing automaton*. Cornell Aeronautical Laboratory.
- Rumelhart, D. E., G. E. Hinton, and R. J. Williams (1985). *Learning internal representations by error propagation*. Tech. rep. California Univ San Diego La Jolla Inst for Cognitive Science.
- Ruthotto, L. and E. Haber (2019). “Deep neural networks motivated by partial differential equations”. In: *Journal of Mathematical Imaging and Vision*, pp. 1–13.
- Sadowski, P., N. Fooshee D. and Subrahmanya, and P. Baldi (2016). “Synergies between quantum mechanics and machine learning in reaction prediction”. In: *Journal of Chemical Information and Modeling* 56.11, pp. 2125–2128.
- Sahoo, S., C. Lampert, and G. Martius (2018). “Learning equations for extrapolation and control”. In: *International Conference on Machine Learning*. PMLR, pp. 4442–4450.
- Simsir, U. and S. Ertugrul (2009). “Prediction of manually controlled vessels’ position and course navigating in narrow waterways using Artificial Neural Networks”. In: *Applied Soft Computing* 9.4, pp. 1217–1224.
- Smola, A. J. and B. Schölkopf (2003). “Bayesian kernel methods”. In: *Advanced Lectures on Machine Learning*, pp. 65–117.
- Solonen, A. et al. (2020). *Hierarchical Bayesian propulsion power models for marine vessels*. arXiv: 2004.11267.
- Sultan, M. M., H. K. Wayment-Steele, and V. S. Pande (2018). “Transferable neural networks for enhanced sampling of protein dynamics”. In: *Journal of chemical theory and computation* 14.4, pp. 1887–1894.

- Swischuk, R. et al. (2019). “Projection-based model reduction: Formulations for physics-based machine learning”. In: *Computers & Fluids* 179, pp. 704–717.
- Tasneem, F. (2019). *Study the effects of approximation on conjugate gradient algorithm and accelerate it on FPGA platform*. URL: <https://cs.uni-paderborn.de/fileadmin/informatik/fg/hit/research/hit-seminar/2018-04-11-Filmwala-HIT-Seminar.pdf> (visited on 03/06/2019).
- Tasnim, S. et al. (2018). “Wind power prediction in new stations based on knowledge of existing Stations: A cluster based multi source domain adaptation approach”. In: *Knowledge-Based Systems* 145, pp. 15–24.
- Taylor, R. (1990). “Interpretation of the correlation coefficient: a basic review”. In: *Journal of Diagnostic Medical Sonography* 6.1, pp. 35–39.
- Tibshirani, R. (1996). “Regression shrinkage and selection via the lasso”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1, pp. 267–288.
- Townsin, R. L. et al. (1993). “Estimating the influence of weather on ship performance.” In: *Transaction of the Royal Institution of Naval Architects* 135, pp. 191–209.
- Tversky, A. and D. Kahneman (1971). “Belief in the law of small numbers”. In: *Psychological Bulletin* 76 (2), pp. 105–110.
- Uyanik, T., C. Karatug, and Y. Arslanoğlu (2020). “Machine learning approach to ship fuel consumption: a case of container vessel”. In: *Transportation Research Part D: Transport and Environment* 84, p. 102389. ISSN: 1361-9209.
- Vapnik, V. (1998). *Statistical learning theory*. Wiley, New York.
- Voosen, P. (2017). *The AI detectives*. American Association for the Advancement of Science.
- Wackers, J. et al. (2011). “Free-surface viscous flow solution methods for ship hydrodynamics”. In: *Archives of Computational Methods in Engineering* 18.1, pp. 1–41.
- Wang, H. et al. (2017). “Deep learning based ensemble approach for probabilistic wind power forecasting”. In: *Applied Energy* 188, pp. 56–70. ISSN: 03062619. DOI: 10.1016/j.apenergy.2016.11.111.
- Wang, J. et al. (2016). “A multi-scale convolution neural network for featureless fault diagnosis”. In: *2016 International Symposium on Flexible Automation (ISFA)*, pp. 65–70.
- Wang, S. et al. (2018). “Predicting ship fuel consumption based on LASSO regression”. In: *Transportation Research Part D: Transport and Environment* 65, pp. 817–824.
- Weymouth, G. D. and D. K. P. Yue (2014). “Physics-Based Learning Models for Ship Hydrodynamics”. In: *Journal of Ship Research* 57.1, pp. 1–12. DOI: 10.5957/JOSR.57.1.120005. arXiv: 1401.3816. URL: <http://arxiv.org/abs/1401.3816> <http://dx.doi.org/10.5957/JOSR.57.1.120005>.
- Willard, J. et al. (2020). *Integrating physics-based modeling with machine learning: a survey*. arXiv: 2003.04919.

- Yang, S. et al. (2021). “A gradient-guided evolutionary approach to training deep neural networks”. In: *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15. DOI: 10.1109/TNNLS.2021.3061630.
- Yao, K. et al. (2018). “The TensorMol-0.1 model chemistry: a neural network augmented with long-range physics”. In: *Chemical Science* 9.8, pp. 2261–2269.
- Yoo, B. and J. Kim (2019). “Probabilistic modelling of ship powering performance using full-scale operational data”. In: *Applied Ocean Research* 82, pp. 1–9.
- Yuan, J. and V. Nian (2018). “Ship energy consumption prediction with Gaussian process meta-model”. In: *Energy Procedia* 152, pp. 655–660.
- Zeiler, M. D. (2012). *ADADELTA: an adaptive learning rate method*. arXiv: 1212.5701.
- Zhang, L. et al. (2018). “End-to-end symmetry preserving inter-atomic potential energy model for finite and extended systems”. In: *Advances in Neural Information Processing Systems*. Vol. 31, pp. 4436–4446.
- Zhu, J. and T. Hastie (2005). “Kernel logistic regression and the import vector machine”. In: *Journal of Computational and Graphical Statistics* 14.1, pp. 185–205.

Appendix A

Supplementary Figures for Section 3.5

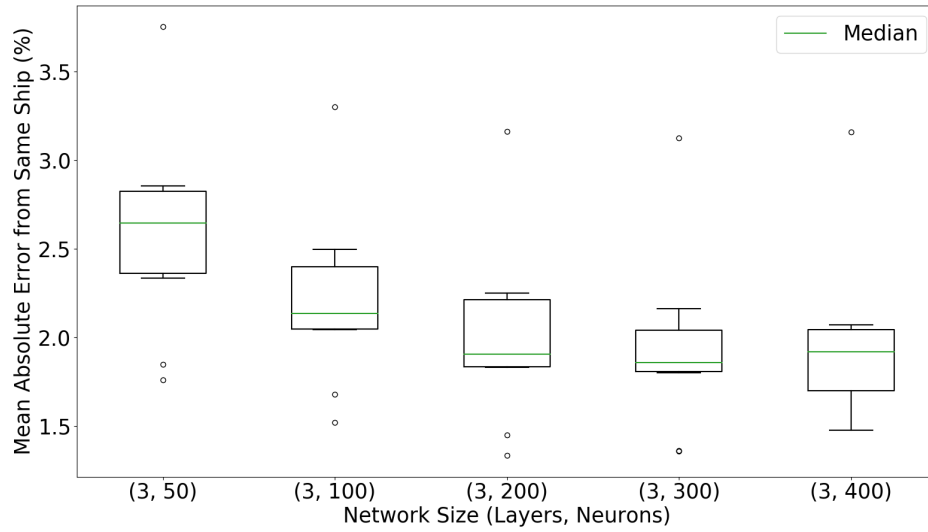


FIGURE A.1: Boxplots showing the distribution of Mean Absolute Relative Error of networks trained and tested on data from the same ship, for all ships, for varying network sizes.

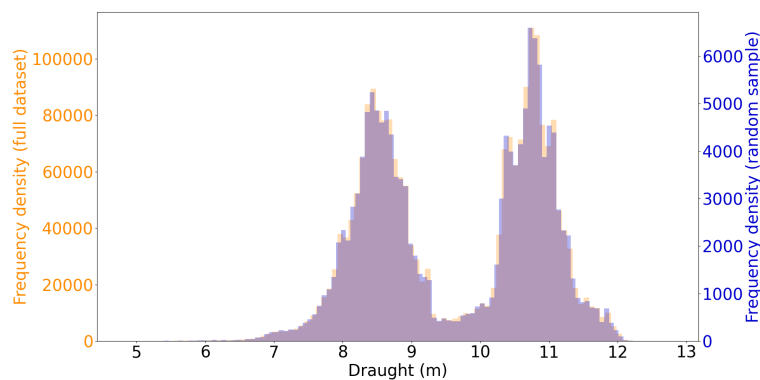


FIGURE A.2: Histograms of draught from full dataset and random sample of size 150,000, for ship C1.

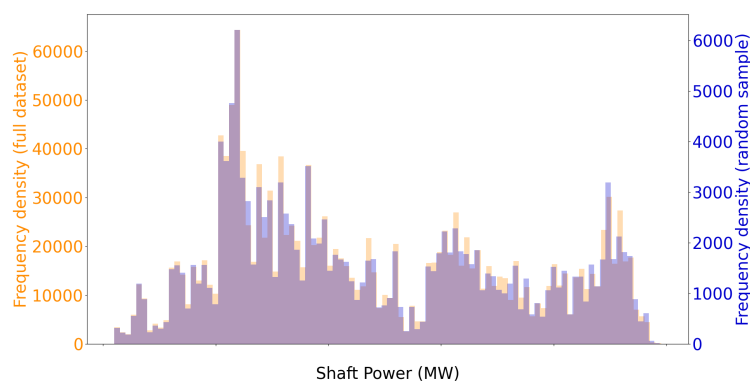


FIGURE A.3: Histograms of shaft power from full dataset and random sample of size 150,000, for ship C2.

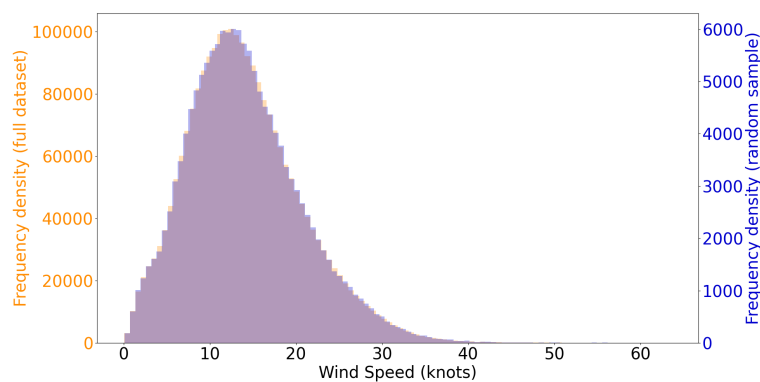


FIGURE A.4: Histograms of wind speed from full dataset and random sample of size 150,000, for ship C4.

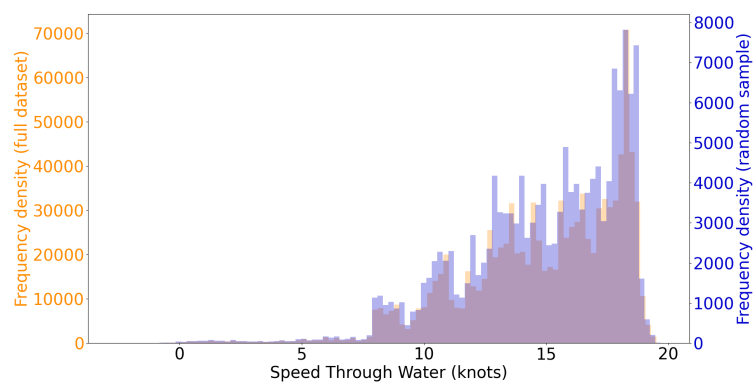


FIGURE A.5: Histograms of vessel speed from full dataset and random sample of size 150,000, for ship D1.

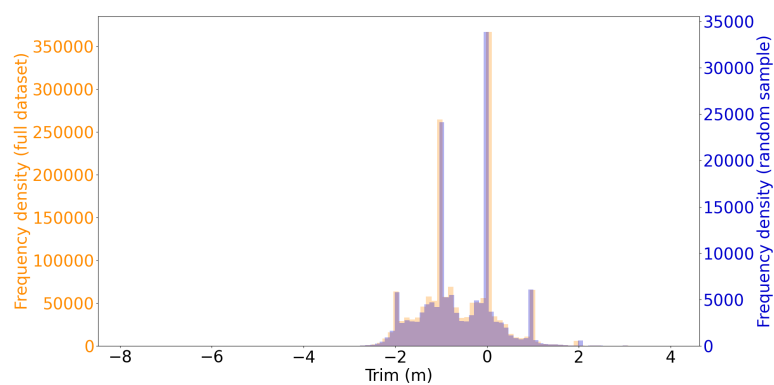


FIGURE A.6: Histograms of trim from full dataset and random sample of size 150,000, for ship D2.

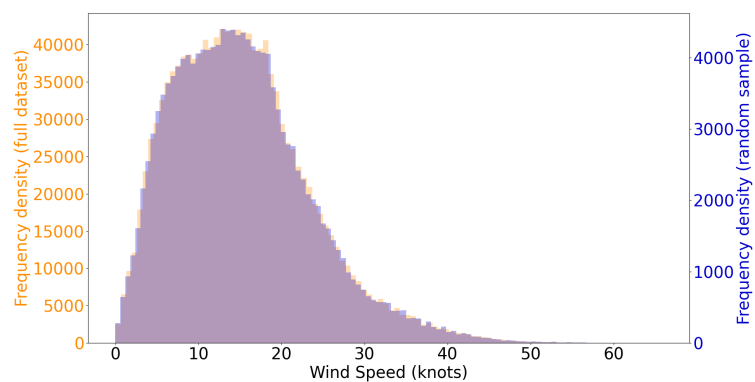


FIGURE A.7: Histograms of wind speed from full dataset and random sample of size 150,000, for ship E1.

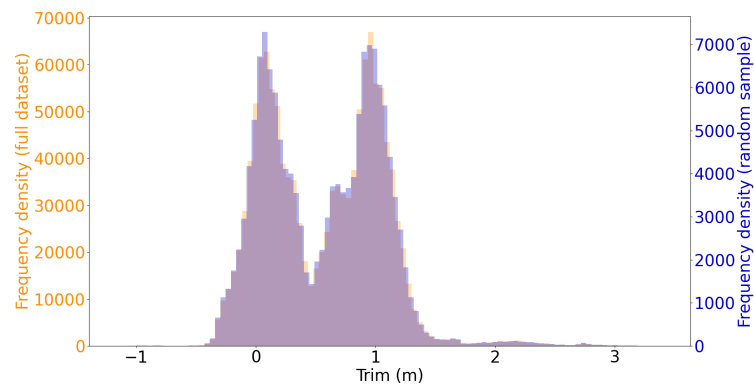


FIGURE A.8: Histograms of trim from full dataset and random sample of size 150,000, for ship E2.

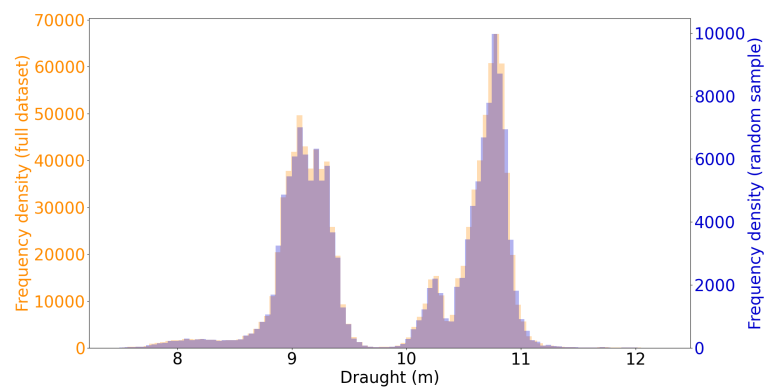


FIGURE A.9: Histograms of draught from full dataset and random sample of size 150,000, for ship F.

Appendix B

Supplementary Figures for Section 4.5

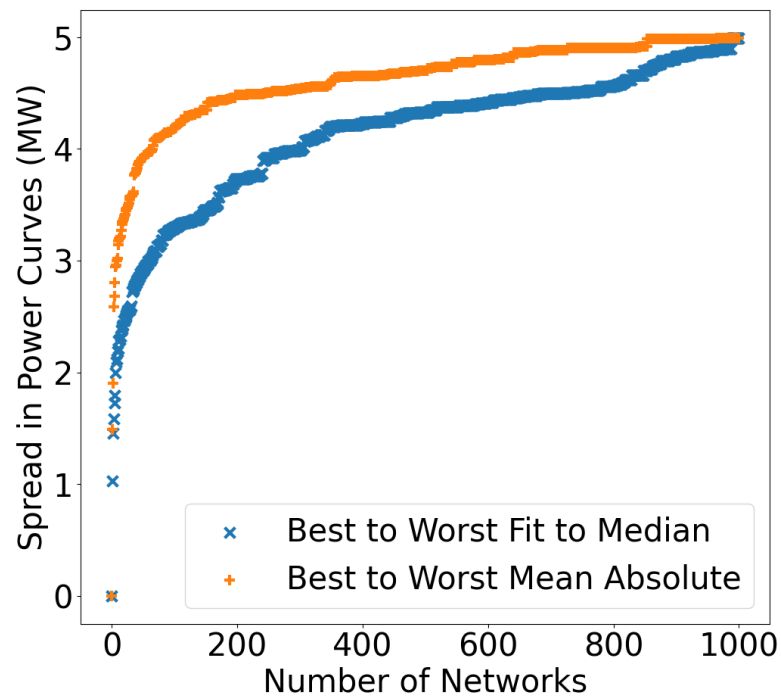


FIGURE B.1: Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error, averaged over all input-output curves.

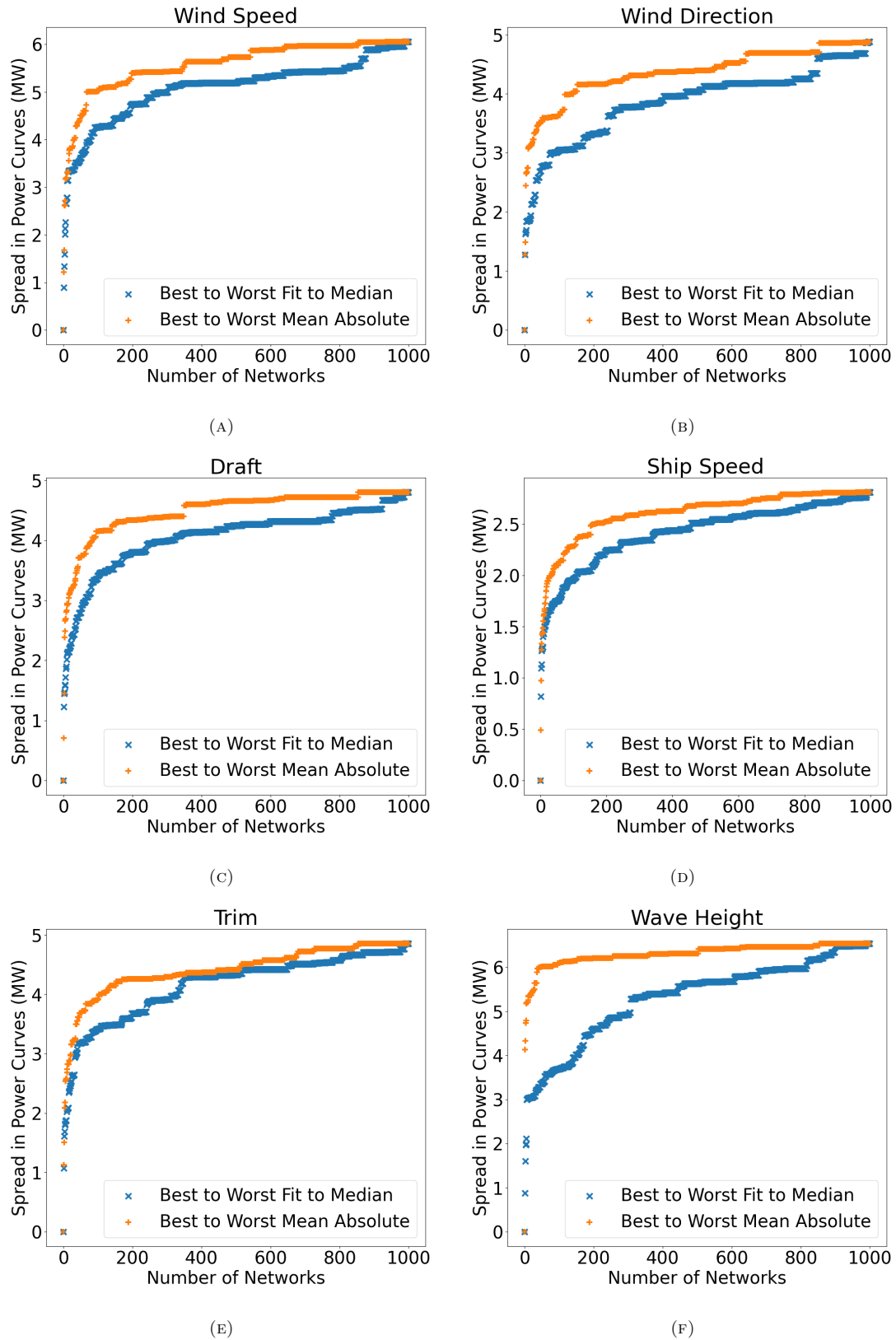


FIGURE B.2: Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error, for each input-output curve individually. (A) Wind Speed, (B) Wind Direction, (C) Draft, (D) Ship Speed, (E) Trim and (F) Wave Height.

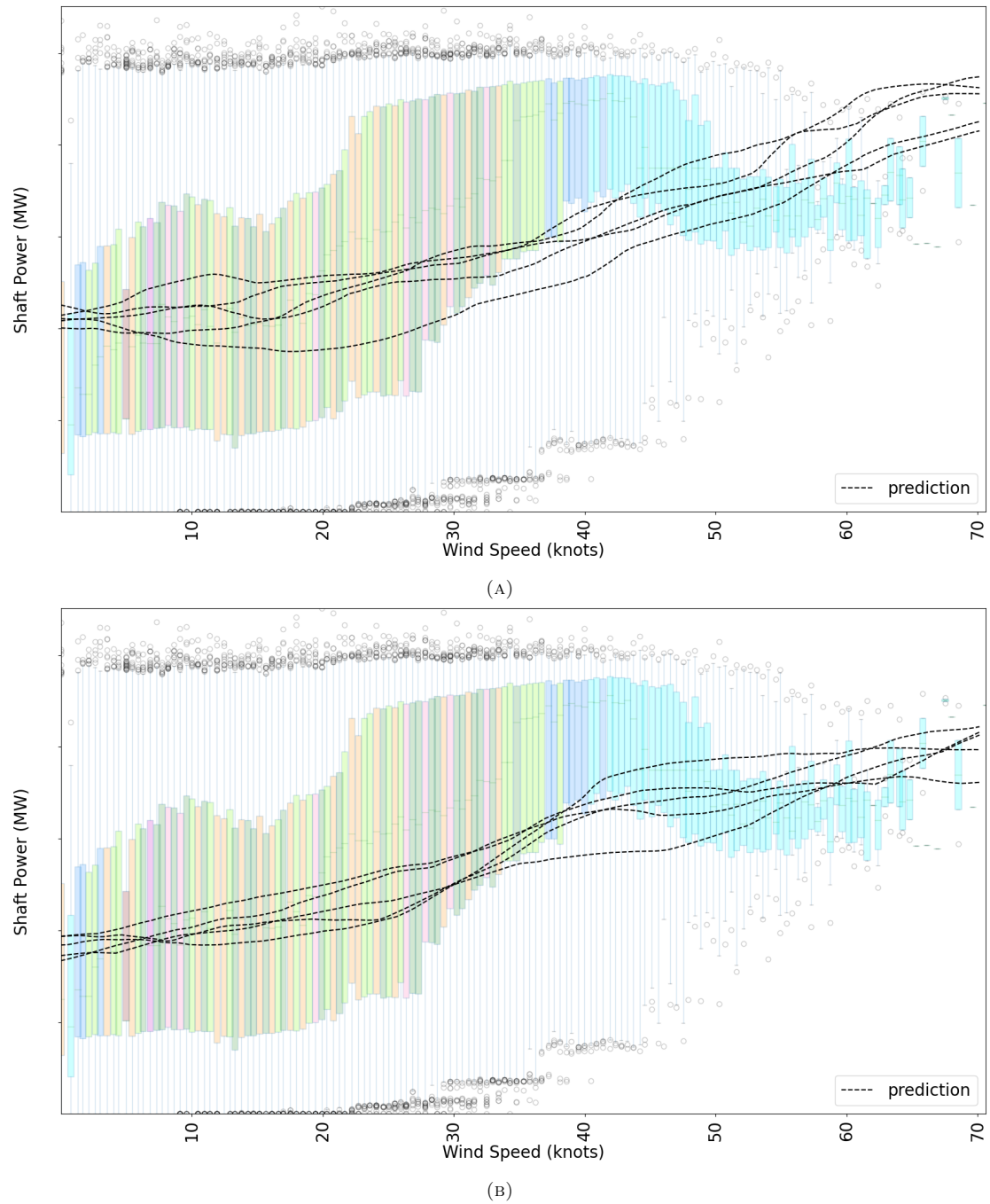


FIGURE B.3: Visualisations of isolated relationships learnt, between wind speed and power. (A) from the 5 networks out of 1,000 with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.

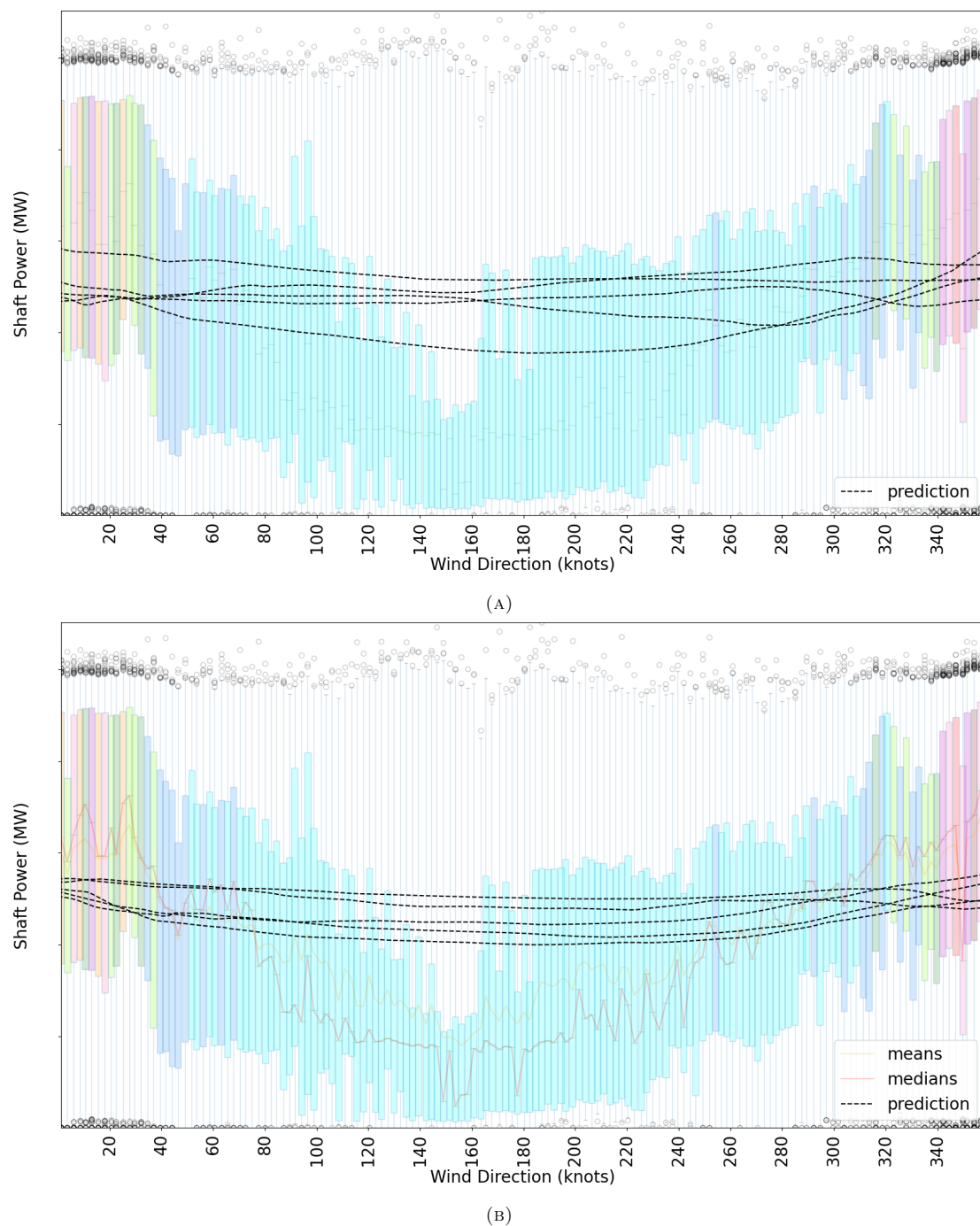


FIGURE B.4: Visualisations of isolated relationships learnt, between wind direction and power. (A) from the 5 networks out of 1,000 with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.

Appendix C

Supplementary Figures for Section 4.6

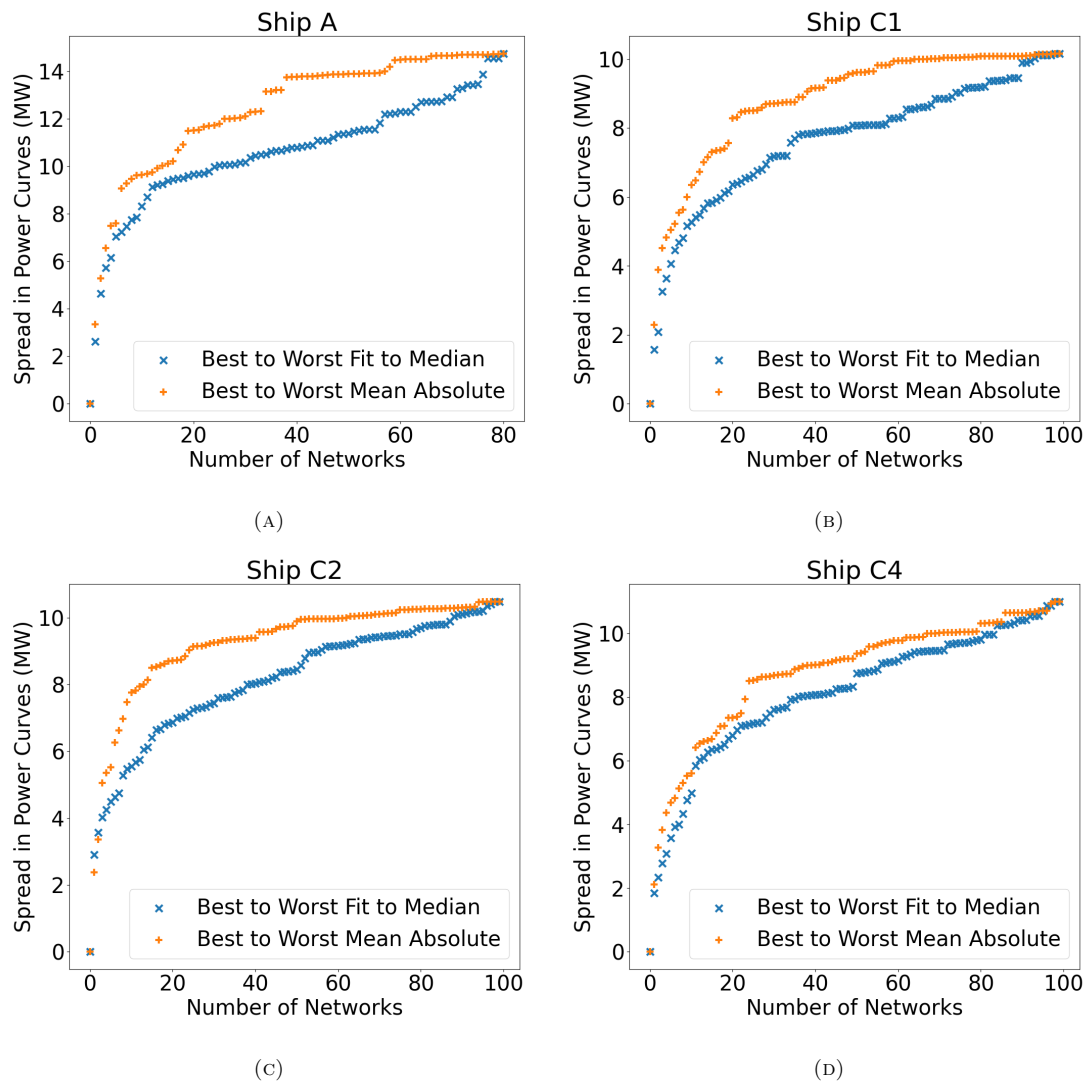


FIGURE C.1: Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error, averaged over all input-output curves for (A) Ship A, (B) Ship C1, (C) Ship C2, (D) Ship C4.

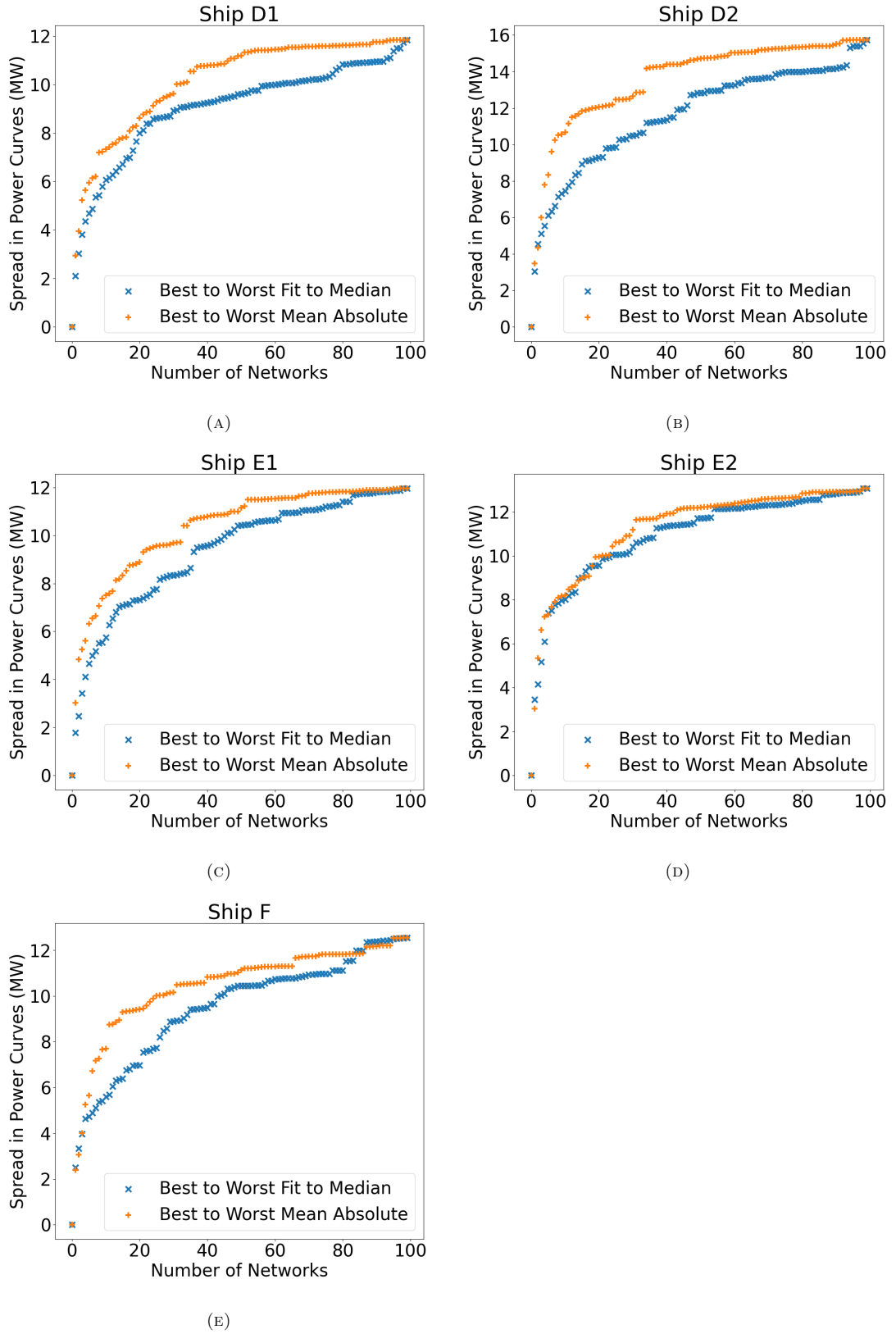


FIGURE C.2: Increase in spread for increasing number of networks, when ordered for increasing Mean Fit to Median Error and increasing Mean Absolute Relative Error, averaged over all input-output curves for (A) Ship D1, (B) Ship D2, (C) Ship E1, (D) Ship E2, and (E) Ship F.

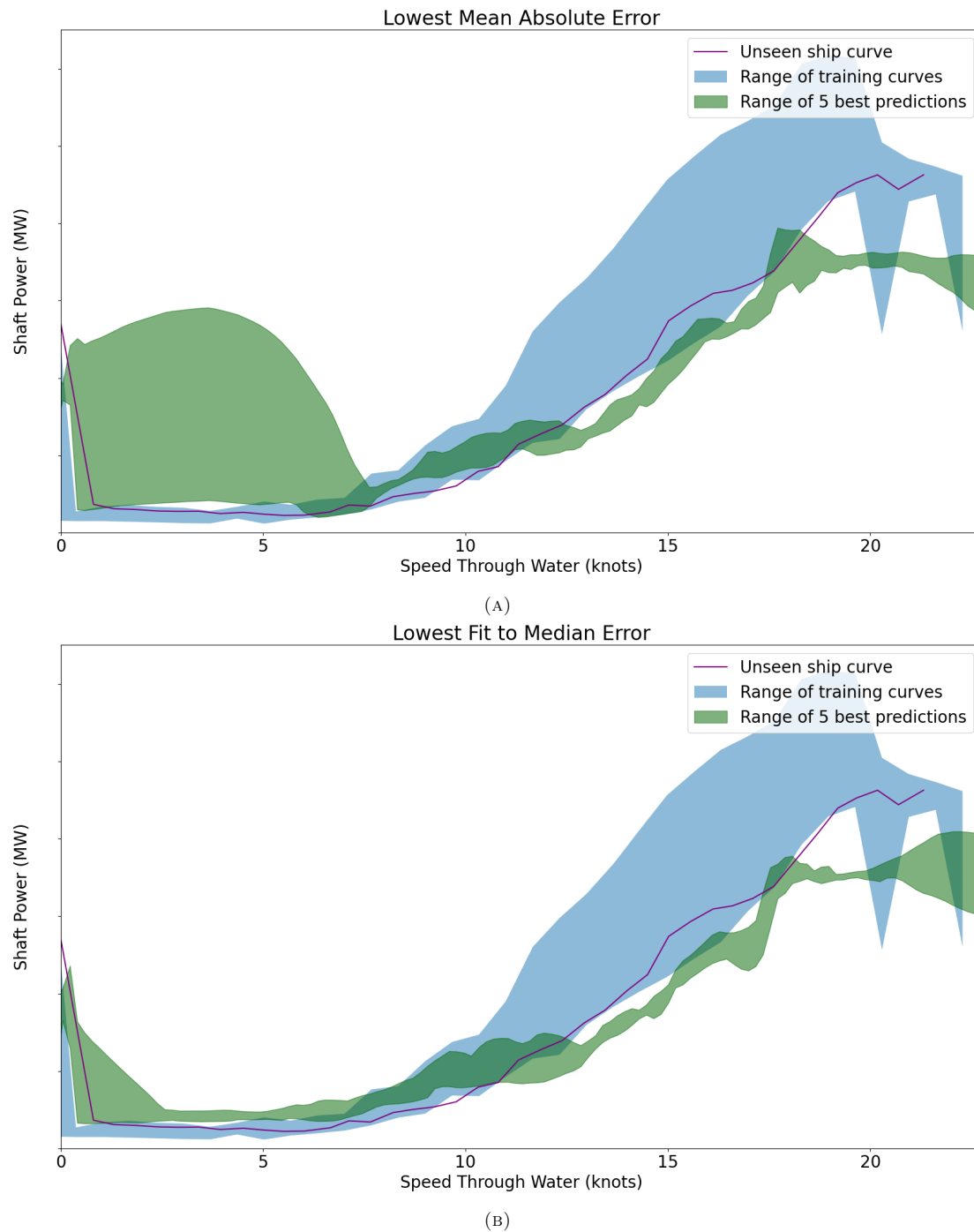


FIGURE C.3: Visualisations of range of isolated relationships learnt, between ship speed and power for Ship E2. (A) from the 5 networks with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.

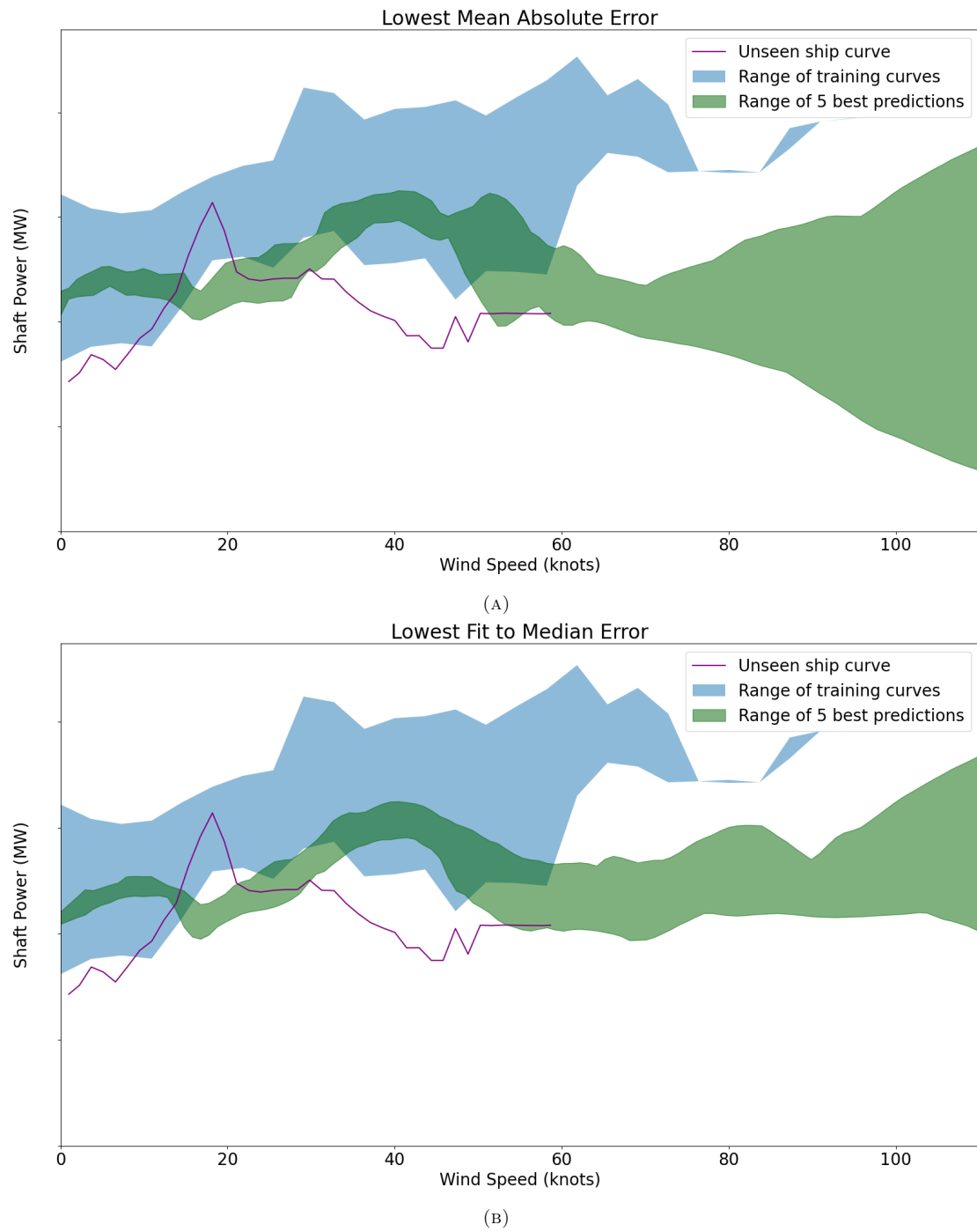


FIGURE C.4: Visualisations of range of isolated relationships learnt, between wind speed and power for Ship C1. (A) from the 5 networks with the lowest Mean Absolute Relative Error, and (B) from the 5 networks with the lowest Mean Fit to Median Error.