UNIVERSITY OF SOUTHAMPTON

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES
(FEPS)
ELECTRONICS AND COMPUTER SCIENCE (ECS)
NEXT-GENERATION COMPUTATIONAL MODELLING
(NGCM)

Integrative Modelling of Protein Abundance via Sequence Information

by

Gregory M. Parkes

Thesis for the degree of Doctor of Philosophy

August 2021

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES
Electronics and Computer Science

Doctor of Philosophy

INTEGRATIVE MODELLING OF PROTEIN ABUNDANCE VIA SEQUENCE INFORMATION

by Gregory M. Parkes

Understanding the complex interactions between the transcriptome and proteome is essential in uncovering cellular mechanisms both in health and disease contexts. The underwhelming correlation between corresponding transcript and protein abundance suggests that regulatory processes tightly govern information flow surrounding transcription, translation and post-translation; particularly in higher order organisms. Inherent difficulties associated with global proteome measurement make modelling protein abundance via proxies desirable, given the pivotal role that intra-cellular proteins play in cell regulation and function. In this thesis, a protein abundance predictor is developed across the human cell cycle using mRNA and translation abundance, determining that mRNA level alone insufficiently explains the transcriptome-proteome relationship. To expand the feature space, some 30 sequence-derived features (SDFs) were engineered that impact proteins before translation, and we demonstrated in our published works that overestimated outliers to fitted models ($r^2 = 0.67$) are associated with post-translational regulation and degradation. It made sense then to expand on

the concept of using sequence-engineered features as generalized predictors to expression; a large dataset was curated covering the entire human transcriptome to derive over 180 new features, spanning from genome to estimated post-translational modifications. SDFs were designed with scale and generality in mind; allowing for their application in a variety of 'omic studies. This newly generated resource was validated by systematically analysing intra-feature correlations and unsupervised learning techniques to mitigate inevitable multicollinearity. Finally, global protein abundance prediction using SDFs was attempted, finding that sequence information alone leads to model scores of $r^2 = 0.45$, with mRNA abundance included adding 5% to explaining model variance. Unpacking fitted SDF models using gene ontology analysis revealed a close relationship between SDFs and translation; helping to explain their improved model performance over mRNA level. This data-driven approach helps to isolate proteins of interest by outlier detection, with SDF use biased towards predicting steady-state protein abundance.

Contents

Notation

I have attempted to maintain a minimum level of necessary understanding with regards to mathematical notation within this thesis, however many of the concepts regarding statistical and machine learning theory require prerequisite knowledge in calculus, linear algebra and probability theory. Vectors are denoted by lower case bold Roman letters such as $\mathbf{x}$, and all vectors are assumed to be column-vectors. The superscript $^{\mathrm{T}}$ denotes the transpose of a vector or matrix, such that $\mathbf{x}^{\mathrm{T}}$ is a row-vector. The notation $(w_1, \ldots, w_M)$ denotes a row vector $\mathbf{w}^{\mathrm{T}}$ with M elements, whereas the corresponding column-vector is written as $(w_1, \ldots, w_M)^{\mathrm{T}}$. Uppercase bold Roman letters, such as $\mathbf{M}$, denote matrices. The identity matrix $\mathbf{I}$ refers to a matrix where elements $I_{ij} = 1$ where $i = j$ and zero where $i \neq j$. It will be common practice to refer to the number of samples/rows/data points as N, and the number of features/parameters/dimensions as P. If we have N values $(\mathbf{x}_1, \ldots, \mathbf{x}_N)$ of a P-dimensional vector $\mathbf{x} = (x_1, \ldots, x_P)^{\mathrm{T}}$ then these observations are combined into data matrix $\mathbf{X}$ in which the n^{th} row of $\mathbf{X}$ corresponds to the row vector $\mathbf{x}_N^{\mathrm{T}}$. When referring to the covariance between vectors $\mathrm{cov}(\mathbf{x}, \mathbf{x}) \equiv \mathrm{cov}(\mathbf{x}) \equiv \mathrm{var}(\mathbf{x})$. All *latent* parameters are denoted with Greek letters and *observed* parameters with Roman letters, but this may not be entirely consistent throughout the thesis.

Declaration of Authorship

I, Gregory Michael Parkes, declare that the thesis entitled **Integrative Modelling of Protein Abundance via Sequence Information** and the work presented in this thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- 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;
- where I have consulted the published work of others, this is always clearly attributed;
- 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;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by other and what I have contributed myself;
- parts of this work have been previously published [1].

Signed: _____

Date: _____

Acknowledgments

I would like to begin by thanking my supervisors Niranjan and Rob. In particular Niranjan was the one to spot me, convince the CDT to select me and has consistently believed in me from the very beginning of this research. His charisma, communication style, wit and frankness lead me not only to respect him as my supervisor, but also to admire him as a great teacher, role model and friend. To all of the staff, academics and students of the VLC research group. It remains in my view one of the leading research groups in the University for the fantastic community, social atmosphere and incredible research that each and everyone does. Trips to Stags, lunch breaks, table football, actual football, taco parties and barbecues barely scratch the surface of events we have done together as a research family.

To my close friends and/or former housemates, thank you for the many moments of joy, laughter and minor despair throughout the years of this work. To some it is a closing chapter for now, but for many I do not doubt shall dwell eternally. To my church family, thank you for your friendship, free meals and many other blessings you have provided me throughout my time in Southampton. To my wonderful parents, Michael and Carol, who brought me into this world and remain a bedrock of support. One could not ask for better parents - thank you for your continued love, support and prayers. To Almighty God, thank you for your provision, protection and anointing during these years and through those I have been fortunate to meet.

Veritas numquam perit. Pravda vítězí. La verdad prevalecerá.

"I live and love in God's peculiar light" - Michelangelo

Chapter 1

Introduction

1.1 Motivation

As we assume the majority of cellular behaviour is determined by the agglomeration of proteins at different concentrations, aberrant behaviour in the proteome is indicative of cell failure, such as by cancer, viral infection and/or disease. Further to this, a large number of proteins and modifications are used as candidates for biomarkers of known diseases. Systematic methods of predicting protein abundances could aid in identifying biomarkers within individuals when it comes to personalised/stratified treatments. There are many pitfalls associated with peptide-based mass spectrometry (MS) method used to quantify the proteome, such as the lack of unique sequence tags to identify specific proteins [2], the complex pipeline and its expensive cost. Hence indirect methods of estimating protein concentration via the transcriptome and other cheaper methods have grown in interest over recent years. This serves not only to overcome the shortfalls with proteomic experimentation, but also to reduce the experimental workspace by computationally modelling parts of the discovery workflow.

We build from previous work by Gunawardana [3] which introduces the hypothesis of finding post-translationally regulated proteins using a protein abundance predictor with features that inform *before* or during translation. This leads us to assume that predicted outlier proteins which do not correlate strongly with their actual (measured) concentration are more associated with regulation to their expression post-translation through proteolytic

degradation, modifications of translation factors and co-factor activity. Subsequently iterating this process whilst using new features will help to isolate relevant outliers through novelty detection to qualify protein function. One of the main interests in this domain from a computational standpoint is the unfathomable amounts of genetic data now publicly available in multiple forms, and how to compress this into useful information. Multi-'omics datasets quickly become intractable when combined in terms of computational resources, hence why so few studies (if any) have explored more than two 'omics sources in tandem.

1.2 Contributions

The following are the believed *novel* contributions within this Thesis:

1. **Combining outlier detection for PTR proteins with measured translation.** In Chapter 3 combined are the theoretical concepts by Gunawardana [3] with the measured translation data provided by Aviner [4, 5] to develop ML models with significant post-translationally regulated proteins as outliers. Gunawardana did not have access to measured translation abundance, and Aviner did not consider outlier detection analysis.

2. **Machine learning modelling considering with more than two measured 'omics in tandem.** In Chapter 3 predictors of protein abundance are developed utilising both mRNA and translation rates as inputs. Gunawardana [3] performed ML modelling with utilised mRNA vs. protein (2 'omics), and Aviner [5] utilised mRNA, translation and protein (3) but did not perform ML modelling.

3. **Generalized global human Sequence-Derived Feature dataset.** In Chapter 4 over 200 features are engineered to encompass the entire sequence transcriptome (N >17k). Comparable equivalents such as Vogel's [6] feature set provides around $N \sim 500$ samples with smaller P and insufficient coverage over the transcriptome. Other studies have studied sequence-based features in non-human species or performed mathematical modelling, usually with significantly fewer features [7, 8]. The developments of this dataset are considered a novelty and resource.

1.3 Report organization

This report is organised as follows. Chapter 2 presents a literature review which includes the Central Dogma of Molecular Biology, an introduction to features engineered from the sequence and the Cell Cycle. It also contains a review of Sequence Technologies and a large amount of statistical and machine learning (ML) theory. Finally, the literature review finishes with an overview and historical progression of transcriptome-proteome understanding, firstly by correlation and secondly by modelling. Chapter 3 develops linear and non-linear predictors across the HeLa cell cycle and provides evidence of post-translation regulators within outliers to such models. Chapter 4 develops a novel large sequence-derived feature corpus and compares its performance to previously considered sequence information. Chapter 5 utilises said corpus with protein-protein interaction networks to develop advanced protein abundance predictors, discovering the link between sequences and translation inference. Finally conclusions and future work are presented in Chapter 6.

1.4 Publications

* Paper - Parkes, G.M., Niranjan, M and Ewing, R. (2021). The Influences of Sequence-Derived Features across the Human Proteome. *Nucleic Acids Research*. Under review.

* Paper - Parkes, G.M. and Niranjan, M. (2019). Uncovering Extensive Post-Translation Regulation During Human Cell Cycle Progression By Integrative Multi-'omics Analysis. *BMC Bioinformatics*.

* Poster/Presentation - Parkes, G and Niranjan, M. (2018). Uncovering Extensive Post-Translation Regulation During Human Cell Cycle Progression By Integrative Multi-'omics Analysis. *CompBioMed Conference 2019*, Kings College London, UK, 23-26th September 2019.

* Poster - Parkes, G and Niranjan, M. (2018). Uncovering Extensive Post-Translation Regulation During Human Cell Cycle Progression By Integrative Multi-'omics Analysis. *Workshop on Quantitative Systems Biology 2018*, Kings College London, UK, 9th November 2018. Submitted.

Chapter 2

Literature Review

This chapter aims to provide a comprehensive introduction and review to the many interdisciplinary concepts touched upon in this research thus far. This material is divided into a number of key sections:

1. Central Dogma of Molecular Biology
2. Sequencing Technologies
3. Machine Learning and Statistical Theory
4. Transcriptome-Proteome Analysis

We begin with an introduction to the key compounds which govern genetics and the information of life, with cellular processes describing the flow of information from DNA to protein. Next, the interactions in the normal cell cycle process are explored, alongside technological developments that have enabled the genetic revolution over the last two decades. Then a number of the machine learning (ML) and statistical methods are covered and deployed in high-throughput analysis and their application to multi-'omics analysis. Finally the relationship between transcriptome-proteome as the precursor of this original research is considered and frames the genesis of this thesis.

2.1 Central Dogma of Molecular Biology

The central dogma of molecular biology describes the flow of genetic information encoding all of life to generate proteins from DNA to RNA to

protein. Francis Crick, whom with James Watson first proposed the double-helix structure of deoxyribonucleic acid (DNA) in 1953 (and republished by Nature in 1969) [9, 10], first coined the term and describes a protein synthesis pipeline containing three major classes of biopolymers; DNA, RNA and protein.

2.1.1 Basic Building Blocks

All of known biological life consists of information contained within a series of chemical molecules which transform, copy and delete themselves at appropriate moments within their lifecycle. In this section groundwork on these molecules before detailing the biological processes which influence upon them is covered.

DNA and RNA Deoxyribonucleic acid (DNA) is a variable-length stable polymer of nucleotides which contain the genetic information of an individual organism within the nucleus necessary for the development and activity of the cell. A single nucleotide consists of a pentose deoxyribose sugar, phosphate group and one of four possible bases; Guanine (G), Cytosine (C), Adenine (A) or Thymine (T). These bases are complementary, where adenine only pairs with thymine, and guanine only pairs with cytosine, via hydrogen bonding. [11, 12]. These base letters constitute the primary DNA sequence and henceforth the information is to be understood downstream by a collection of RNA and protein components. DNA as a structural compound is highly stable, and has possible future use cases as an efficient long storage mechanism for large datasets [13]. Ribonucleic acid (RNA) is similar to DNA, except T is replaced with uracil (U) [9, 10], and contains higher versatility and diversity of roles. A major subgroup of RNA is messenger RNA (mRNA), which carries transcribed information (genes) from the DNA to extra-nucleolar ribosomes for translation. See Table 2.1 for an overview of RNA types.

rRNA Ribosomal RNA (rRNA) is part of the non-coding RNA group, which makes up 80% of cellular RNA and constitutes roughly 60% of the ribosomal mass which play an essential role in translation of all mRNA [14], as it binds to ribosomal proteins to form small and large ribosome subunits. Production of rRNA is the rate-limiting step in the synthesis of a ribosome, which lends them a crucial role in protein production and indeed cellular

Type	Abbr.	Function	References
Messenger RNA	mRNA	Encodes amino acids	[9, 10]
Ribosomal RNA	rRNA	Translation	[14]
Transfer RNA	tRNA	Translation	[15]
Small nuclear RNA	snRNA	Splicing	[16]
Small nucleolar RNA	snoRNA	Nucleotide modification of RNAs	[16]
MicroRNA	miRNA	Gene/protein regulation	[16]
Long non-coding RNA	lncRNA	Regulation of epigenetic, transcriptional	[16]
Small interfering RNA	siRNA	Gene regulation	[16]

Table 2.1: *Overiew: Different types of RNA.*

activity. rRNA is synthesized in the nucleolus by RNA polymerase I using the specialty genes that encode for them. In humans, these are RNR, RNA18S, RNA28S and RNA5S families, as well as MT-RNR1, MT-RNR2, and MT-TV which are mitochondrial genes. rDNA sequences are heavily duplicated across eukaryotic genomes as tandem repeats, with humans having approximately 300-400 repeats which present in clusters on chromosomes 13, 14, 15, 21 and 22 [17]. rRNA sequences undergo substantial modification within the nucleolus before ribosomal integration, including methylation, folding and nucleolytic cleavage via snoRNA/protein complexes [18]. The process of rRNA synthesis is tightly regulated (particularly in eukaryotes) to maintain homeostasis; below are included a few of the interactions:

- Kinase AKT promotes rRNA synthesis as it regulates RNA polymerase I [19].
- Accumulation of angiogenic ribonucleases in the nucleolus can lead to increased rRNA transcription [20].
- Formation of heterochromatin in rDNA regions helps to silence transcription [21].

rRNA is ubiquitous across all living organisms, and the gene regions are heavily conserved across evolution. rRNA sequences can vary across a number of organisms, and hence can form unique configurations; these variants, such as the 16S and 18S, are widely used to discover evolutionary relation-

ships among organisms, and play a significant role in metataxanomics, as this provides a method for identifying bacterial species within a sample of unknown composition. This has seen a resurgence in interest following the research developments within the growing field of microbiomics [22].

miRNA microRNAs (miRNA) are single-stranded small non-coding RNAs (21-22nt) [23] that do not translate into proteins and primarily regulate mRNA expression by binding to the 3'-untranslated region (UTR). Two miRNAs were first discovered to regulate the timing of larval development in *C. elegans*, known as lin-4 [24] and let-7 [25], subsequently named as miRNAs when it became clear they were a large family of endogenous RNAs [26]. miRNAs are produced by transcription and by splicing of long non-coding RNAs (lncRNA), into an inactive pre-cursor form. This script is then extensively processed into an RNA-induced silencing complex (RISC) which complements target mRNAs to induce translational repression or degradation via deadenylation [27]. The regulation of miRNA genes occurs in similar fashion to protein-coding genes, such as auto-regulatory feedback loops or the regulation of miRNA maturation machinery, such as Drosha and Dicer enzymes [27]. miRNA molecules are highly stable as molecules, having half-lives in hours or days which is significantly longer than most other RNA types [28]. miRNA dysregulation is associated with tumorigenesis, whereby miRNAs can act as tumour suppressors such as in chronic lymphocytic leukaemia [29] or proto-oncogenes by upregulating Thiamine levels in cancer cell lines [30]. Further to this, many miRNAs play roles in a number of non-tumour disorders, such as neurological disorders and Down's syndrome [31]. Increasingly, miRNAs are utilised as biomarkers; due to their high stability, versatility, protection from RNase activity, long half-life and low cost in terms of assay development. Despite this, a significant drawback is the low sensitivity and specificity that miRNAs exhibit; meta-analyses demonstrate that certain miRNAs can be unreliable as biomarkers [32].

Protein Proteins are variable-length amino-acid polymers that perform a vast number of functions within living organisms. Each amino acid is an organic compound containing an amine (-NH2) and carboxylic acid (-COOH) group at each terminus, with a central chiral carbon joined to an R group. There are 20 known variants of R group which cumulatively provides unique and variant functionality within proteins. These amino acids join together to

form polypeptide chains which dictates the resulting secondary and unique tertiary structure of the protein once it is folded; this determines the functional activity of the protein. The order and selection of amino acids is determined by the corresponding gene sequence, as encoded by the 'genetic code'; a codon-triplet system whereby three DNA bases encode for each of the 20 possible amino acids. Folded proteins usually undergo post-translational modification (PTM), whereby chemical groups are attached to a number of amino acid residues. These PTMs can significantly alter the activity and function of the protein, for example increasing stability or altering the active site of an enzyme. Proteins fulfill many roles including catalysts (enzymes), DNA replication, stimuli response, molecular transportation, and cell signalling [33].

Most proteins are capable of folding into interesting and unique tertiary 3D structures, for many proteins this fold occurs naturally, but others require molecular chaperones to assist folding. The main types of structure are:

- **Primary structure**: The amino acid sequence.
- **Secondary structure**: Local structures within the amino acid sequence formed by hydrogen bonds. The most popular examples are α-helix, β-pleated sheets and coils. Many regions of secondary structure can exist on the same protein molecule.
- **Tertiary structure**: The overall 3D shape and structure of a single protein molecule. In addition to hydrogen bonds, disulfide bonds, PTMs and salt bridges, many proteins also contain a hydrophobic core (i.e resistant to water).
- **Quaternary structure**: A structure formed by several protein molecules bonding together to form a protein complex. A classic example of this is the tetrahedral Haemoglobin protein complex, formed from two Heme 'α' and 'β' groups.

Many proteins contain several protein domains, which are protein segments that fold into distinct structural arrangements. These domains have specific functionality, such as kinase activities or binding modules [34]. One of the primary advantages for this behaviour is that each domain can fold independently, reducing the complexity of residue interactions for particularly large polypeptide chains. Furthermore, these domains appear as motifs

which help to mediate protein-protein interaction. Short Linear Motifs (or SLiMs) are 3 to 11 contiguous amino acids that exist often in intrinsically disordered regions of a polypeptide, that upon interaction with a secondary partner induce secondary structure formation [35, 36]. Many of these motifs are recorded in the Eukaryotic Linear Motif (ELM) database [37].

2.1.2 Information Processing

Now that the basic building blocks of biological life are covered, the processes that bring about processing within the cell will also be detailed.

Transcription Transcription is the primary step, whereby a gene located on one or more of the chromosomes is transcribed/copied into an anti-parallel mRNA strand, also known as the primary transcript. Transcription proceeds as follows:

1. **Prelude: Histone modification**: The 3-D structure of the chromosome must enable physical access to the replicating enzyme, RNA polymerase. This is managed by histone proteins.

2. **Binding and Elongation**: The RNA Polymerase enzyme, alongside several transcription factors, forms a complex that binds to the promoter region on the gene of interest [44]. The bonds joining both complementary DNA bases are broken, and a complementary RNA-strand copy of the template strand is made [44]. Transcription rates vary by eukaryote, but can manage roughly 10-100 nucleotides per second, depending on the chromatin structure or the amount of methylation etc.

3. **Termination**: RNA polymerase moves along the template strand until it reaches the termination region, whereby the new RNA strand is released. The RNA is then preprocessed with polyadenylation, whereby a series of Adenines (A) are concatenated to the primary transcript 3' end [45].

If the gene is part of coding region, the resulting RNA is mRNA, which in turn serves as a template during translation to produce a protein. However the gene may also be non-coding, such as miRNA, rRNA or transfer

RNA (tRNA). In this case, the RNA produced will go on to separate post-transcriptional processing, or fold into a 3-dimensional RNA structure to perform separate functions in a cell.

Post-transcription Post-transcriptional activities involve all of the key events that occur between transcription and translation; and primarily concern the newly produced mRNA strand. They are broadly broken down into two focuses:

1. **Post-transcriptional modification**: Processing of precursor to mature RNA, 5' cap, 3' tail, splicing.

2. **Post-transcriptional regulation**: Regulation of transcripts, alternative splicing, nuclear degradation, processing, nuclear export.

In terms of mRNA processing, the primary transcript receives an added 7-methylguanosine (m^7G) to the 5' end, known as 5' capping. This is essential in helping the ribosome bind during translation, and helps to protect ti from exonuclease degradation. Notable exemptions from this process include mitochondrial mRNA [46] and plant chloroplastic mRNA [47]. In addition to this, around 250 adenine residues are added to the 3' end to form a poly(A) tail [45] protecting it from ribonuclease digestion. For protein-coding mRNAs, introns (non-coding sections) are spliced out and exons (coding-sections) are connected to produce the matured mRNA. The splicing reactions are conducted by a large complex called the spliceosome [48] which consists of snRNAs and proteins that recognise specific splice sites in the pre-mRNA, much like endonuclease enzymatic activity. Many pre-mRNAs have differential splicing options to produce different mature mRNAs from the same pre-mRNA sequence. This is known as alternative splicing [49] and is highly prevalent in the product of antibody proteins among other functional protein groups. This leads to a considerably greater variation in the proteome than in the transcriptome.

Post-transcriptional regulation is known to contribute substantially the control of gene expression both at the RNA and protein level. There are known mechanisms of feedback whereby mature RNA can interact directly with the genome (either self-feedback or another gene) or via complexes to regulate the expression of future RNAs [50]. They can also regulate other

RNA or protein located in any organelle as miRNAs by binding to the 3'-UTR region of other mRNAs [27].

Translation Translation is primarily concerned with protein synthesis via ribosome activity, and occurs either in the cytosol or in the rough endoplasmic reticulum (RER). Translation as a process is broadly split into three phases: initiation, elongation and termination. Translation initiation is complex and involves at least 10 proteins (see Figure 2.1 for illustration). Both ribosomal subunits must assemble around the mRNA start codon, which is downstream from the 5'-UTR. In eukaryotes this is nearly always AUG which encodes for the amino acid methionine (M) [15]. tRNAs which are associated to one of the 20 amino acids then attempt to bind with the mRNA-ribosome complex on the next codon triplet in the sequence, with only a tRNA with the correct anti-sense codon being successful. The tRNA transfers its amino acid to the tRNA corresponding to the next codon, shortly before the ribosome translocates to the next codon and so on, forming a primary amino acid sequence (or chain). This process repeats until a STOP codon is reached (UAG, UAA or UGA), whereby the ribosome releases a new nascent polypeptide chain [15]. Like the 5'-to-3' direction of DNA/RNA, polypeptide chains are directed N-terminus to C-terminus, whereby the first amino acid is near the amino-group (N for NH_2), and the final amino acid is near the carboyl group (C for COOH).

The rate of translation varies substantially by organism; in general it is considerably higher in prokaryotes (up to 17-21 amino acid/s) than eukaryotes (6-9 amino acid/s) [51, 52]. Further to this, the rate can be affected by many factors, such as the prominence of AUGs [53], temperature, pH, ATP abundance and others. The ATP required via translation for locomotion is significant, once one factors in the movement of mRNA, tRNA binding and peptide bond formation.

Post-translation Nascent polypeptide chains produced post-translation require extensive post-translational modification (PTM) into the mature protein product. These modifications often involve appending chemical groups to certain amino-acid side chains, expanding the repertoire of normal R groups and modifying existing chemical groups. Common functional modifications include glycosylation, phosphorylation, acetylation and lipidation,

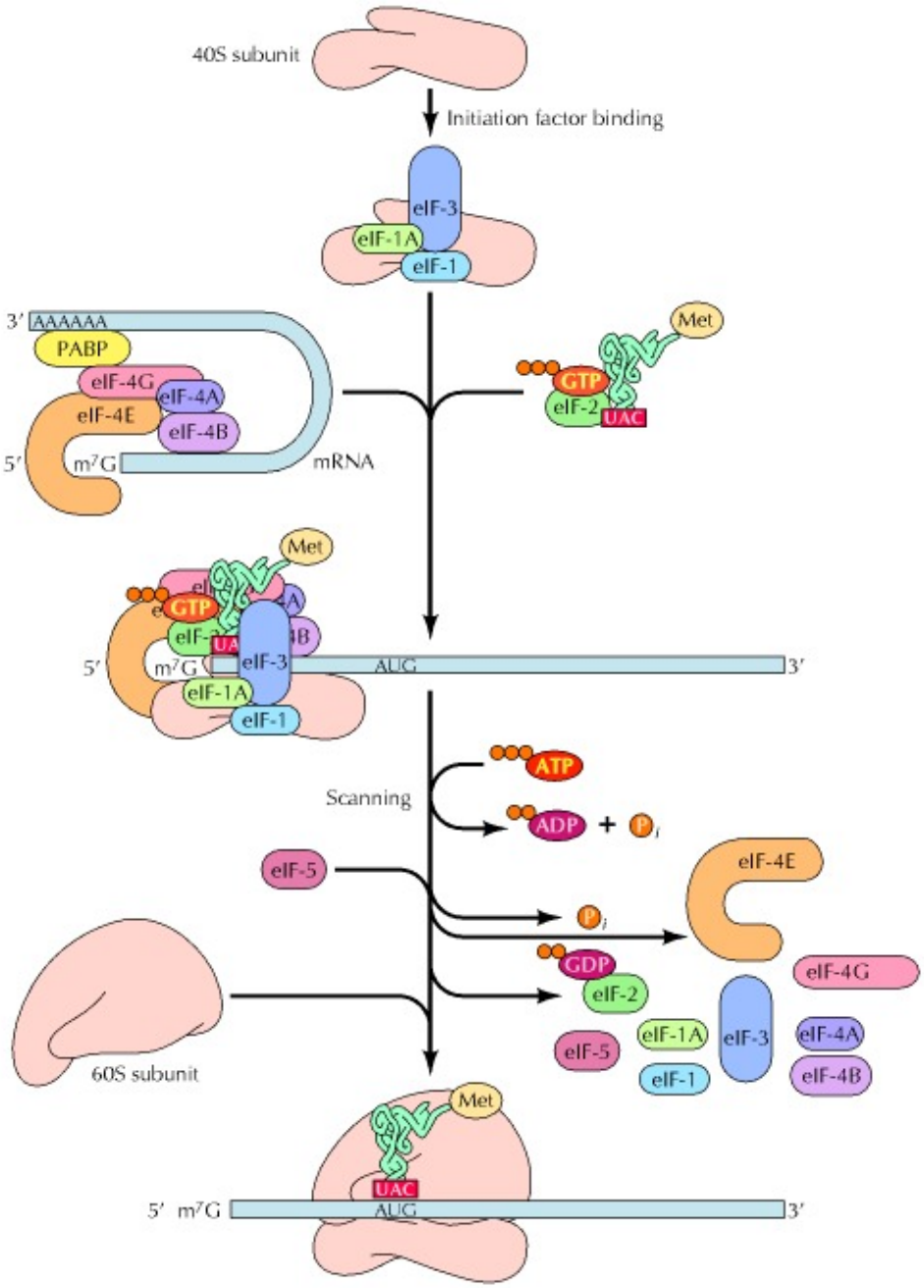


Figure 2.1: Translation initiation of a mRNA molecule. *Ribosome is represented in cyan. mRNA strand in orange, tRNA molecules in yellow. Image is taken with permission from the book: 'The Cell: A Molecular Approach. 2nd edition., Figure 7.10'*

which serve a host of functions such as improving stability, cell signalling and adhesion [54]. Other forms of PTM consist of proteolysis, such as cleaving peptide bonds to remove the initiator methionine (M) residue. Prevalent modifications are covered in detail within Table 2.2 below:

PTM	Description
Phosphorylation	As the most frequent modification (1/3 of the human proteome)[55], this process involves the addition of phosphate groups to serine, threonine and tyrosine [56]. Functionalities include multi-level regulation, protein degradation, enzyme regulation and modulation of CDKs [56].
Acetylation	The second-most common modification, disproportionately found among chromatin/metabolic enzyme proteins. Acetylation affects gene expression and metabolic rates, in addition to protein stability/localization [56].
Glycosylation	Linked to improved protein folding, stabilization, cell-to-cell adhesion and immunology. Glycoproteins have high heterogeneity with their proteins having highly diverse roles in the proteome [57].
Ubiquitination	Named after ubiquitin, due to its presence ubiquitously, its mark is commonly known to signal protein degradation via the 26S proteasome, relocalise proteins or inhibit PPIs [58].

Table 2.2: *Descriptions of the most common PTMs.*

2.1.3 Codon Bias

Analysis of DNA sequence material has been a significant field of interest since its discovery, and accelerated beyond the Human Genome Project. Given the rich textual-based format of sequence information, there has been significant progress in deriving properties about genes and their downstream proteins. These have a very wide scope and include:

1. The comparison of sequences: To find similarity between DNA or other sequences across species is very common for evolutionary discovery.

2. Identifying intrinsic features of the sequence, such as *active sites*, post-translational modification sites, introns, exons and regulatory components.

3. Estimating the 2-D and 3-D structure.

One of the most interesting sequence-derived features has been the differences in base frequency of occurrence across different species, also known as *codon usage bias*. Codon bias is factoring in differences in frequency between synonymous codons in the coding sequence [59]. This is because natural selection will balance between mutational bias and translational optimization, in addition to reflecting the available tRNA pool to the cell. In addition to these; GC content, raw base frequencies, biophysical properties of amino acids, and text-mined features all contribute to the global picture of static features. In this section, more intricate SDFs will also be covered.

GC content This simply reports the fraction of G and C bases that fall within a given sequence s_i, as scaled by the sequence length n_i.

Codon Adaptation Index In the literature, there is a strong assumption that gene products are likely to correspond to biased amino acid composition that might minimize the biosynthesis energy costs of translation and its rate [53]. Firstly define a reference table of relative synonymouse codon usage values from highly expressed genes:

$$RSCU_{ij} = \frac{X_{ij}}{\frac{1}{n}\sum_{j=1}^{n_i} X_{ij}} \tag{2.1}$$

where X_{ij} is the number of occurrences of the jth codon for the ith amino acid, n is the number of alternative codons for amino acid i [60]. The relative adaptability of a codon w_{ij} is then:

$$w_{ij} = \frac{RSCU_{ij}}{RSCU_{imax}} - \frac{X_{ij}}{X_{imax}} \tag{2.2}$$

where $RSCU_{imax}$ and X_{imax} are the most frequently used codon for ith amino acid over j. CAI is then calculated as the geometric mean over the weights:

$$CAI = \left(\prod_{k=1}^{L} w_k \right)^{1/L} \tag{2.3}$$

Relative Codon Bias Proposed by Roymondal (2009) [61], by attempting to eliminate the many artefacts of previous metrics, such as varying sequence length. Let $f(x, y, z)$ be the normalized codon frequency for the codon triplet (x, y, z) of a gene, the RCB of a codon triplet is then defined as:

$$d_{xyz} = \frac{f(x, y, z) - f_1(x) f_2(y) f_3(z)}{f_1(x) f_2(y) f_3(z)} \tag{2.4}$$

where $f_1(x)$ is the normalized frequency of base x at codon position 1, $f_2(y)$ is the normalized frequency of base y at position 2, etc. Normalization of frequency occurs over the gene length in codons. The total RCB of a gene is then the geometric mean of all codon biases:

$$\text{RCB} = \left(\prod_{i=1}^{L} \left[1 + d^i_{xyz} \right] \right)^{1/L} - 1 \tag{2.5}$$

where L is the number of codons in the gene, d^i_{xyz} is the codon usage difference of codon i [61]. RCB values close to 0 indicate a lack of bias for the codons, higher values indicate more bias.

tRNA Adaptation Index This metric is a measure of translational efficiency which takes into account the intracellular concentration of tRNA molecules and the efficiencies of each codon-anticodon pairing [62]. The estimated translational efficiency of the ith codon (out of 61) is given thus:

$$W_i = \sum_{j=1}^{n_i} (1 - s_{ij})\, tGCN_{ij} \tag{2.6}$$

where n_i is the number of tRNA types/anticodons that pair with codon i, $tGCN_{ij}$ is the tRNA gene copy number (retrieved from the genomic tRNA database) [63], s_{ij} weights represent wobble interactions between codon i and j. Normalized weights w_i are then obtained from W_i by scaling them with respect to the maximum value among all codons.

2.1.4 The Cell Cycle

All known living eukaryote and prokaryote cells survive by growing and then dividing into two daughter cells. This process is rigorously controlled at all levels, as the cell environment must be maintained near constancy to ensure survival. See Figure 2.2 for illustration.

2.1.4.1 Cell cycle phases

G_1 phase corresponds to the first growth phase, replicating organelles and obtaining nutrients. This then leads into S phase, where each of the chromosomes is semi-conservatively replicated. Once this is complete, the cell enters a second growth phase (G_2), and mitosis (M); where the cell divides into two daughter cells. There are notable exceptions, such as during embryonic development, where the cells only undergo S and M phase, and do not have any growth phases within their cell cycle.

G_1 phase Primary focuses in this phase involve cell growth in size, with synthesis of mRNAs and histones required for the next stage; DNA synthesis. Further to this, biosynthesis is greatly increased, and duplicates organelles such as mitochondria and ribosomes. The duration during this phase varies considerably depending on the type of cell in question. For a typical rapidly proliferating 24-hour human cell, G_1 could be expected to take around 11 hours, constituting roughly 45% of the entire cycle [64]. Environmental factors such as nutrient supply and temperature can limit growth, with cell senescence (G_0) if it is unable to meet the prerequisites. G_1 is tightly regulated as most of the CDK inhibitors are highly expressed, with a peak in Cyclin E1 (CCNE1) towards the end of G_1 [65, 66]. The tumour suppressor protein pRB binds to E2F family transcription factors to down-regulate S phase cyclins via Ubiquitin E3 ligases [58].

S phase Once a cell passes the G_1/S checkpoint, it undergoes intensive DNA replication, whereby every chromosome and centrosome is duplicated to form two sister chromatids. Rates of RNA transcription and protein synthesis fall by orders of magnitude, with an exception to histone production, which is mostly high during S phase. The concentration of DNA gradually doubles throughout the time in this phase, which has been observed [64]. The pathways that govern this replication are highly conserved, as to minimize

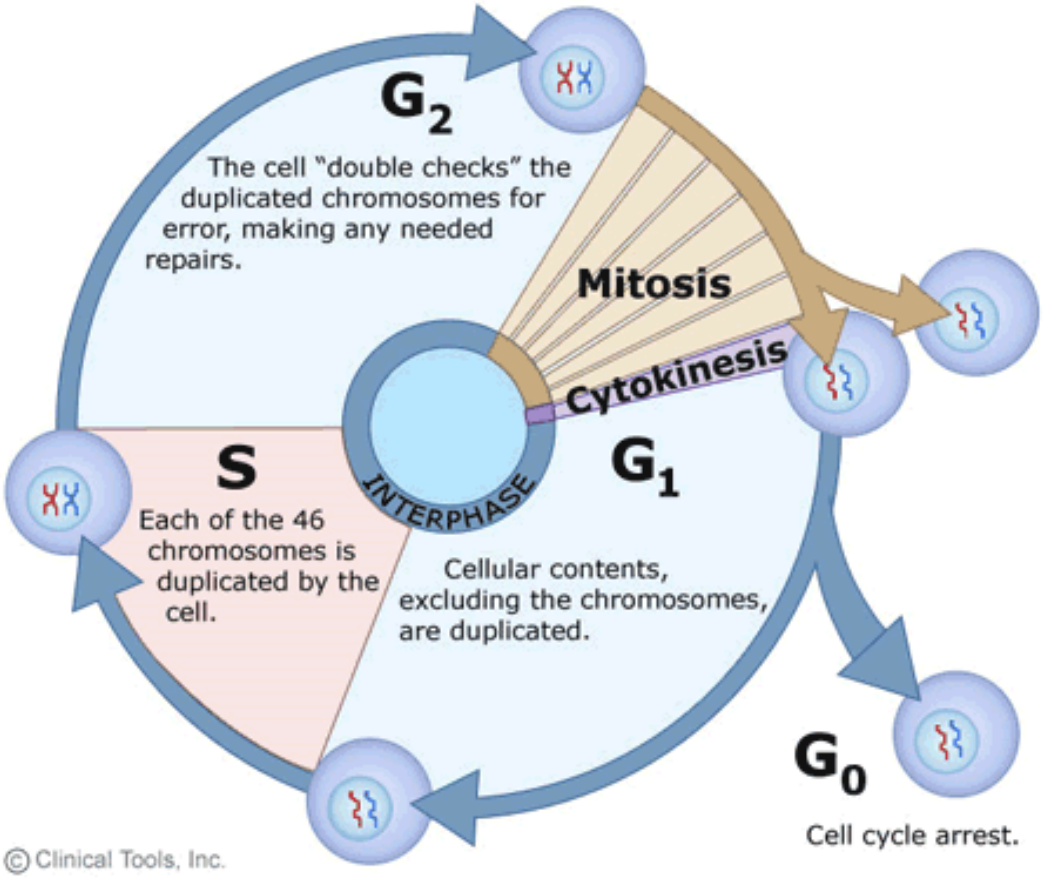


Figure 2.2: The Cell Cycle. *Diagram assumes human cell cycle, given there are 46 chromosomes duplicated at S phase. Each area section is not to timescale. I acknowledge the University of Leicester in producing this image.*

single nucleotide polymorphisms (SNP) or errors that may accrue through aberrant copying [66].

G_2 phase Once DNA replication is complete, the cell undergoes a second growth/gap phase known as G_2. This phase is shorter than that of the previous gap phase, and aids to prepare the cell for mitosis by synthesising necessary proteins and, in particular, cytoskeletal infrastructure like microtubules [64]. Another major checkpoint (G_2/M) at the end of this phase must be passed before the cell undergoes mitosis , requiring a threshold activation of cyclin B1/CDK1 (MPF) [65].

M phase Mitosis is by far the most complicated phase within the cell cycle, and begins with nuclear membrane dissolution. Mitosis and cytokinesis take roughly 1 hour, and hence represent only 5% of the time spent in one cycle [64]. Mitosis occurs (Table 2.3) as follows:

Step	Short Description
Prophase	Chromatin condense and become visible, nucleolus dissolves, formation of spindles [64]
Metaphase	Spindles associate with kinetochores on centromeres, Centromeres align to equatorial plate with microtubules, key checkpoint of chromosome alignment [64]
Anaphase	Anaphase-promoting complex (APC) cascades to separate sister chromatids, which are pulled to pole-ends of the cell [64, 65]
Telophase	Nuclear envelope reforms to produce two cells, spindle fibres degradated, chromatin decondensation begins [64]
Cytokinesis	Cytoplasm divides into two daughter cells, cytoskeletal backbone rearranged

Table 2.3: *Steps within M phase.*

Once a cell finishes cytokinesis, it is considered to have returned to interphase/G_1.

G_0 phase Traditionally thought of as a *resting phase*, G_0 describes cellular senescense whereby the cell still undergoes metabolic activity but does not progress in the cell cycle (i.e does not duplicate). Cells in G_0 that can return

to G_1 are quiescent from extrinsic signal pathways, characterized by low RNA concentration and turnover. Senescence is irreversible and caused by DNA damage or degradation, which could be from a number of internal or external factors [67]. Most fully differentiated cells are in G_0, particularly mature erythrocytes and neurons.

2.1.4.2 Cell cycle checkpoints

The cell cycle has various 'checkpoint' mechanisms which halt progression until it can ensure an earlier process has been completed. Most commonly this is due to DNA damage response (DDR), which comprises of proteins that signal DNA damage to downstream effectors that arrest the cell cycle and promote DNA repair [68].

Checkpoint	Description
G_1-to-S	ATM kinase activated by dsDNA breaks, cell cycle arrested by Chk2. ATM induces p53 by lowering Mdm2 affinity, Stable p53 promotes DNA repair and apoptosis [65, 68]
S	If significant DNA damage detected during DNA replication, replication is halted. ATR activates Chk1 inducing Cdc25A degradation, preventing M phase progression [68]
G_2-to-M	Entry requires high Cdk1, which is inhibited by T14 phosphorylation induced by Wee1/Myt1 kinases [69, 68]
Spindle assembly	Correct partitioning of chromatids in anaphase is protected, where APC/C is inhibited, cycB/securin is delayed until spindles associate to all chromosomes [68]

Table 2.4: *The most important cell cycle checkpoints.*

There are many mathematical models of the eukaryotic cell cycle, and particularly the checkpoint/progression milestones. Much of the early work was done using ordinary differential equations (ODEs) by Tyson [70, 71] using *P. polycephalum* and *X. embryos*. Other models such as stochastic, boolean and hybrid models have also been attempted [72], including recently in mammalian somatic cells [73]. One of the key challenges with this approach is providing strong levels of data-driven modelling to accompany and validate stimulations.

2.2 Sequencing Technologies

As mentioned previously, since the discovery of DNA and its helical structure [9, 10], great advances in the complexity, diversity and intelligibility of genomes have been reached, many of which beginning with the completion of the Human Genome Project [74, 75, 76]. The field of DNA sequencing has thereafter moved extraordinarily quickly, in terms of cost reduction per megabase, throughput and diversity of species for which the full genome has been sequenced. Here a number of the key technologies are covered, that have developed into what is now known as Next-Generation Sequencing (NGS) and even beyond. Dozens of next-generation sequencing companies and technologies have formed, resulting in the emergence of bioinformatics as a major scientific sub-discipline [77]. Significant global consortiums since the advent of NGS include the 1000 Genomes Project [78], the Exome Sequencing Project [79], and the UK's 100,000 Genomes Project which are the beginnings of attempts for population-scale sequencing efforts. Whilst initial efforts have mainly been focused on human genome sequencing, a number of projects now aim to sequence other species also, such as the 100K Pathogen Genome Project based in the University of California, Davis.

NGS technologies broadly fall into two categories: *Short-read* approaches aim to sequence a small DNA region which is lower cost and higher accuracy, whereas *long-read* approaches enable *de novo* genome assembly by providing longer sequence read lengths. In this section we begin with microarray technologies and then move on to cover NGS technologies.

2.2.1 DNA microarrays

Microarray technology is one of the oldest forms of sequencing technologies and have been used extensively in research since the mid-to-late 1980s [86]. This involves immobilizing different single-stranded DNAs (ssDNA) on a substrate in distinct and separate wells [87]. Target DNA is labelled with a fluorescent probe and hybridized on to the array. The intensity value of the light signal produced (if the target DNA binds to a particular ssDNA) can be converted using Beer-Lambert law to estimate the number of bound molecules. Microarrays have a vast number of applications, such as identifying single-nucleotide polymorphisms (SNPs); which are variations in the DNA that may be indicative of disease, and genome-wide association study

(GWAS) analysis [88]. DNA microarrays can easily be adapted to measure expression levels by measuring the amount of gene-specific cDNA, which is the corresponding mRNA strand that has been transcribed back to DNA by the viral enzyme *reverse transcriptase*. One of the major benefits of microarrays are their very low cost in comparison to NGS technologies, however technical complications can arise in relation to normalization and hybridization of certain probes.

2.2.2 Short-read NGS

Short-read sequencing technologies broadly fall under two main categories: Sequencing By Ligation (SBL) and Sequencing By Synthesis (SBS) [85]. In both approaches, DNA is clonally amplified on a solid surface, where thousands of identical copies of DNA fragments are produced in parallel, each with their own reaction centre, thus allowing the sequencing of many millions of DNA molecules at the same time. In this subsection a number of the key techniques are explored, alongside principles and companies which inhabit the NGS biosphere.

Generation of Clonal Template Populations Generating the clonal template population as a prerequisite to SBL/SBS elicits a number of strategies; such as a) bead-based, b) solid-state or c) DNA nanoball generation methods. As a precursor to the subsequent methods, the sample DNA is fragmented, followed by ligation to a common adaptor set for clonal amplification and sequencing.

- **Bead-based** - An adaptor is used that is complementary to an oligonucleotide fragment immobilized on a bead. Emulsion PCR (emPCR) is then used to amplify the DNA template to create millions of clonal DNA fragments. These beads can then be distributed/arrayed onto a large surface.
- **Solid-state** - Instead of using emPCR (as with bead-based), amplification occurs directly on a slide, using forward and reverse primers which initiate replication.
- **DNA nanoball** - One set of adapters are ligated to either end of a DNA template, forming a template ring. The circular DNA templates are then cleaved downstream of the adapter sequence and iteratively

ligated to integrate different adapters. These templates are then amplified to generate DNA nanoballs, which are then hybridized onto a patterned flow cell.

Sequencing by Ligation SBL approaches involve the hybridization and ligation of labelled probe and anchor sequences to a DNA strand. The labelled probe encodes one or two known bases (as an encoding mechanism) and a series of degenerate/universal bases, which complementarily bind between the probe and template [89]. The anchor fragment encodes a known complementary sequence to an adapter sequence and provides an initiation site for ligation.

The SOLiD platform (by ThermoFisher) utilizes two-base-encoded probes, whereby each fluorometric signal represents a dinucleotide. Because there are 16 combinations of each dinucleotide leading to issues with spectral resolution, only 4 fluorescent signals are used, where each represents a DNA base. Thus the combination of these colours leads to a colour-space result, which requires post-experimental deconvolution using data analysis techniques. Following cluster generation/bead deposition onto a slide, fragments are sequenced by ligation and added to the DNA library. The two-base probe is ligated onto an anchor that is complementary to an adapter, and the slide is imaged to identify the first two bases in each fragment. Unextended strands are capped by unlabelled probes to maintain cycle synchronization. Finally, terminal degenerate bases and the fluorophore are cleaved off the probe, leaving a five base pair extended fragment. This process is repeated ten times until two out of every five bases are identified. [85]

The other main SBL approach is using Complete Genomics (BGI); whereby DNA is sequenced using a combinatorial probe-anchor ligation (cPAL) approach. Post-DNA nanoball deposition, a complementary anchor to one of four adapter sequences and a fluorophore-labelled probe are bound to each nanoball. The probe is degenerate at all but the first position. Anchor and probe are then ligated into position and imaged to identify the first base on either the 3' or 5' side of the anchor. The probe-anchor complex is then removed and the process is repeated with the same anchor but a different probe with known base at $n + i$ positions, where $i < 5$ is the number of iterations.

Sequencing by Synthesis SBS approaches describe a large array of DNA-polymerase-dependent methods, but we will focus on two approaches in particular; Cyclic Reversible Termination (CRT) and Single-Nucleotide Addition (SNA) methods.

CRT approaches, such as those adopted by Illumina and Qiagen, use terminator molecules similar to those used in Sanger sequencing, whereby the ribose 3'-OH group is blocked to prevent elongation [90]. A DNA template is primed by a complementary basr sequence to an adapter region, which initiates DNA polymerase binding to the dsDNA region. During each cycle, all four labelled and 3'-blocked dNTPs are added, and incorporated into each elongating complementary strand. Imaging technology (total internal reflection flourescence microscopy) relies on the flourescence of each dNTP to determine sequence for each cycle.

SNA approaches (such as *Ion Torrent*), also known as Pyrosequencing, on the other hand rely on a single signal to mark dNTP incorporation into an elongating strand. Subsequently, each of the four nucleotides must be added sequentially to ensure only one dNTP is responsible for the signal. This requires no blocking as the absence of the next nucleotide in the sequence will prevent automatic elongation [85].

2.2.3 Long-read NGS

Genomes are highly complex with regions that can contain many repetitive elements, copy number alterations and structural variants [91]. As a consequence to this complexity, short-read technologies are often insufficient to resolve them, as read-lengths rarely exceed one kilobase. Long-read sequencing on the other hand delivers reads in excess of several kilobases, allowing for the resolution of large structural features. Long-read sequencing also plays a key role in transcriptome research, as a long-read can cover entire mRNA transcripts, which can uncover novelties regarding exon-intron interactions and gene isoforms. The two main categories of long-read sequencing currently used are single-molecule real-time (SMRT) sequencing, and synthetic strategies which adapt from short-read technologies to construct longer-reads *in silico* [85]. These technologies are mentioned for the readers interest but they are not utilized within this research, nor are measurements derived from these particular technologies.

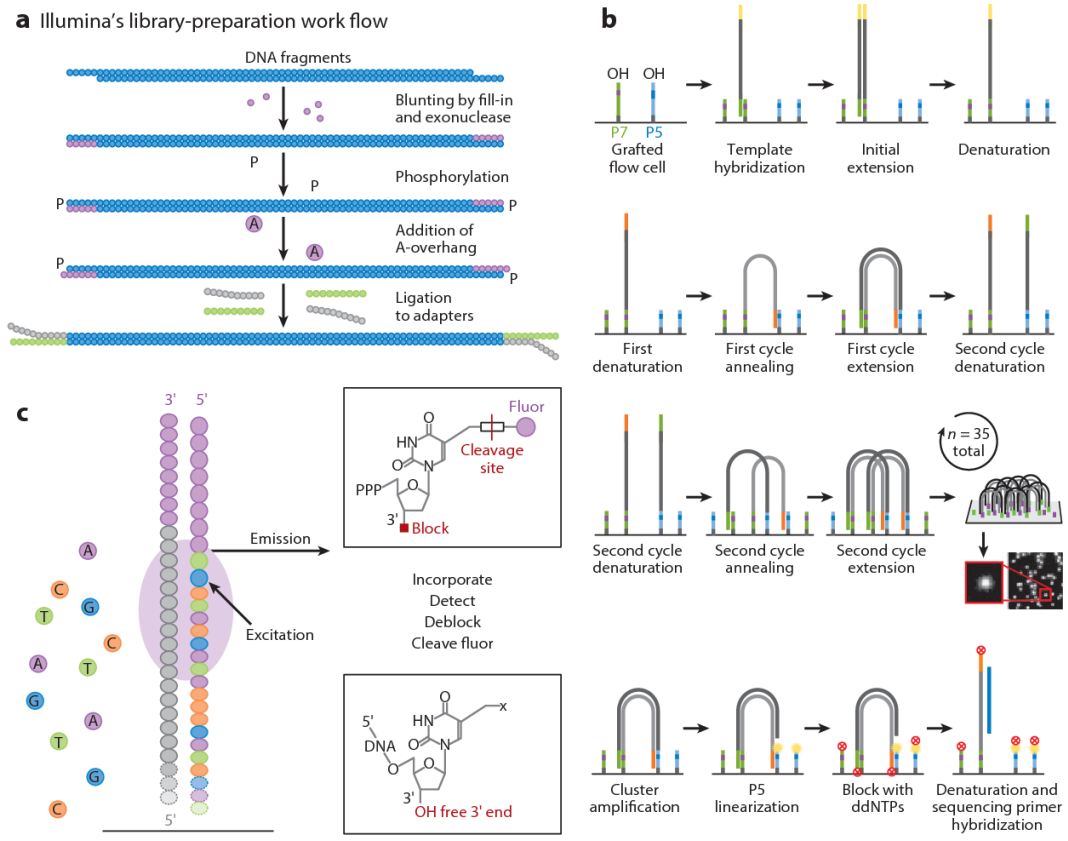


Figure 2.3: The Illumina Sequencing Platform pipeline.
This image is taken with permission from Mardis, 2013 [76].

2.2.4 Proteomic Techniques

Proteomics is the study of characterizing and measuring proteins within a cell or organism. The goals of such analysis are in the quantitative and/or qualitative properties that proteins possess, such as their profile, interaction network, concentrations and more. Understanding the assumptions and techniques behind measuring protein abundance, predicting structural and functional properties of proteins is crucial in bridging the gap between transcriptomics and proteomics, and forms a bedrock to the targets of prediction within the work of this thesis. Whilst the sum of techniques developed is vast and is a research thesis in an of itself, only techniques that can measure protein expression level will be mentioned, as this is the primary target for machine learning algorithms as conducted within this thesis.

Analysis and separation of proteins and other compounds is well known, with early techniques expanding from Gel electrophoresis in 1D (as is common in DNA analysis) to using 2D Gel electrophoresis (2D-E) and 2D Flourescense Differential Gel electrophoresis (DIGE). The separated proteins are then stained with compounds such as silver, which binds to cysteine-groups in the proteins. Silver quantity can be determined by the relative darkness in a gel area under UV light, which relates to the quantity of protein in a given gel area. Isotope-Coded Affinity Tag (ICAT) uses isotopic labelling *in vitro*, and Isobaric Tag for Relative and Absolute Quantification (iTRAQ) uses isobaric labelling; in conjunction with chromatography and Mass Spectrometry (MS) for quantitative proteomics [92].

Here a table summary is provided of some of these proteomics techniques, and we will focus on MS in more detail as this is the technique that the majority of our proteomic data is collected by.

2.2.4.1 Mass Spectrometry

Used for mass analysis of protein characterization, MS is one of the most versatile and comprehensive tools available for large-scale proteomics. Due to some inherent limitations of biological MS [94], the MS setup pipeline (including sample preparation, front-end preparation, ionization, data acquisition and analysis) differs depending on sample complexity and goals of said analysis [95, 96]. Mass spectrometers consist of an ion source and optics,

Technology	Applications	Strengths	Weaknesses
2D-E	Separate proteins; Profiling quant. expressions	Relative quant. expression	Some proteins poor separation; low abundance
DIGE	Separate proteins; Profiling quant. expressions	Relative quant. expression; High sensitivity; Variability reduction	Requires unique visualization; Expensive; Proteins require lysine
ICAT	Chemical isotope labelling for quant. proteomics	High sensitivity; High reproducibility; Can detect low expression levels	Acidic proteins not detectable
iTRAQ	Isobaric tagging of peptides	Relative quant. expression; High through-put; Parallelizable	Requires fractionation of peptides; Increased sample complexity
MS	Protein identification; Protein characterization	High sensitivity; Very high through-put; High specificity; Relative qual. and quant. expression	Protein fragments must be ionizable; Variable sensitivity; Expensive;

Table 2.5: *An overview of proteomic techniques, as discussed by Chandramouli and Qian (2009) [92, 93].*

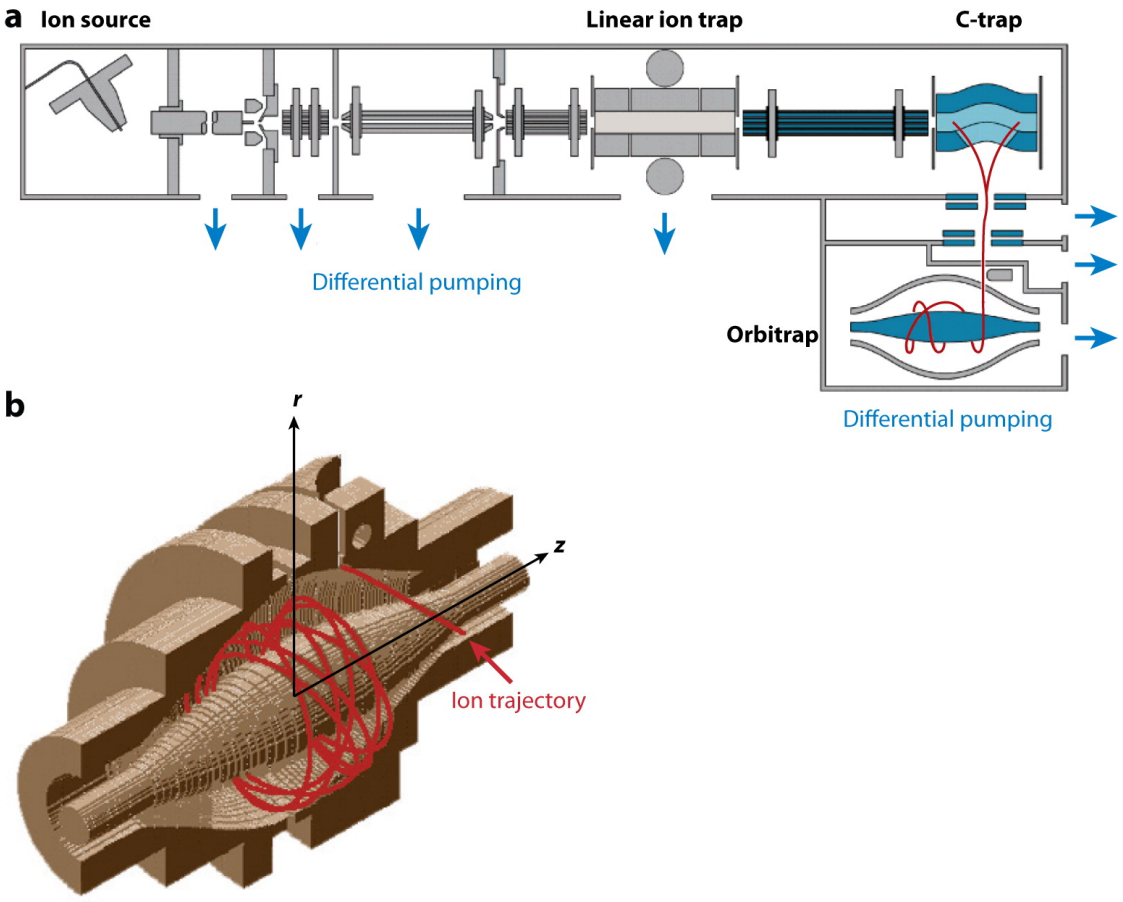
the mass analyzer, and electronics to perform data processing. Peptides are converted to a gaseous form and projected through the spectrometer until they reach the detector, which measures the mass-to-charge (m/z) ratio of the particle. Liquid or gas chromatography (GC/LC) can often be deployed pre-MS as this helps to separate peptide fragments by mass. Here the key components of the mass spectrometer will be covered in more detail and cover the advantages/disadvantages with differences in methodology.

One of the major technological developments that have enabled MS analysis is the soft ionization techniques, as proteins are polar, nonvolatile and thermally unstable. Ionization transfers an analyte to the protein within the gas phase without extensive degradation to the peptides. Two of the main technologies for achieving this are:

- **MALDI** - Short for 'matrix-assisted laser desorption ionization', this technique deploys rapid laser heating to the MALDI matrix, causing desorption of matrix $[M+H]^+$ ion analytes into the MS gas phase [97].
- **ESI** - Short for 'electrospray ionization', analyte ions are provided from an electrified, high-voltage solution. The spray is released between the inlet and emitter of the mass spectrometer [98].

Mass analysers are an integral aspect of each mass spectrometer since they can store ions and separate based on mass-to-charge ratios. A number of technologies exist to do this, including Ion Trap, Orbitrap and Ion Cyclotron resonance (ICR) analyzers, which can separate ions based on their m/z resonance frequency, m/z stability and time-of-flight (TOF) analysis to determine the time of flight of each analyte. Often mass spectrometers can have hybrid functionality combining these analysers such that different needs can be met during analysis. The detail of these different instruments is beyond the scope of this work, but highlight the depth of the field of proteomics analysis via this family of techniques [96].

Proteins are identified by m/z of their peptides and fragments, which means that biological samples require separation before performing MS to allow for unambiguous identification. This plays a significant role as the accuracy and sensitivity of the experiment rely on sufficient separation. Historic techniques include various gel-based methods such as 2D-E as described previously [92], however current technologies include:



Yates JR, et al. 2009.
Annu. Rev. Biomed. Eng. 11:49–79

Figure 2.4: Schematic of the LTQ-Orbitrap hybrid instrument by Thermo Scientific. *(a) Overall diagram of the LTQ-Orbitrap. (b) Cross section of the Orbitrap analyzer. Taken from Yates et al. 2009 with permission.*

- **HPLC** - High pressure liquid chromatography directly couples to instruments with an ESI source, which allows for a continuous separation pipeline directly fed into a mass spectrometer. This is the most common technique, and there are many variants of LC/MS dependent on application [99].
- **RPLC** - Reverse phase liquid chromatography is different as it separates compounds based on their hydrophobicity, and subsequent buffers are compatible with ESI [100].

Often pipelines require adjustment for the identification and quantification of phosphorylated proteins, or other post-translational modifications associated with peptides. For example, in Phosphoproteomics, a selective enrichment technique using immobilised Fe^{3+} ions is used to selectively bind to phosphorylation sites.

2.3 Machine Learning and Statistical Theory

In this thesis an array of machine learning (ML) algorithms are considered and applied to different bioinformatics problems, ranging from protein abundance prediction and covariance analysis on sequence-derived features. Here we will cover the basics of regression and tree-based models in addition to going into more detail for the algorithms that give current state-of-the-art with the notable exception of deep learning. Note that classification techniques are largely ignored in this Thesis as nearly all of the response variables we analyse are continuous.

2.3.1 Linear Regression Models

The aim of any multivariate linear regression model is to predict the value of one or more continuous target variables y given the value of a P-dimensional vector $\mathbf{x}$ of *input* variables. These models are always *linear* with respect to the parameters, however the input variables can undergo non-linear transformations via basis functions $\phi(\cdot)$. Given a training set $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^{N}$ with N samples, a hypothesis function $h(\mathbf{X})$ attempts to predict $\mathbf{y}$ given some unknown parameters $\mathbf{w} = (w_1, \ldots, w_P)^{\mathrm{T}}$. Mathematically this is formulated as:

$$\hat{y}_n \equiv h(\mathbf{x}_n) = w_0 + \sum_{p=1}^{P} w_p \phi_p(x_{np}), \quad \forall n \tag{2.7}$$

where $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_P)^{\mathrm{T}}$ are the optional non-linear basis functions (e.g Gaussian). The parameter w_0 is the bias parameter or fixed offset. It is common to introduce a dummy basis function $\phi_0(\mathbf{x}_0) = \vec{1}_N$ such that the term w_0 vanishes from (2.7), leading to the design matrix $\boldsymbol{\Phi}$ and matrix-vector product $h(\boldsymbol{\Phi}) = \boldsymbol{\Phi}\mathbf{w}$. Throughout this chapter we will now on use $\boldsymbol{\Phi}$ and $\mathbf{X}$ to refer to the design matrix interchangeably. The unknown or un-modelled effects of the linear model are accounted by an error term $\boldsymbol{\epsilon}$:

$$y_n = \mathbf{w}^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{x}_n) + \epsilon_n, \quad \forall n \tag{2.8}$$

where $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_N)$ is distributed $\epsilon_i \sim \mathcal{N}(0, \lambda^{-1})$ using scalar precision λ, and refers to the *residual* which quantifies the remaining difference between the predicted and actual target value. Note that the Gaussian distribution assumption plays an important role in model interpretation, but can be relaxed in Generalized Linear Models (GLM) to any distribution belonging to the exponential family. See Figure (2.5) for an illustration using the sum-of-squares error function (6.1). Now we can write the likelihood (probability of observed data given parameters) as:

$$p(\mathbf{y}|\mathbf{w}, \lambda) = \prod_{n=1}^{N} \mathcal{N}(y_n | h(\mathbf{x}_n), \lambda^{-1}) \tag{2.9}$$

This likelihood can be maximised with respect to each of the parameters $\mathbf{w}, \lambda$ to obtain point estimates (maximum likelihood), via solving of the partial derivatives.

Generalized Linear Models (GLM) Another linear model extension where the residuals $\boldsymbol{\epsilon}$ do not need to be normally distributed, but can belong to any exponential distribution family member. In practice, this means transforming the response variable with an *activation function* $g(\cdot)$ such that it can vary linearly with respect to the predictors. The mean of the distribution chosen then corresponds to the prediction of that target variable:

$$\mathbb{E}[y_n | \boldsymbol{\phi}(\mathbf{x}_n)] = \mu_n = g(\mathbf{w}^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{x}_n)) \tag{2.10}$$

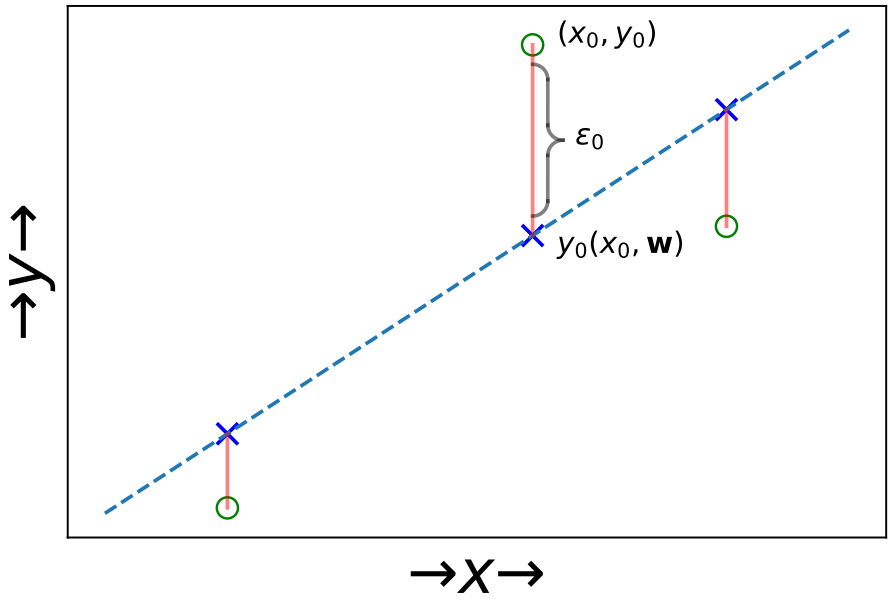


Figure 2.5: *The error function (eq 6.1) corresponds to the summation over all residuals ϵ_n with respect to each data point (x_n, y_n). Blue crosses/line represent predictions, green circles are (input, target) pairs, red lines indicate residuals.*

here μ_n does not represent the mean of $\mathbf{x}_n$, but rather the predicted value for y_n. Logistic regression is a special case GLM which essentially performs a classification task.

Multivariate Adaptive Regression Splines (MARS) A natural extension to linear models is MARS [102, 103], a non-parametric regression technique that can model non-linearities and variable interactions automatically. If y is approximated using an expansion of basis functions $\boldsymbol{\phi}$:

$$\hat{f}(\mathbf{x}) = w_0 + \sum_{p=1}^{P} w_p \prod_{k=1}^{K_p} \left[s_{kp} \cdot (x_{v(k,p)} - t_{kp}) \right]_+ \tag{2.11}$$

which is very similar to equation 2.7, where $s_{kp} = \pm 1$, t_{kp} is the split point for parameter p on polynomial k. The entire region within the square brackets is known as a two-sided truncated power spline function in the form:

$$b_q^{\pm}(x-t) = [\pm(x-t)]_+^q \tag{2.12}$$

where t is the knot location and q is the order of the spline. Hence the basis functions $\phi_p^{(q)}(\mathbf{x}) = \prod_{p=1}^{P} b_q^{\pm}(x_{v(k,p)} - t_{kp})$. Algorithmically, the approximation is learnt in two stages:

1. **Forward stepwise**: Adds basis functions as pairs in an additive fashion to the model by minimizing RSS, to find optimal hinge points t. The process of searching over all variables to add basis functions occurs until Δ RSS is small or maximum number of terms is reached.

2. **Backwards stepwise**: Forward pass has a tendency to cause significant overfitting. Backward stepwise prunes the model by deleting unnecessary basis functions via generalized cross-validation (GCV) of the form:

$$GCV(P) = \sum_{n=1}^{N} \left[\hat{f}_P(\mathbf{x}_n) - y_n\right]^2 \frac{1}{[N - C(P)]^2} \tag{2.13}$$

where $C(P)$ is a complexity cost function as determined by the number of basis functions.

2.3.2 Linear Model Estimation

To estimate the parameters of a linear regression model, several techniques have been developed which vary in computational complexity, assumptions made or re-imagined and/or robustness. Various regularization techniques are also considered in order to constraint the parameters from overfitting. The following estimation techniques described below provide a beginner platform for complex modelling of biological regulation. The assumption of linearity mostly holds for the data we are working with, assuming appropriate and well-defined reversible transformations such as log, and in combination with stronger interpretability, we use them particularly in feature selection, to find features of interest and for a base-level accuracy to improve upon.

Ordinary least squares (OLS) Assuming a linear model with unknown coefficients $\mathbf{w} = \{w_0, w_1, w_2, \ldots, w_P\}$ and design matrix $\mathbf{\Phi}$ we minimize the loss/residual sum of squares between the observed responses $\mathbf{y}$ and the predicted responses by linear approximation [104]. Mathematically we can rearrange equation (2.8) to formulate the sum-of-squares error function (6.1):

$$\mathcal{E}_{\mathbf{y}|\mathbf{\Phi}} = \mathcal{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left(\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) - y_n\right)^2 \tag{2.14}$$

Including one-half is traditionally done as this simplifies differentiation but is not necessary. This function is also known as the *residual sum of squares* (RSS) or the sum of squared errors. This minimization is a convex error function and has a unique solution, provided that the P-columns of the design matrix $\mathbf{\Phi}$ are linearly independent:

$$\hat{\mathbf{w}}_{\mathrm{OLS}} = \mathbf{\Phi}^{\dagger}\mathbf{y} \tag{2.15}$$

$$= \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{y} \tag{2.16}$$

where $\mathbf{\Phi}^{\dagger}$ is the Moore-Penrose pseudoinverse (see Supplementary 6.2 for more details). If features are correlated and columns have linear dependence, the design matrix becomes singular and $\left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}$ cannot be inverted [101]. The negative gradient of $\mathcal{E}(\mathbf{w})$ leads to an analytical solution of $\mathbf{w}$ in the specified conditions. Alternatively, the optimal weights can be estimated iteratively via gradient descent or via the singular value decomposition (SVD). In this case let the decomposition $\mathbf{\Phi} = \mathbf{U}\mathbf{S}\mathbf{V}^T$. Then we have:

$$\hat{\mathbf{w}}_{\mathrm{OLS}} = \mathbf{V}\mathbf{S}^{\dagger}\mathbf{U}^T\mathbf{y}$$

where $\mathbf{S}$ is a diagonal matrix of singular values, $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices ($\mathbf{U}^T\mathbf{U} = \mathbf{U}\mathbf{U}^T = \mathbf{I}$ and $\mathbf{V}^T\mathbf{V} = \mathbf{V}\mathbf{V}^T = \mathbf{I}$). We could assume that there is a linear relationship between the concentrations of [mRNA] and [protein]. In this case OLS would seem to be a reasonable starting choice given its simplicity and interpretability.

Ridge In complex models with many parameters, or where $P \gg N$, models can very easily *overfit*. This can be overcome by imposing a penalty term on the coefficient magnitude. The most popular ways of doing this is by

using the ℓ_1 and/or ℓ_2-norms. For example, using Ridge [106] regression our objective function becomes:

$$\mathcal{E}_R(\mathbf{w}) = \mathcal{E}(\mathbf{w}) + \frac{\alpha}{2}\mathbf{w}^T\mathbf{w}, \quad \alpha \geq 0 \tag{2.17}$$

where $\mathbf{w}^T\mathbf{w}$ corresponds to the ℓ_2-norm, and α is a regularization parameter that controls the amount of shrinkage. The larger α is, the greater the amount of shrinkage and thus features become more robust to collinearity. In this case $||\mathbf{w}||_2^2$ takes the ℓ_2-norm which has the effect of smoothing the weights. Ridge preserves the convex optimization problem leading to a global minimum and unique solution as with OLS:

$$\hat{\mathbf{w}}_R = (\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi} + \alpha\mathbf{I})^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{y} \tag{2.18}$$

where $\mathbf{I}_P$ is the identity matrix with P dimensions. We illustrate the impact of Ridge regularization on the sinusoidal dataset (see Fig 2.6), where increasing values of α on the weights generated by the polynomial eliminate overfitting of the predictions.

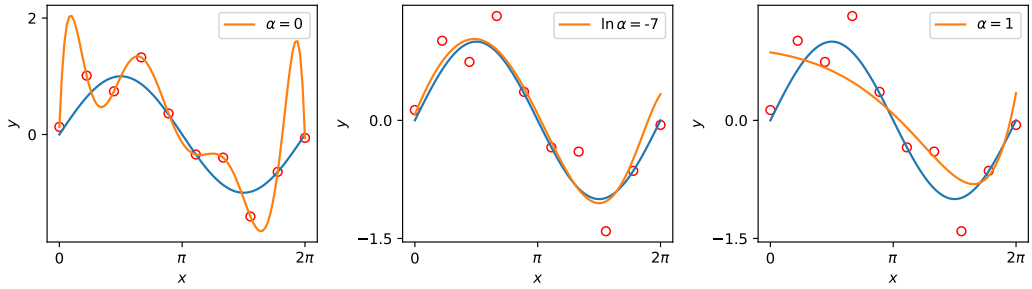


Figure 2.6: ***Effects of Ridge regularization with varying parameters of*** α**.** *Data vector* $\mathbf{x} \sim \mathcal{U}(0, 2\pi)$ *and* $y = \sin x$*.* $P = 9$ *polynomial terms are generated to construct* $\mathbf{X}$ *followed by z-score transformation. From left to right: No regularization,* $\ln \alpha = -7$ *and* $\alpha = 1$*. As* $\alpha \to \infty$*,* $\mathbf{w} \to 0$*. Fitted values in orange, data points in red, true values in blue. Middle graph visually gives the best fit.*

Lasso Short for *least absolute shrinkage and selection operator*, here an ℓ_1-norm cost is applied on the loss function [107]:

$$\mathcal{E}_L(\mathbf{w}) = \mathcal{E}(\mathbf{w}) + \alpha \left|\left|\mathbf{w}\right|\right|_1 \tag{2.19}$$

where $\alpha \geq 0$ and $\left|\left|\mathbf{w}\right|\right|_1$ is the ℓ_1-norm. The use of this norm has the effect of inducing sparsity in $\mathbf{w}$, causing certain features to not be considered in linear model calculations. This leads to Lasso also being used for *feature selection* as well as for prediction. However, if certain features $\mathbf{x}_i$ and $\mathbf{x}_j$ happen to correlate strongly, the feature dropped is usually dependent on the random initialization of the weights (w_i, w_j) at runtime and this can lead to training inconsistencies. This can exacerbate if multiple features $\mathbf{X}_K, K > 2$ all co-correlate, as only one out of K features is selected. From a Bayesian perspective, the coefficients can be considered to be drawn from a *Laplace* prior distribution, which peaks sharply at zero.

Elastic Net A very popular recent addition, which combines Ridge and Lasso is known as Elastic Net developed by Zou and Hastie [108], where it attempts to capture the benefits of both techniques into a single model:

$$\mathcal{E}_E(\mathbf{w}) = \mathcal{E}(\mathbf{w}) + \alpha_1 \left|\left|\mathbf{w}\right|\right|_1 + \alpha_2 \left|\left|\mathbf{w}\right|\right|_2^2 \tag{2.20}$$

where $\alpha_1 \geq 0, \alpha_2 \geq 0$ are parameters controlling the ℓ_1 and ℓ_2-norms, respectively. This helps to overcome problems with Lasso, which has a tendency to random select one variable from a group of highly correlated variables, but also induces sparsity which is desirable when $P \gg N$, where Ridge does not. It is common practice from a computational perspective to re-arrange the regularizers such that:

$$\alpha = \alpha_1 + \alpha_2 \tag{2.21}$$

$$\gamma = \frac{\alpha_1}{\alpha_1 + \alpha_2} \tag{2.22}$$

and so re-organize the objective function to be:

$$\mathcal{E}_E(\mathbf{w}) = \mathcal{E}(\mathbf{w}) + \alpha\gamma \left|\left|\mathbf{w}\right|\right|_1 + \alpha(1-\gamma) \left|\left|\mathbf{w}\right|\right|_2^2 \tag{2.23}$$

thus $\gamma \in [0, 1]$ acts as a ratio between Ridge and Lasso regularization; as $\gamma \to 0$, the Lasso term increases and Ridge decreases, and the opposite holds as $\gamma \to 1$.

Bayesian Linear Regression A frequentist approach assumes data is generated from the model as described in (equation 2.8), instead a Bayesian perspective assumes the responses y_n are sampled from a probability distribution such as a Gaussian distribution:

$$\mathbf{y} \sim \mathcal{N}(\mathbf{\Phi w}, \lambda^{-1}) \tag{2.24}$$

where $\mathbf{\Phi w}$ represents the maximum likelihood estimate for μ. further to this, the parameters in addition to the responses are assumed to be sampled from an appropriate distribution; thus the objective is to determine the posterior distribution using *Bayes theorem* for the model parameters $\mathbf{w}$ given the likelihood in combination with some prior information:

$$p(\mathbf{w}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathbf{w})\, p(\mathbf{w})}{p(\mathcal{D})} \tag{2.25}$$

Using domain knowledge regarding protein expression for instance, it is possible to include this within the prior, but we can also use non-informative priors (i.e distributions with large variance). Common choices for prior distributions are:

$$\mathbf{w} \sim \mathcal{N}(\mu_w, \sigma_w^2), \qquad \sigma_w^2 > 0 \tag{2.26}$$

$$\sigma \sim \text{HalfCauchy}(\gamma_\sigma), \qquad \gamma_\sigma \geq 0 \tag{2.27}$$

One of the key advantages with a Bayesian approach is the possibility of modelling the uncertainty surrounding the parameters as well as the response variables themselves. This is straightforward if the conjugate prior of the given distribution is known, else sampling methods such as Markov Chain Monte Carlo (MCMC) deploy finite sampling of the desired posterior.

2.3.3 Regression Trees

Tree-based methods partition the feature space into a set of rectangles (or equivalent higher P-dimensional shape) and fits a simple model in each domain. This is known as the Classification and Regression Tree (CART) methodology [109]. Gradient-boosted trees are considered current state-of-the-art for classical machine learning (excluding deep neural networks) in performance. We make extensive use of GBRT models in this work, for

constructing protein abundance predictors utilising expression data and for SDFs alone.

Let $\mathbf{y} = \{y_n\}$ be a vector of N continuous responses and $\mathbf{X} = \{\mathbf{x}_n\}$ be a $N \times P$ matrix of inputs, where each $\mathbf{x}_p$ is a column vector. We now create a set of M partitions to split the input space domain R until some stopping rule is applied. For a binary tree example see Figure 2.7 where we partition R into 5 domains, where we first split $\mathbf{x}_1 = \theta_1$, then the region $\mathbf{x}_1 \leq \theta_1$ is split at $\mathbf{x}_2 = \theta_2$ and $\mathbf{x}_1 > \theta_1$ is split at $\mathbf{x}_1 = \theta_3$. Finally $\mathbf{x}_1 > \theta_3$ is split at $\mathbf{x}_2 = \theta_4$. Mathematically we could represent the regression model as:

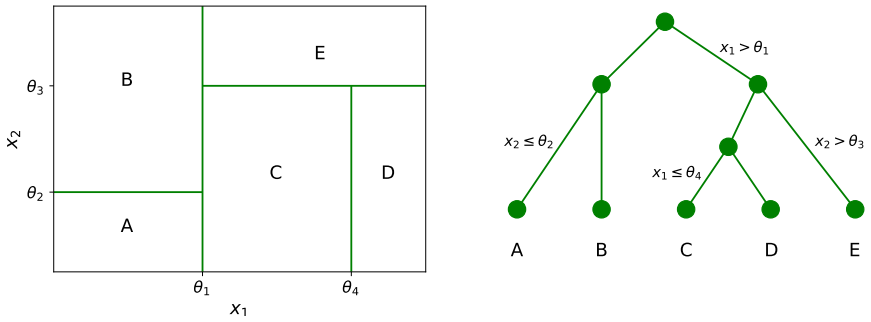


Figure 2.7: ***Decision Tree splits over a domain.*** *Left: Illustration of a two-dimensional input space (x_1, x_2) that is partitioned by 4 parameters $\theta_1, \ldots, \theta_4$. Right: Binary tree corresponding to the partitioning of input space shown in [left]. Note that a parameter is required for each non-leaf node.*

$$f(\mathbf{X}) = \sum_{m=1}^{M} c_m I\{\mathbf{X} \in R_m\} \tag{2.28}$$

where c_m represents the constant for each region, where our criterion is the mean-squared-error function (MSE), c_m is the average over $y_n \in R_m$.

Decision Tree Finding the best binary partition in terms of MSE is computationally infeasible, hence it is common to take a greedy algorithm approach to solving. Considering the splitting variable j and split point s, we define a pair of half-planes:

$$R_1(j,s) = \{\mathbf{X}|\mathbf{X}_j \leq s\} \tag{2.29}$$
$$R_2(j,s) = \{\mathbf{X}|\mathbf{X}_j > s\} \tag{2.30}$$

Then we seek j and s that solve:

$$\min_{j,s}\left[\min_{c_1}\sum_{\mathbf{x}_n\in R_1(j,s)}(y_n - c_1)^2 + \min_{c_2}\sum_{\mathbf{x}_n\in R_2(j,s)}(y_n - c_2)^2\right] \tag{2.31}$$

where best estimates are found as:

$$\hat{c}_m = \frac{1}{N}\sum_{n=1}^{N}[y_n|\mathbf{x}_n \in R_m(j,s)] \tag{2.32}$$

This process is repeated in a recursive fashion onto all resulting regions. The appropriate depth of the tree is a tuning parameter whereby large trees tend to overfit, but small trees may not capture the important structures of the underlying data. A common strategy is to grow a large tree T_0, cease splitting when some minimum node size is reached, and then *pruning* some of the tree branches.

Decision trees are very simple to interpret, scale well to large datasets and also capable of handling categorical input data, but can be prone to overfitting and non-robust, whereby small changes in the input can produce huge changes in tree structure which is inherent within the hierarchical nature of the process; therefore bagging and/or boosting can help to overcome these shortcomings [109, 110].

Boosting To overcome the known problems with base learner decision trees, boosting combines the outputs of many "weak" learners to produce a more powerful committee-based model. Conceptually, this bears a resemblance to bootstrap-aggregating or *bagging* but in the case of boosting, learners are fitted in an additive fashion using elementary basis functions. The most popular algorithm that achieves this is called *AdaBoost* by Friedman et al. [111]:

$$f(\mathbf{X}) = \sum_{m=1}^{M} \beta_m \boldsymbol{\phi}(\mathbf{X}|\gamma_m) \tag{2.33}$$

where β_m are the expansion coefficients, and $\boldsymbol{\phi}(\mathbf{X}|\gamma_m)$ are the basis functions. γ_m for tree models denotes the split variables and split points for internal nodes at node m. Now we will discuss the concept assuming the weak learners are decision tree models, an individual tree can be formally expressed (ignoring basis function transformation) as:

$$T(\mathbf{X}, \boldsymbol{\theta}) = \sum_{j=1}^{J} \gamma_j I(\mathbf{X} \in R_j) \tag{2.34}$$

where we package the parameters $\boldsymbol{\theta}_j = \{R_j, \gamma_j\}$. Here we consider J as a hyperparameter. Thus the parameters are found by minimizing the empirical risk:

$$\hat{\boldsymbol{\theta}} = \arg\min \sum_{j=1}^{J} \sum_{\mathbf{x}_n \in R_j} L(y_n, \gamma_j) \tag{2.35}$$

Note that this is an intensive optimization problem, and as such an appropriate approximation can be found by the following steps:

- **Finding** γ_j. Given R_j, estimating γ_j is trivial as for regression problems $\hat{\gamma}_j \approx (\bar{\mathbf{y}} \in R_j)$, the mean number of observed points falling in region R_j.
- **Finding** R_j. This is more challenging, requiring a greedy top-down recursive partitioning algorithm.

Then the boosted tree model is merely the sum of such weaker trees:

$$f_M(\mathbf{X}) = \sum_{m=1}^{M} T(\mathbf{X}, \boldsymbol{\theta}_m) \tag{2.36}$$

induced in a forward stage-wise manner, whereby at each step we solve:

$$\hat{\boldsymbol{\theta}}_m = \arg\min \sum_{n=1}^{N} L(y_n, f_{m-1}(\boldsymbol{\phi}(\mathbf{x}_n)) + T(\boldsymbol{\phi}(\mathbf{x}_n), \boldsymbol{\theta}_m)) \tag{2.37}$$

Gradient-boosting The additive models as described previously can be subject to numerical optimization via Gradient boosting (GBRT) [112]. This assumes the loss criterion is differentiable. Our goal is to minimize $L(f)$ with respect to f, where $f(\mathbf{X})$ is constrained to be a sum of trees $T(\mathbf{X}, \boldsymbol{\theta})$. By using steepest descent, we define the problem as:

$$f_m = -\rho_m \, \nabla L(f)|_{f=f_{m-1}} \tag{2.38}$$

where ρ_m is the step length. The components of the gradient are:

$$\nabla L(f_n)|_{f=f_{m-1}} = \left[\frac{\partial L(y_n, f(\boldsymbol{\phi}(\mathbf{x}_n)))}{\partial f(\boldsymbol{\phi}(\mathbf{x}_n))}\right]_{f(\boldsymbol{\phi}(\mathbf{x}_n))=f_{m-1}(\boldsymbol{\phi}(\mathbf{x}_n))} \tag{2.39}$$

This can be viewed as a very greedy strategy, with the negative gradient being the local direction for which $L(f)$ is most rapidly decreasing. Below an implementation of the algorithm 1.

The additive nature of gradient-boosting leaves it prone to overfitting, so introducing a regularization parameter by shrinkage [114] on the learning rate helps to reduce this:

$$f_m(\mathbf{X}) = f_{m-1}(\mathbf{X}) + \nu \, T(\mathbf{X}, \theta_{jm}) \tag{2.40}$$

where $0 < \nu < 1$ scales the contribution of each tree. Shrinkage has been demonstrated to improve generalization, at the cost of increasing computational time due to requiring more iterations. It is common to trade-off ν against the number of weak learners M.

2.3.4 Covariance and Correlation

Here we will cover a number of the key methods we use for analysing biological features and the relationships between them. Correlation analysis is prevalent in this research, to analyse how useful each feature is, since a lot of work has gone into feature engineering and selection to find biological features. Particularly as a number of the biological features display multicollinearity (such as length, base pair counts), partial correlations are needed to eliminate that dependency wherever possible.

Algorithm 1: Gradient-Tree Boosting Algorithm [112, 113]

Result: Output $\hat{f}(\mathbf{X}) = f_M(\mathbf{X})$.
Initialize $f_0(\mathbf{X}) = \arg\min_\gamma \sum_n^N L(y_n, \gamma)$;
for $m = 1, \ldots, M$ **do**

1. For $n = 1, \ldots, N$ compute

$$r_{nm} = - \nabla L(f_n)|_{f=f_{m-1}}$$

2. Fit a regression tree to the targets r_{nm} giving terminal regions R_{jm}, $j = 1, \ldots, J_m$.

3. For $j = 1, \ldots, J_m$ compute

$$\gamma_{jm} = \arg\min_\gamma \sum_{\mathbf{x}_n \in R_{jm}} L(y_n, f_{m-1}(\mathbf{x}_n) + \gamma)$$

4. Update $f_m(\mathbf{X}) = f_{m-1}(\mathbf{X}) + T(\mathbf{X}, \theta_{jm})$

;
end

Covariance Covariance is a measure of joint variability between two variables. Formally, between two distributed real-valued variables $\mathbf{x}$ and $\mathbf{y}$ it exists as the expected product of their deviations from the expected value [115]:

$$\begin{aligned} \operatorname{cov}(\mathbf{x}, \mathbf{y}) &= \mathbb{E}\left[(\mathbf{x} - \mu_{\mathbf{x}})(\mathbf{y} - \mu_{\mathbf{y}})\right] && (2.41) \\ &= \mathbb{E}\left[\mathbf{x}\mathbf{y}\right] - \mathbb{E}\left[\mathbf{x}\right]\mathbb{E}\left[\mathbf{y}\right] && (2.42) \end{aligned}$$

where $\mu_{\mathbf{x}}$ and $\mu_{\mathbf{y}}$ refers to the means of $\mathbf{x}$ and $\mathbf{y}$ respectively. For a given design matrix $\mathbf{X}$, the column vectors $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p\}^{\mathrm{T}}$ are assumed to be random variables with finite variance and expected value. The covariance matrix $\boldsymbol{\Sigma}$ [116] is the matrix whose (i, j) entries correspond to covariance between features i and j:

$$\boldsymbol{\Sigma} = \begin{bmatrix} \operatorname{var}(\mathbf{x}_1) & \operatorname{cov}(\mathbf{x}_1, \mathbf{x}_2) & \ldots & \operatorname{cov}(\mathbf{x}_1, \mathbf{x}_p) \\ \operatorname{cov}(\mathbf{x}_2, \mathbf{x}_1) & \operatorname{var}(\mathbf{x}_2) & \ldots & \operatorname{cov}(X_2, \mathbf{x}_p) \\ \vdots & \vdots & \ddots & \vdots \\ \operatorname{cov}(\mathbf{x}_p, \mathbf{x}_1) & \operatorname{cov}(\mathbf{x}_p, \mathbf{x}_2) & \ldots & \operatorname{var}(\mathbf{x}_p) \end{bmatrix} \tag{2.43}$$

where $\operatorname{var}(\mathbf{x}_i)$ is a function to estimate the *variance* of a vector. It follows from definition that the covariance matrix has the following properties [115], such as positive-semidefinite and symmetry. The inverse of $\boldsymbol{\Sigma}^{-1} = \boldsymbol{\Lambda}$ is known as the *precision matrix*. The covariance can be estimated directly via Maximum Likelihood (ML) as:

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{N-1} \sum_{n=1}^{N} (\mathbf{x}_n - \boldsymbol{\mu}_{\mathrm{ML}})(\mathbf{x}_n - \boldsymbol{\mu}_{\mathrm{ML}})^{\mathrm{T}} \tag{2.44}$$

where $\boldsymbol{\mu}_{\mathrm{ML}}$ is the maximum likelihood for the expected value. Note that this is an unbiased estimator of covariance using $N - 1$ instead of N.

Correlation Describing the statistical dependence between bivariate data, normalized covariance or *correlation* is one of the most frequently used statistical metrics. The most common metric is *Pearson product-moment correlation* coefficient [118]. When applied to a population of random variables $\mathbf{x} \in \mathbb{R}^N$ and $\mathbf{y} \in \mathbb{R}^N$, *Pearson's* coefficient is represented traditionally as ρ, and is mathematically formulated as:

$$\rho(\mathbf{x}, \mathbf{y}) = \frac{\mathbb{E}\left[(\mathbf{x} - \mu_{\mathbf{x}})\,(\mathbf{y} - \mu_{\mathbf{y}})\right]}{\sigma_{\mathbf{x}}\sigma_{\mathbf{y}}} \tag{2.45}$$

where $\mu_{\mathbf{x}}$ and $\mu_{\mathbf{y}}$ are the mean expectation values for the respective vectors. When applied to samples, which we denote $r(\mathbf{x}, \mathbf{y})$, the coefficient can be calculated by estimating the covariance and variance of the samples, given paired data $\{(x_1, y_1), \ldots, (x_N, y_N)\}$ consisting of N pairs, the estimated correlation is given [118] as:

$$r(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^{n}(x_i - \bar{\mathbf{x}})\,(y_i - \bar{\mathbf{y}})}{(N-1)s_{\mathbf{x}}s_{\mathbf{y}}} \tag{2.46}$$

$$= \frac{N\sum_i x_i y_i - \sum_i x_i \sum_i y_i}{\sqrt{N\sum_i x_i^2 - (\sum_i x_i)^2}\sqrt{N\sum_i y_i^2 - (\sum_i y_i)^2}} \tag{2.47}$$

where $\bar{\mathbf{x}}$ is the sample mean and $s_{\mathbf{x}}$ is the sample standard deviation. Spearman-rank correlation performs this correlation, but on the *rank variables*, not the raw scores. Closely resembling the covariance matrix $\mathbf{\Sigma}$, the *correlation matrix*, or the *Pearson product-moment correlation* coefficients $\mathbf{R}$ between each random variable in vectors $\mathbf{x}_i$ is normalized as:

$$\text{corr}(\mathbf{X}) = \mathbf{R} = \mathbf{\Sigma}^{-\frac{1}{2}}.\mathbf{\Sigma}.\mathbf{\Sigma}^{-\frac{1}{2}} \tag{2.48}$$

where the diagonal matrix is given as

$$\mathbf{\Sigma}^{-\frac{1}{2}} = \text{diag}\left(\frac{1}{\sqrt{\Sigma_{11}}}, \ldots, \frac{1}{\sqrt{\Sigma_{pp}}}\right) \tag{2.49}$$

The diagonal elements $\mathbf{R}_{i=j} = 1$, with each off-diagonal element in the range $-1 \leq \mathbf{R}_{i \neq j} \leq 1$. Unlike covariance, the scale of difference between features is normalized for correlation allowing unbiased comparisons [116]. The coefficient of determination, r^2 is simply the power of this coefficient, and in addition an adjustment is commonly added given the metrics propensity to increase in multi-dimensional situations:

$$r^2_{\text{adj}} = 1 - (1 - r^2)\frac{N-1}{N-P-1} \tag{2.50}$$

We make use of correlation matrices extensively in this work to analyse the inter-relationships between similar and distant biological features.

Point-biserial Correlation Closely related to the two-sample unpaired Student's T hypothesis test, this metric is used to calculate the correlation between one continuous random variable $\mathbf{x} \in \mathbb{R}^N$ and one dichotomous variable $\mathbf{y} \in \mathbb{Z}^N$ where values of $y_n \in [0, 1]$. We partition observed samples $\mathcal{D} = \{\mathbf{x}, \mathbf{y}\}_{n=1}^N$ into groups

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_0 | y = 0 \\ \mathbf{x}_1 | y = 1 \end{pmatrix} \tag{2.51}$$

where $\bar{\mathbf{x}}$ is the sample mean. Let N_0 is the number of samples in $\mathbf{x}_0$, with N_1 for $\mathbf{x}_1$, then the sample correlation coefficient becomes:

$$r_{pb}(\mathbf{x}, \mathbf{y}) = \frac{\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_0}{s_{\mathbf{x}}} \sqrt{\frac{N_1 N_0}{N(N-1)}} \tag{2.52}$$

where $s_{\mathbf{x}}$ is the sample (unbiased) standard deviation over $\mathbf{x}$.

Partial Correlation The degree of association between two variables can be further analysed by eliminating M set of controlling variables $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_M\}$, written as $\rho(\mathbf{x}, \mathbf{y}|\mathbf{Z})$ [119]. Like the correlation, it takes values in the range $[-1, 1]$, and from a probabilistic standpoint can be viewed as a *conditional* correlation between two jointly distributed random variables. There are two main ways of deriving the partial correlation:

1. Using **linear regression**: To find $r(\mathbf{x}, \mathbf{y}|\mathbf{Z})$ where vectors $\mathbf{x} \in \mathbb{R}^N$, $\mathbf{y} \in \mathbb{R}^N$ and matrix $\mathbf{Z} \in \mathbb{R}^{N \times M}$, compose two linear regression models $\mathcal{E}_{\mathbf{x}|\mathbf{Z}}$ and $\mathcal{E}_{\mathbf{y}|\mathbf{Z}}$ (see notation in ordinary least squares 2.3.2) yielding residual vectors $\boldsymbol{\epsilon}_{\mathbf{x}}$ and $\boldsymbol{\epsilon}_{\mathbf{y}}$ and thus we compute the sample Pearson correlation $r(\boldsymbol{\epsilon}_{\mathbf{x}}, \boldsymbol{\epsilon}_{\mathbf{y}})$ with respect to the residuals.

2. Using **matrix inversion**: Estimate the precision matrix $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$ via the covariance $\mathbf{S} = \text{cov}(\mathbf{X})$ using MLE, and/or shrinkage techniques, then normalize using:

$$r(\mathbf{x}_i, \mathbf{x}_j | \mathbf{Z}_{i \neq j}) = -\frac{\Lambda_{ij}}{\sqrt{\Lambda_{ii}\Lambda_{jj}}} \qquad \forall i, j,\ i \neq j \tag{2.53}$$

where $i, j = 1, \ldots, P$, $\mathbf{x}_i$ and $\mathbf{x}_j$ are column-vectors and $\mathbf{Z}_{i \neq j}$ refers to all remaining columns not selected by either i or j. We can think of this as the normalized precision matrix.

There are a number of techniques to compute the precision matrix if the covariance matrix is ill-conditioned, these methods are beyond the scope of this Thesis. The primary advantages of using partial correlation is that the additional variables can help to control the confounding variable of interest. We utilize this approach when analysing the dependencies between our extracted SDFs, to help eliminate multicollinearity effects within our modelling.

2.3.5 Dimensionality Reduction

In practice, it is rare that the number of dimensions of a given dataset accurately corresponds to the true *degrees of freedom* (DOF) of variability. For example, within molecular biology it is unlikely that all of our sequence-extracted features will affect protein abundance or function. Furthermore, correlations between these features can produce an ill-conditioned matrix that is not full rank. Thus we want to deploy a technique which reduces $P \rightarrow K$ substantially whilst preserving the variability and condensing it into a subspace K. Thus our reasons to reduce the dimensions are as follows:

- Solve the $P \gg N$ problem with respect to matrix ill-conditioning
- Reduce impact of curse of dimensionality with respect to algorithms that scale $\mathcal{O}(P^2)$ or higher
- Find true subspace of DOF variability where significant redundancy exists
- Reduce computation time

In this work we will consider the most classical approach using Principle component analysis (PCA), and variants such as Probabilistic PCA (PPCA) and Factor Analysis. These methodologies mainly come into play within section 4 of the results section when we apply unsupervised learning to our SDF set, with a thorough analysis therein.

Principle component analysis (PCA) Also known as Karhunen-Loève transform, PCA is widely used for dimensionality reduction, lossy compression, feature extraction and data visualization [120]. Conceptually, PCA provides an orthogonal projection of the input data onto a lower dimensional linear space, known as the *principle subspace*, such that variance of the pro-

jected data is maximized [121].

Let $\mathcal{D}|\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ be a set of training examples with dimensionality P. Our goal is to project this onto a space having dimensionality $K < P$ while maximizing the variance of the projected data. The value K in classical PCA is a hyperparameter that is defined by the user, for data visualization $K = 2$ is usually chosen, but K can vary for machine learning purposes. Here we will consider the example of where we project onto a 1-dimensional space $K = 1$, but the concept extends up to $K = P$. We define the direction of the subspace using a vector $\mathbf{u}_1 \in \mathbb{R}^P$ which is also a unit vector such that $\mathbf{u}_1^{\mathrm{T}}\mathbf{u}_1 = 1$, thereby disregarding vector magnitude. Each data point is then projected onto a scalar value $\mathbf{u}_1^{\mathrm{T}}\mathbf{x}_n$, where the variance of the projected data is given as:

$$\max \frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{u}_1^{\mathrm{T}}\mathbf{x}_n - \mathbf{u}_1^{\mathrm{T}}\bar{\mathbf{x}}\right)^2 = \mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1 \qquad \text{s.t. } \mathbf{u}_1^{\mathrm{T}}\mathbf{u}_1 = 1 \tag{2.54}$$

where $\bar{\mathbf{x}}$ is the sample mean and $\mathbf{S}$ is the sample covariance matrix. To enforce the normalization constraint, we introduce a Lagrange multiplier λ_1 (not to be confused with precision), then our maximization becomes:

$$\mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1 + \lambda_1(1 - \mathbf{u}_1^{\mathrm{T}}\mathbf{u}_1) \tag{2.55}$$

If we solve the derivative of 2.55 with respect to $\mathbf{u}_1$ equal to zero, the solution is:

$$\mathbf{S}\mathbf{u}_1 = \lambda_1\mathbf{u}_1 \tag{2.56}$$

where λ_1 is an eigenvalue, and $\mathbf{u}_1$ must be an eigenvector with respect to $\mathbf{S}$. By multiplying $\mathbf{u}_1^{\mathrm{T}}$ and using the normalization constraint, we can see that the eigenvalue represents the maximum variance of the direction:

$$\lambda_1 = \mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1 \tag{2.57}$$

$\mathbf{u}_1$ corresponds to the first principle component. Additional principle components can be defined in incremental fashion by repeating the above process, generating $\mathbf{u}_1, \ldots, \mathbf{u}_M$ eigenvectors with corresponding $\lambda_1, \ldots, \lambda_M$ eigenvalues. See Figures 2.8 and 2.9 for illustrations of how PCA achieves this on synthetic and real-world datasets.

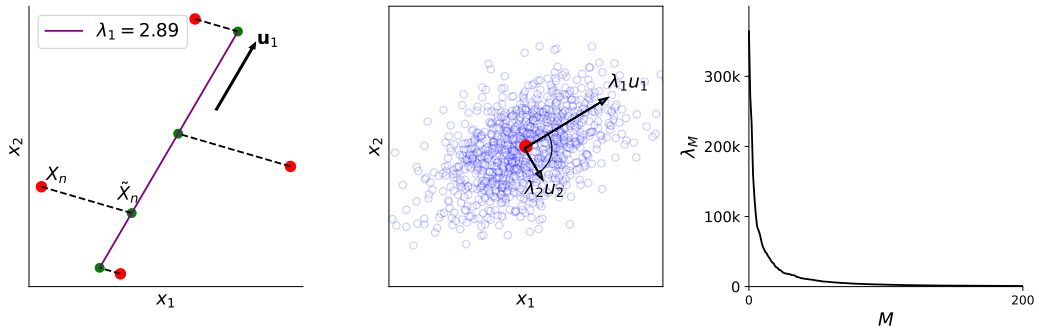


Figure 2.8: *PCA illustrations.* *Left: Illustration of* $\mathbf{x}_n \rightarrow \tilde{\mathbf{x}}_n$ *mapping given toy 2D input space mapped to 1D. Middle: PCA on 2D multivariate Gaussian distribution* $\mathcal{N}(\mu, \Sigma)$*, input points are blue, with optimal projections* $\lambda_1 \mathbf{u}_1$ *and* $\lambda_2 \mathbf{u}_2$ *shown as arrows (inc. magnitude). Eigenvectors are orthogonal (as shown by 90 degree angle). Red point indicates data mean. Right: Eigenvalues* λ_i *against* $i = 1, \ldots, M$ *indicating the fall-off in variance preserved as* M *increases, used on the MNIST dataset. We only show the first* $M = 200$ *as values of* $\lambda_M \rightarrow 0$ *which distorts the figure.*

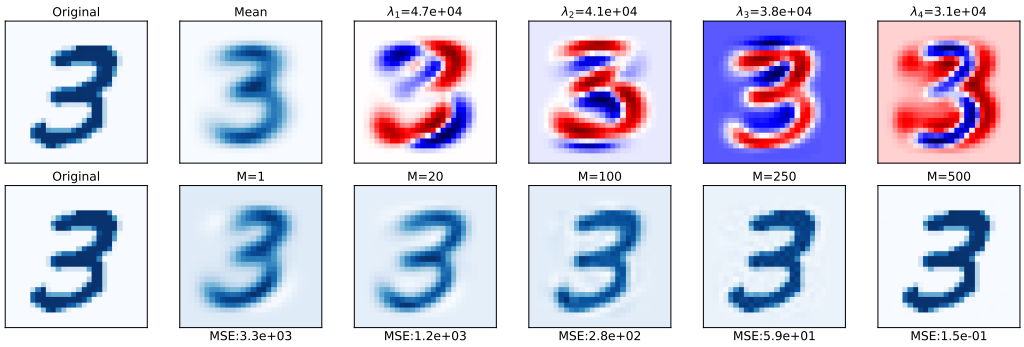


Figure 2.9: ***PCA illustrations on the MNIST dataset.*** *Top row: The original, the mean vector* $\bar{\mathbf{x}}$ *along with the first 4 eigenvectors* $\mathbf{u}_1, \ldots, \mathbf{u}_4$ *for a digit three from the MNIST dataset. Corresponding eigenvalues described as title. Red corresponds to positive values, blue corresponds to negative values. Bottom row: The original, followed with* $\tilde{\mathbf{X}}$ *reconstructions for* $M = 1, 20, 100, 250, 500$. *As* M *increases the accuracy of the reconstruction improves. MSE of reconstructions is displayed underneath.*

PCA can often be used as a preprocessing step to subsequent modelling via classification or regression, because the transformation can standardize certain properties of the input matrix. To achieve this, we generalize equation 2.56 to get:

$$\mathbf{SU} = \mathbf{\Lambda U} \tag{2.58}$$

where $\mathbf{\Lambda}$ is a $P \times P$ diagonal matrix with elements λ_i. Then we define a transformation for each data point $\mathbf{x}_n$ as:

$$\mathbf{y}_n = \mathbf{\Lambda}^{-1/2}\,\mathbf{U}^{\mathrm{T}}(\mathbf{x}_n - \bar{x}) \tag{2.59}$$

the set $\{\mathbf{y}_n\}$ thus has zero mean and the covariance is equivalent to the identity matrix. This process is known as *whitening* or normalization.

Probabilistic PCA In this approach, PCA can be modelled as a maximum likelihood solution of a probabilistic latent variable model rather than merely a linear projection of data points onto a subspace [123, 122]. in PPCA all of the marginal and conditional distributions are Gaussian, thus we introduce the latent variable $\mathbf{z}$ corresponding to the principle component subspace, and the prior distribution $p(\mathbf{z})$ over the latent variable, with $p(\mathbf{x}|\mathbf{z})$ acting as the Gaussian conditional distribution for the observed data. Then we can write the prior and likelihood distributions as:

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|0, \mathbf{I}) \tag{2.60}$$

$$p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2\mathbf{I}) \tag{2.61}$$

where $p(\mathbf{z})$ is governed by a zero-mean unit-covariance Gaussian, and $p(\mathbf{x}|\mathbf{z})$ is dependent on loading matrix $\mathbf{W} \in \mathbb{R}^{P\times K}$ and $\boldsymbol{\mu}$ (which can be removed via centering). The columns in $\mathbf{W}$ span a linear subspace within the data space which corresponds to the principle subspace. Maximum likelihood with respect to $\mathbf{W}$ and σ^2 are relatively straightforward and require an eigendecomposition of the sample covariance matrix:

$$\hat{\mathbf{W}} = \mathbf{U}(\mathbf{L} - \sigma^2\mathbf{I})^{1/2}\mathbf{R} \tag{2.62}$$

$$\hat{\sigma}^2 = \frac{1}{P-K}\sum_{i=K+1}^{P} \lambda_i \tag{2.63}$$

where $\mathbf{U} \in \mathbb{R}^{P \times M}$ is an eigenvector matrix whose columns are eigenvectors of $\mathbf{S}$, $\mathbf{\Lambda}$ is the diagonal matrix (see eq 2.58) whose values are λ_i, and $\mathbf{R}$ is an orthogonal rotation matrix. These matrices can be determined using singular value decomposition (SVD) and there are efficient computational tools to do so. These estimates can be inserted into the posterior distribution $p(\mathbf{z}|\mathbf{x})$ to generate a distribution over our latent variables.

Factor Analysis As described with regards to mixture models, they only use a single latent variable to generate the observations, i,e there is a one-to-one mapping between latent variable and observed variable. An alternative is to use a vector of real-valued latent variables $\mathbf{z}_n \in \mathbb{R}^L$, where we can use a Gaussian prior and likelihood function [122, 162] as:

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_0, \mathbf{\Sigma}_0) \tag{2.64}$$
$$p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \mathbf{\Psi}) \tag{2.65}$$

where $\mathbf{W} \in \mathbb{R}^{P \times L}$ is the factor loading matrix and $\mathbf{\Psi}$ is a $P \times P$ diagonal covariance matrix. Notice that the likelihood function is very similar to PPCA, the only major difference being the presence of a non-unit covariance matrix. Hence the special case $\mathbf{\Psi} = \sigma^2\mathbf{I}$ is equivalent to PPCA.

2.3.6 Graphical Models

Thus far the models previously described assume the training set $\mathcal{D}$ is i.i.d (independent and identically distributed), whereas often there can exist observed and latent interactions between inputs [122]. Let $\mathbf{v}$ be the set of N unique *vertices* within a directed graphical model, and $\mathbf{E} \in \mathbb{R}^{N \times N}$ be the observed adjacency matrix of directed edge interactions between vertices such that $E_{ij} \in [0, 1]$ is the strength of interaction going from vertex $v_i \rightarrow v_j$ to vertex, with $E_{ij} = 0$ indicating no edge connection. Note that in a directional graph $E_{ij} \neq E_{ji}$, hence $\mathbf{E}$ is not symmetric. A graph with no interactions is simply diagonal $E_{ij} = 0$, $\forall i \neq j$, whereas non-zero diagonal elements are loops. The degree of a given vertex [179] is given as the sum of non-zero directional edges emanating from it:

$$D(v_n) = \sum_{i=1}^{N} \mathbb{I}\left[E_{ni} > 0\right] \tag{2.66}$$

Another important metric is derived from the concept of *centrality*, which estimates the importance of a given vertex v_n within the wider graph. A simple approach to centrality is by normalising the degree by all other degrees in the network:

$$C_D(v_n) = \frac{D(v_n)}{\sum_{i=1}^{N} D(v_i)}, \quad i \neq n \tag{2.67}$$

A more complicated but realistic metric of centrality can be derived via eigendecomposition of $\mathbf{E}$ [179], which we will not go into detail here. A degree-based centrality can over-prioritise vertices with a large number of connections without accounting for the strength of interactions. The idea is to utilise the eigenvalues and eigenvectors derived from $\mathbf{E}$ directly as an indicator of vertex importance.

In this Thesis we construct a directed graphical model of protein-protein interactions (PPI) for the purposes of computing additional metrics to insert into our sequence-derived feature set in Chapter 5, including degree and centrality as mentioned above.

2.4 Transcriptome-Proteome Analysis

We now turn to the integrated analysis of transcriptome and proteome measurements.

2.4.1 Correlation of the Transcriptome-Proteome

Many previous authors have looked into the relationship between mRNA and protein abundance [126, 127, 128, 129], particularly within prokaryotic organisms such as yeast. In particular, the early focus lay on using the relationship to explain related cellular functionalities. We will not cover every possible paper produced on this topic, but instead focus on key narrative moments in the understanding of transcriptome-proteome dynamics. Greenhaum et al (2002) [127] compiled together several mRNA expression datasets and found several particular amino acids enriched, along with functional terms such as 'protein synthesis' and 'energy production'. There was also early recognition in the part that translation (or the 'translatome') had to play within their analysis. It has also previously been a common assumption to use mRNA expression

as a proxy for protein abundance [129], given the theoretical assumptions underpinned by the Central Dogma. Beyer et al. (2004) [129] uses *S. cerevisiae* microarray and protein assay data to uncover post-transcriptional regulation pathways by looking at the transcriptome-proteome relationship, and inferring translation as a combination of Ribosome density and occupancy. They also confirm that mRNA-protein correlations on a genomic scale either are statistically insignificant or are weak. This paradox holds for all species studied, and exacerbates for higher-order organisms such as *H. sapiens*. They also introduce a Protein Half-Life Descriptor: a first-order differential equation in the form:

$$\frac{\mathrm{d}[P_i]}{\mathrm{d}t} = k_p \cdot k_{\text{trans}} \cdot [mRNA_i] - k_{d,i} \cdot [P_i] \tag{2.68}$$

where $[mRNA_i]$ and $[P_i]$ are the mRNA and protein abundances of the ith ORF, respectively, with k_p being the elongation speed, k_{trans} as the ribosome occupancy and $k_{d,i}$ being the ORF destruction rate. This attempts to model the protein half-life, another interesting aspect of proteomics which is heavily associated with post-translational processing and protein stability. Correlation values vary according to study, with Beyer et al (2004) reporting a Spearman-rank correlation ($r_s = 0.58$), Futcher et al (1999) [126] report a very high correlation ($r^2 = 0.76$) after transforming data into normal distributions, with Greenbaum et al (2003) [128] review paper reporting a modest ($r = 0.66$, $N = 2044$) correlation in addition to analysing various functional subsets. The differences within statistical techniques used to analyse the data was largely responsible for differing conclusions with respect to the different papers. Greenbaum [127, 128] also began to explore the impact of codon bias through the Codon Adaptation Index (CAI) feature with mRNA and protein abundance, but found there was significant correlation between genes with high CAI and mRNA/protein abundance. Wu et al (2008) [130] further extended this analysis on *S. cerevisiae* cells by incorporating direct mRNA and protein half-life information using the following multiple linear regression model (eq 2.8):

$$y_i = \alpha + [\text{mRNA}]_i \, \beta + \sum_{j=1}^{m} \beta_j x_{ij} \tag{2.69}$$

where α is the intercept, β is the weighting of mRNA abundance, and $\beta_j x_{ij}$ corresponds to the i-th predicted value and the j-th sequence covariate.

Organism (Species)	$\mathbf{r}_p$	$\mathbf{r}_s$	N
S. cerevisiae (1)	0.36	NA	73
S. cerevisiae (2)	0.76	0.74	148
M. musculus	0.59	NA	425
S. cerevisiae (3)	NA	0.45	678
D. vulgaris	0.5	NA	703
E. coli	0.57	0.5	1103
S. pombe	0.58	0.61	1367
S. cerevisiae (4)	0.66	NA	2044
S. cerevisiae (5)	NA	0.57	5251

Table 2.6: *Overview of mRNA-protein correlation studies within different organisms.* *r_p represents Pearson's correlation, r_s is Spearman-rank correlation. Table taken from Maier et al. [131].*

Wu (2008) places explanatory emphasis on the capacity of protein half-life and translation elongation, rather than translation initiation or mRNA half-life in explaining mRNA-protein correlation. Note however that they did not attempt to predict the protein abundance; merely to model the correlation. See Table 2.6 for a comparison by Maier et al [131] regarding mRNA-protein abundance correlation estimates across multiple studies for different species up to 2009.

One of the first attempts to compare RNA sequence and microarray data with label-free protein data is by Ning et al (2012) [132] (see Figure 2.10 for examples). They also made use of DAVID [133] GO enrichment analysis to determine biological functionalities for given RNA subsets. They found that the correlation between mRNA and protein for genes associated with ribosomal activities were not strong, due to post-translational activities such as phosphorylation, acetylation and methylation. Such PTMs modify the protein degradation rate which impact on the correlation towards mRNA abundance. The authors also posit a robust computational framework for gene-protein data interactions for future studies.

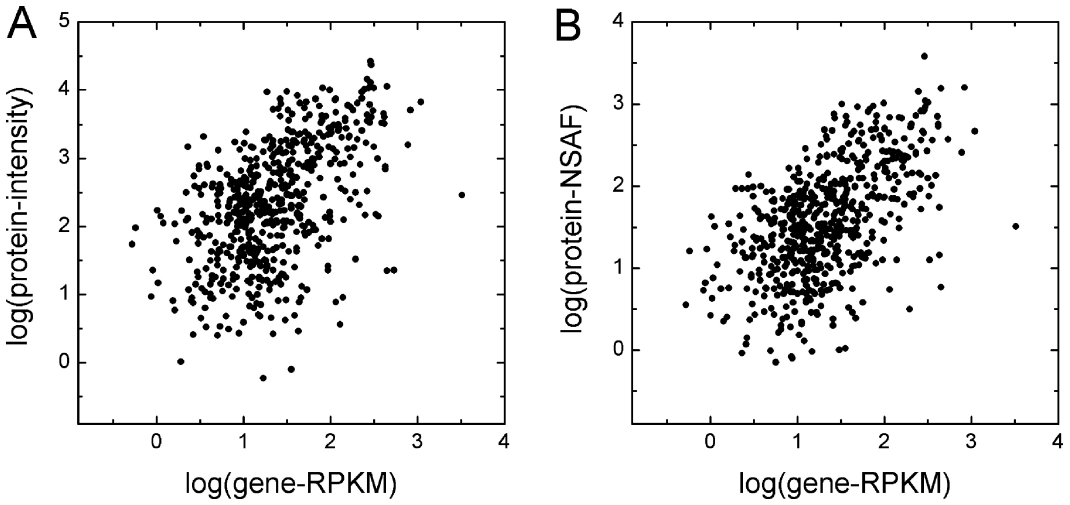


Figure 2.10: Scatterplots of log **mRNA abundance against protein intensity.** *mRNA is normalized using RPKM, A) represents protein-intensity values, B) represents NSAF normalization. Figure taken from Abstract of Ning et al (2012) [132].*

2.4.2 Modelling of the Transcriptome-Proteome

So far in the transcriptome-proteome interface, we have mostly focused on correlation between mRNA and protein, and inferring causal properties by subsequent GO analysis on interesting highly-correlated subgroups, or by subsequent experimental validation methods that look for biological mechanisms. Now we're going to look at more recent data-driven approaches to exploring the transcriptome-proteome interface. As noted by Vogel (2010) [6], the protein-mRNA relationship is non-linear but can be approximated well by a piece-wise linear function, and where concentrations are log-normally distributed, indicating that they can be modelled using multiplicative independent random variables [134].

Classification Approach Pancaldi and Bähler (2011) [135] used Support Vector Machines (SVM) and Random Forests (RF) to classify RNA-binding proteins (RBPs) and mRNA abundance interactions within *S. cerevisiae*. They used a significant number of input features ($P > 100$) including protein localization, GO and genetic interaction properties (from BioGRID) aid in correlation analysis. They achieved 70% accuracy score with 2-fold CV using RF and 68% with a radial-basis SVM classifier. They however faces challenges by the limited amount of experimental data, which is characteristic of attempting to combine together large amounts of input from multiple sources. We will spend some time attempting to overcome similar problems within this thesis.

Bayesian Approach Another way of modelling the relationship is using a Bayesian probabilistic approach, as adopted by Kannan et al (2007) [136], where microarray and MS measurements of *M. musculus* are provided as evidence to a probabilistic model, whereby for each gene $g = 1, \ldots, G$, peptide counts $y_i^{(g)}$ are estimated using mRNA expression $m_i^{(g)}$ and an 'average' protein expression $x_i^{(g)}$. Since peptide counts are integers, they used the independent Poisson distribution to model as:

$$p(\mathbf{y}|\mathbf{x}) = \prod_{i=1}^{T} \frac{e^{-x_i} x_i^{y_i}}{y_i!} \tag{2.70}$$

where T is the number of tissues. Here the average protein expression is

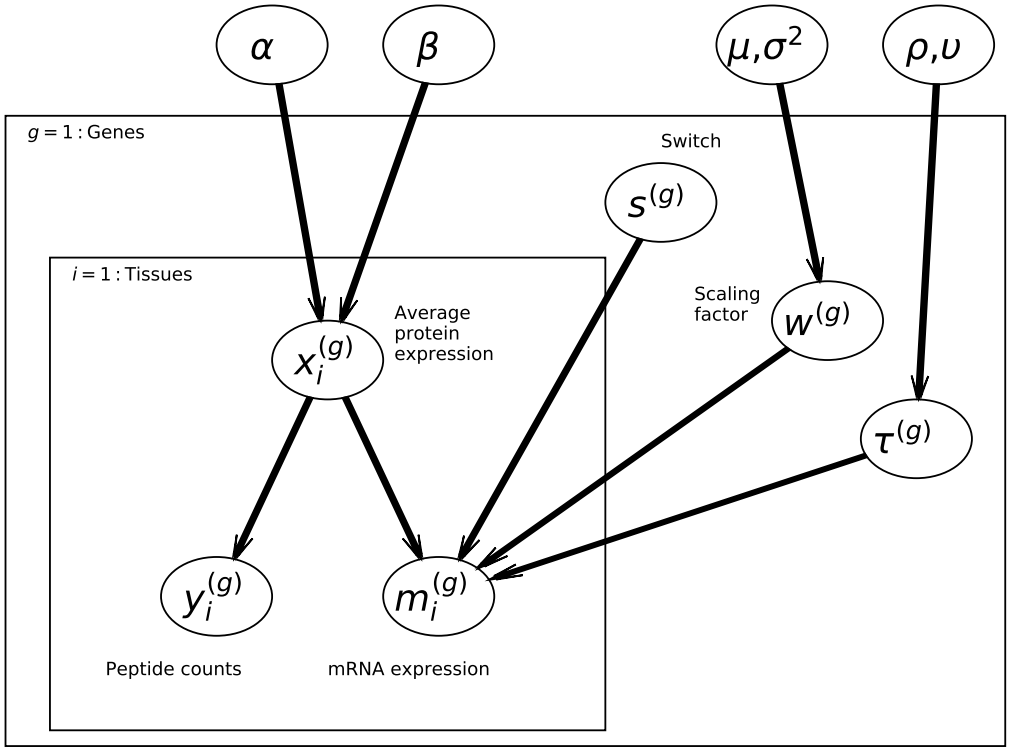


Figure 2.11: Bayesian network modelling of peptide counts using mRNA and protein expression levels. *This network is taken from Kannan et al (2007) [136], inner rectangle represents a single gene g and shares s,w and τ variables.*

used as a rate parameter for the Poisson model. Following that the Gamma distribution is the conjugate prior of the Poisson rate parameter, we get:

$$p(\mathbf{x}) = \prod_{i=1}^{T} \frac{\beta^{\alpha}}{\Gamma(\alpha)} x_i^{\alpha-1} e^{-x_i\beta} \tag{2.71}$$

where α and β are hyperpriors of the Gamma distribution, and $\Gamma(\alpha)$ is the Gamma function (not equivalent to the distribution). From these the posterior distribution of peptide counts can be computed, with uncertainty measurements regarding mRNA and average protein expression. See Figure

2.11 for the Bayesian graph representation of the models fitted. This technique diverges from previous approaches in that the data noise is assumed to be drawn from a non-Gaussian distribution (represented by τ), over 6 main tissue types (brain, heart, kidney, liver, lung and placenta). s represents a Bernoulli switch variable, where $s = 1$ then the noise is modelled as a linear function of average peptide counts, as a Gaussian $\mathcal{N}(0, \tau)$. The joint distribution over the variables modelled is:

$$p(\mathbf{x}, \mathbf{y}, \mathbf{m}, \theta, s) = p(\mathbf{x})p(\mathbf{y}|\mathbf{x})p(\mathbf{m}|\mathbf{x}, s, \theta)p(\theta)P(s) \tag{2.72}$$

where $\theta = (w, \tau)$. The authors do not aim to maximize the property $p(\mathbf{m}, \mathbf{y})$ as the model would also try to account for tissue specificity and gene function. Hence the aim is to maximize the conditional distribution $p(\mathbf{m}|\mathbf{y})$ by integrating out the hidden variables:

$$p(\{\mathbf{m}^{(g)}, \mathbf{y}^{(g)}\}) = \int_{\theta} \prod_{g=1}^{G} \sum_{s} P(s^{(g)}) \int_{\mathbf{x}} p(\mathbf{y}^{(g)})p(\mathbf{x}^{(g)}|\mathbf{y}^{(g)})p(\mathbf{m}^{(g)}|\mathbf{x}^{(g)}, s^{(g)}, \theta)p(\theta) \tag{2.73}$$

where $p(\mathbf{x}^{(g)}|\mathbf{y}^{(g)})$ is the Gamma posterior distribution of average protein expression. For a given gene, the relationship strength between mRNA and protein abundance is given by $P(s|\mathbf{m}, \mathbf{y})$. This can be computed using Bayes' rule:

$$P(s|\mathbf{m}, \mathbf{y}) = \frac{\int_{\mathbf{x}} p(\mathbf{m}|s, \mathbf{x})p(\mathbf{x}|\mathbf{y})P(s)}{\sum_{s} \int_{\mathbf{x}} p(\mathbf{m}|s, \mathbf{x})p(\mathbf{x}|\mathbf{y})P(s)} \tag{2.74}$$

hence the linear relationship between the measurements is given as $P(s = 1|\mathbf{m}, \mathbf{y})$. Subsequent analysis of GO annotations on various group partitions found that outliers were highly linked to their respective tissue functions.

Regression Approach Instead of a classification approach, one can predict the protein abundance directly using a regression-based approach. Tuller et al (2007) [137] developed a linear regression model incorporating mRNA expression in conjunction with sequence-derived features (SDFs) such as tRNA adaptation inde (tAI) and evolutionary rate (ER), in *S. cerevisiae*. The full feature set explored by Tuller include:

- Protein Molecular weight, Length, GRAVY and aromaticity

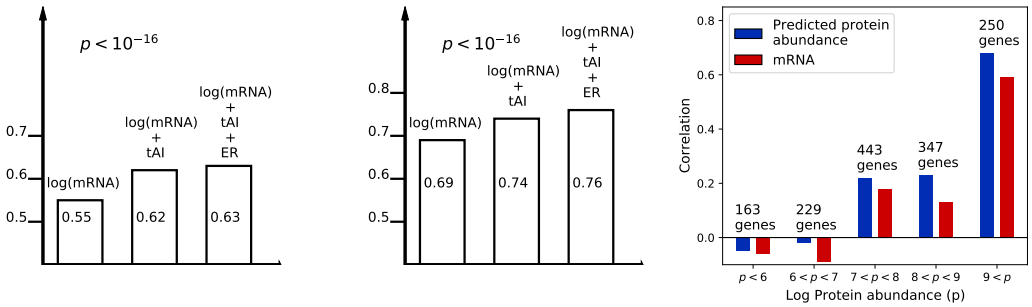


Figure 2.12: Performances of the Linear Predictor on log Protein abundance. *Left and middle: Accuracies of various linear predictors using Spearman rank correlation r_s, with inclusion of features using greedy forward feature selection. Right: Correlations of predicted (mRNA) vs. actual protein abundance binned at various levels of actual protein abundance. Graph recreated with small adaptations from Tuller et al (2007) [137].*

- Frequency of every amino acid (ACDEFGHIKLMNPQRSTVWY).
- Protein half-life
- Translation efficiency (TE), ER, tAI, CAI and codon bias

In order to discover which features were the most important, they used Greedy Forward Feature Selection (GFFS) which additively builds more complex models by selecting the model whose feature improves the correlation score between the predicted and target values [137] (see Figure 2.12 for details). They found that tAI and ER were the next 2 best features to be added to a linear predictor, following log mRNA expression levels. However this method is prone to statistical bias, as Spearman-rank correlation and $r^2 \to 1$ as $P \to \infty$, leading to a problem with overfitting if not careful. This can be overcome by using an adjusted r^2 metric as we mention previously (2.3.4). Interestingly they found that non-linear predictors such as radial-basis SVM did not significantly improve prediction, and hence linear predictors do seem to capture a significant portion of explanatory power. Tuller achieved a correlation of $r_s = 0.76$ for averaged data, but did not

extend this concept to look for model failures (i.e outliers) and their significance. Tuller goes on to focus on translational efficiency and its relationship to codon bias and folding energy in subsequent research [53].

The relationship between sequence-derived features and protein abundance is more thoroughly established by the work of Vogel et al (2010) [6]. They analyze Daoy medulloblastoma cell cultures from *H. sapiens*, using a >1000 gene dataset to measure steady-state mRNA and protein expression levels, with $\sim$ 200 sequence features. They found that mRNA-protein correlation was $r_s = 0.46$, significantly lower than correlations found in *S. cerevisiae*. Further to this, they performed individual partial correlations between each SDF and the protein abundance, controlling for mRNA abundance, whereby sequence length (-0.53) and mRNA decay rate (-0.37) correlated most strongly with protein abundance. In conjunction, they use nonlinear Multivariate Adaptive Regression Spline (MARS) [103] modelling to fit the data and for feature selection(2.3.1). Overall, they found that two-thirds of the variance of the model could be explained using their input features to explain protein abundance, a rather significant portion and a breakthrough discovery. Of this, 31% of the variance came from the coding sequence, 27% from mRNA level, 8% from 3'UTR, and only 1% from 5'UTR. Many of the SDFs considered in this study are also applied in this thesis, and this work is considered a benchmark for comparison within this thesis as they developed their own expansive SDF dataset.

Outlier Novelty Detection Developing intricate and powerful regression models are of some benefit, but require coupling with post-regression analyses and identification of weak areas. Outlier detection methods attempt to identify residual 'outliers' ϵ_n and select this subgroup to gain biological insight as to why these proteins are not predicted well. This is one of the main purposes of the work done by Gunawardana et al (2013) [3, 93] and in their thesis.

For the authors thesis, Gunawardana used Lasso for sparse regression and feature selection, incorporating all of the features used by Tuller et al (2007) [137]:

$$\mathcal{E}_L(\mathbf{w}) = \min_{\mathbf{w},b}\{y - \langle \mathbf{w}, \mathbf{x}\rangle + b\}^2 + \lambda||\mathbf{w}||_1 \tag{2.75}$$

they selected weights $\{w_p\}$ outside the range [-0.2, 0.2] as lower and upper thresholds, respectively, selecting the 5 features that actively contribute to protein abundance prediction (see Figure 2.13, Left), which include tAI, ribosome occupancy and codon bias. Overall the lasso model produced an $r^2 = 0.86$ using the 5 best features, up from $r^2 = 0.8$ when using all 37 features. Subsequently they found that training a neural network by SGD did not improve r^2, and that performance substantially dropped when all 37 features were used. Outliers were then determined as belonging in the 2.5% percentile with respect to the squared error ϵ_n^2 of $\mathcal{E}_L(\mathbf{w})$ (see Figure 2.13, Right). Specifically, post-translationally regulated proteins were expected as outliers, given that the model inputs accounted for sequence features that would help to account for translation rate, such as ribosome occupancy and density. 50 proteins were selected by this percentile, of which 48 were 'over-estimated' by the model, which then lead to two subsequent analyses:

- **Coarse-level PTM**: 42 out of 48 contained PTM terms including phosphorylation and glycosylation, which are associated with protein stability.

- **Fine-level PTM**: Combination of coarse-level with PEST-motif sequence identification. This is followed up by GO enrichment analysis, which identified nearly half of the proteins as ribosomal proteins.

Gunawardana also formulates Outlier Rejecting Regression (ORR), which obtains a portion of the data points as robust outliers using truncation and clipping techniques. This makes the regression problem non-convex, requiring a difference of convex approximation algorithm to solve. The clipped loss function is defined mathematically as:

$$\mathcal{E}_U(\mathbf{w}) = \min\{U, \mathcal{E}(\mathbf{w})\} \tag{2.76}$$

where $\mathcal{E}(\mathbf{w})$ is the error function for OLS. In her Thesis Gunawardana uses L2-norm penalty $\mathcal{E}_R(\mathbf{w})$, with intercept b being rolled into $\mathbf{w}$. Here $\mathbf{x}$ and $\mathbf{y}$ are not included for notational brevity. ORR introduces a new parameter μ (not the mean) which reformulates the problem as:

Algorithm 2: Outlier-Rejecting Regression (ORR) [3, 93]

Initialize w_0, hyperparameter $\mu \in [0, 1]$, $k \leftarrow 0$;

repeat

1. Obtain $\boldsymbol{\eta}_k$ from equation 2.79 by sorting $\mathcal{E}(\mathbf{w}_k)$.
2. Compute the gradient using $\boldsymbol{\eta}_k$:

$$\mathbf{g}_{\mathbf{w}} = \frac{1}{\mu N} \sum_{n=1}^{N} (1 - \eta_{nk}) \nabla_{\mathbf{w}} \mathcal{E}(\mathbf{w}_k)$$

3. Update w_{k+1} as:

$$\min_{\mathbf{w}} \frac{1}{(1-\mu)N} \left[\sum_{n=1}^{N} \mathcal{E}(\mathbf{w}) - \mu N \langle \mathbf{g}_{\mathbf{w}}, \mathbf{w} \rangle \right]$$

4. $k \leftarrow k + 1$.

until *convergence*;

$$\min_{\mathbf{w},\boldsymbol{\eta}} \frac{1}{(1-\mu)N} \sum_{n=1}^{N} \eta_n \mathcal{E}(\mathbf{w}), \tag{2.77}$$

$$\text{s.t.} \quad \sum_{n=1}^{N} (1-\eta_n) \leq \mu N, \quad , 0 \leq \eta_n \leq 1, \quad \forall n \tag{2.78}$$

where $\mu \in [0, 1]$ is a hyperparameter that defines the number of outlier samples needed as a ratio of the total data samples. Estimates of $\boldsymbol{\eta}$ can be obtained iteratively using a Difference of Convex Functions algorithm:

$$\boldsymbol{\eta}_k \in \arg\max_{\boldsymbol{\eta}} \sum_{n=1}^{N} (1-\eta_n) \mathcal{E}(\mathbf{w}) \quad \text{s.t. (eq 2.78)} \tag{2.79}$$

where k is the iteration number. The full algorithm is given in (2). Using ORR and comparing to Quantile Regression, Gunawardana found interesting biological insights on the subsequent outlier groups that were isolated, including over half being involved in translation (GO:0006412) and part of cellular-component ribosome (GO:0005840).

2.5 Summary

High-throughput sequencing data has been revolutionary in uncovering breakthroughs with respect to RNA and protein concentration, function, interaction and structure. A large corpus of literature has been covered, including the central dogma of molecular biology, technologies that have led the revolution in genetic research, some underlying statistical and machine learning theory and key concepts surrounding the transcriptome-proteome interface. Many previous authors simply model the correlation between proteome and transcriptome, leaving open space for more sophisticated modelling. Further to this, a lot more research with respect to sequence-derived features have been conducted on *S. cerevisiae* with its comparatively smaller genome with fewer protein-protein interactions (PPI). We have covered Classification, Bayesian, Regression and Outlier-Detection based approaches to transcriptome-proteome modelling, we take inspiration from all these sources to expand in this domain, leading to novel biological insights and novel methods using ensemble and stratified-based modelling.

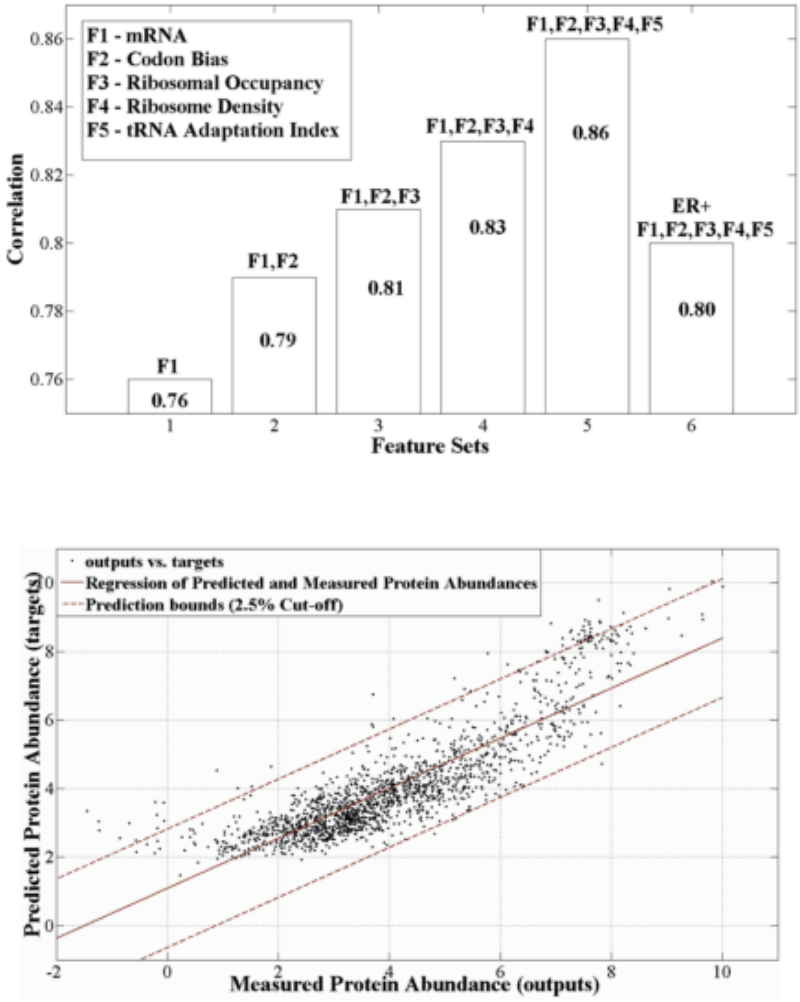


Figure 2.13: Feature selection and outlier detection for Gunawardana et al (2013) [3]. *Top: Greedy forward feature selection for top 5 features in protein abundance prediction. Bottom: outlier selection on predicted vs. actual protein abundance scatterplot using 2.5% quantile cut-off.*

Chapter 3

Developing a protein abundance predictor across the cell cycle

In this chapter, we begin with the work achieved by Gunawardana [3, 93], developing a protein abundance predictor in yeast, in combination with a cell cycle study by Aviner [5], apply data-driven modelling across three expression level types, across three time points in the human cell cycle, with additional SDFs.

3.1 Data Preparation

We begin by describing the processes conducted to obtain mRNA, translation and protein abundance measurements as described by Aviner et al (2015), and a breakdown of their sufficient statistics (see Table 3.1).

Expression Levels HeLa S3 cells were grown in Invitrogen supplemented with 10% fetal calf serum, 2mM L-glutamine and antibiotics. HeLa cells were synchronized using 2mM double-thymidine block for 19h, released from G1/S block in fresh DMEM for 9h, treated again with 2mM thymidine for 18h, released again, and harvested at 2h, 8h (mRNA)/8.5h(translation, protein) and 12h (mRNA)/14h (translation, protein) [5, 138]. mRNA comes from microarray dataset (GSE26922) using Affymetrix Human Gene 1.0 ST Array, from the Gene Expression Omnibus (GEO), with robust-multi array

Expression	**Technology**	**Cycle**	μ_{ML}	Repl. σ^2	Intra σ^2	Repl. ρ	N
mRNA	Microarray (RMA)	G1	10.58	6.0e-05	2.55	0.99	6785
		S	10.56	6.2e-05	2.53		
		G2/M	10.62	1.7e-04	2.53		
Translation	PUNCH-P [4] /MS	G1	19.23	4.9e-03	5.77	0.96	5055
		S	19.08	2.4e-02	5.75	0.95	
		G2/M	19.39	1.9e-01	5.58	0.94	5110
Protein	MS (iBAQ)	G1	24.18	2.5e-02	10.00	0.97	5783
		S	21.72	8.1e-01	9.88	0.97	5763
		G2/M	24.38	3.3e-02	9.59	0.98	5822

Table 3.1: Sufficient statistics from Cell Cycle expression set. *Data taken from Aviner et al (2015) [5].* μ *represents* log *MLE of expression across replicates and samples. Replicate* σ^2 *is the variance between replicates, Intra* σ^2 *is the variance across samples. Replicate* ρ *is the correlation between replicates in that subgroup.* N *is sample size.*

(RMA)-normalized expression values for the given timepoints. Proteomics data (PXD002802) from PRIDE, along with translation (generated via novel method PUNCH-P [4]) are normalized using iBAQ algorithm [139]. Expression levels are normalized by analyzing the same quantity of biological material at each phase to allow for comparisons across the gene product hierarchy. The dataset is combined by first joining translation and protein expressions by Uniprot/Swissprot Accession IDs, then joining to Microarray HuGene 1.0 st c1 probeset IDs using Biomart.

Sequence-Derived Features mRNA transcript variants were extracted from NCBI Entrez Direct [140, 141] via Biopython v1.7 [142] package (Python 3.6). Unique gene names (HGNC) [143] were mapped to curated Refseq accession numbers, obtaining GenBank files for all *H. sapiens* mRNA transcripts. Exon data and elements from feature table were extracted and counted. We filtered for mRNA transcripts whose Refseq ID began with "NM_". Each amino acid sequence was extracted from the "cds" feature in the corresponding mRNA transcript. The mRNA sequence is subsequently split into coding sequence (CDS), 5'UTR and 3'UTR, whereby a number of features are counted such as exons, sequence-tagged sites (STS) and more. We list the features in Appendix S1. GC content and base/amino acid frequencies are calculated directly from the corresponding sequence. We extracted CAI and 'the effective number of codons' (Nc) using CAIcal [144] server using CDS sequence as input in conjunction with the Human Codon Usage table as frequencies per thousand from the Ensembl database (release 57). We used ExPASy's ProtParam [145] module in Biopython to predict pI, Aromaticity, Instability Index, GRAVY and protein secondary structure. tAI values are calculated using stAIcalc by Sabi et al [62], using the offline version with human tRNA gene copy numbers taken from GtRNAdb [63] for hg19 (NCBI build 37.1 Feb 2009). Codon Usage Bias is calculated following the method from Roymondal et al [61], requiring no reference codon usage table. Changes in Gibbs Free folding energy ΔG for 5'UTR is predicted using RNAstructure EnsembleEnergy algorithm [146].

Combined Expression-Sequence Dataset Due to multiple mature mRNA/amino acid transcripts encoding for a single protein, we select the longest mRNA transcript for each protein ignoring inter-transcript variability. SDF count features are scaled by relevant sequence length, for instance mRNA

base counts are normalized by mRNA transcript length. SDF features are then merged into the cell cycle dataset leading to a dataset of 6592 proteins; with $N = 3500$ with no missing values.

Notation Given that our data is drawn from a pseudo-time-series dataset, we will refer to the current cell cycle phase as t, with references to the next 'timestep' as $t+1$ and so on. mRNA expressions are described as $m_n^{(t)}$, where $n = 1, \ldots, N$ mRNA abundances; translation rates $r_n^{(t)}$ and protein abundances $p_n^{(t)}$. Furthermore, G1 corresponds to 2h, S as 8h for mRNA, 8.5h for translation/protein and G2/M as 12h mRNA or 14h translation/protein. Assume, if not explicitly declared, that $m_n = m_n^{(t)}$ refers to the same timepoint across all 'omics.

3.2 Results

The majority of the research in this chapter is published by Parkes & Niranjan (2019) [1], with additional supplementary material and tangents. To begin with, we briefly explore the main cell cycle markers and whether they share a common pattern of mRNA, translation and protein expression (see Figure 3.1). UNG is known to peak prior to the G1/S checkpoint, PCNA and CCNA2 peak in S-phase, CCNB1 peaks in G2/M and AURKA peaks in mitosis. For each of these genes, we plot the proportion of abundances where the peak level is defined at 100%. Interestingly both UNG and PCNA demonstrate lag protein expression whereby high levels of mRNA and translation within G1 phase (but not in S phase) lead to a high protein level in S phase where they operate.

3.2.1 Translation Level Significantly Improves Prediction Over mRNA Level

Since multiple copies of a protein are often produced from a single mRNA strand, we expect translation/protein abundance and variance to be greater than mRNA levels. Indeed, we see translation and protein levels to be several orders of magnitude larger than mRNA (Figure 3.2A), with a larger span indicative of higher variance. Hierarchical clustering of Spearman-rank correlations between triplicative measurements of gene products (Figure 3.2B)

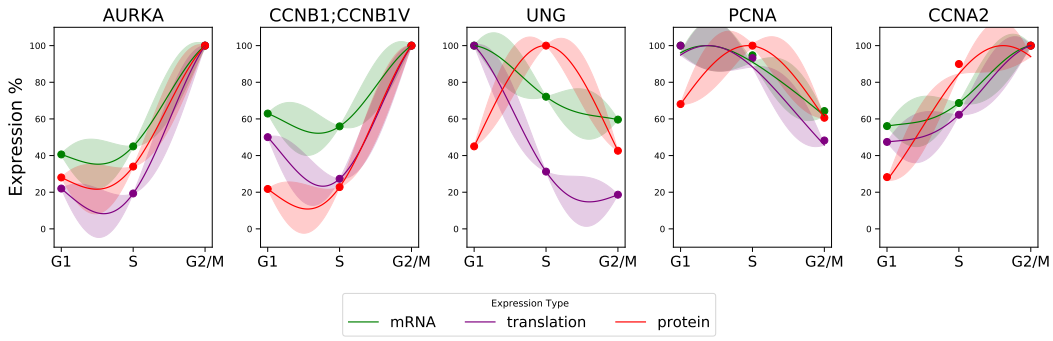


Figure 3.1: Expression patterns of prominent cell cycle markers. *Line plots representing the mean mRNA (green), translation (purple) and protein (red) abundance. Expressions are normalized using 'minmax', scaled by the maximum value to 100% expression. Error regions are modelled using RBF(1) Gaussian Process, with replicate error modelled as* $\alpha = \sigma_R^2$. *Adapted from Aviner et al. [5]*

shows high intra-correlations across the 'omic scale, with translation clustering closer to protein than mRNA. This demonstrates the apparent invariance across the three cell cycle phases in preference to differences between gene products, with mild correlation between transcript and protein levels ($r_s = 0.47$-0.49) across all phases, as demonstrated in the original work and by other authors for mammalian cells [147, 5, 148]. Correlations of translation against protein are significantly higher ($r_s = 0.66$-0.67) at all time points, which is not due to the technical similarity in measurement technique. This is likely due to translation level accounting for robust post-transcriptional mechanisms applied across the transcriptome, such as alternative splicing and mRNA degradation [149]. Visualisation of correlation (Figure 3.2C,D) shows an consistent left skew in mRNA versus protein plots, contributing to a reduction in positive correlation compared to translation. To see whether this artefact is due to the reduction in sample size N alone (5500 to 4000), we separated mRNA measurements by whether they had missing translation level data or not, and calculated r_s for each sub sample (Figure S8A). We do see a drop in correlation ($r_s = 0.23$-0.24) in samples with missing translation data versus samples with data (maintained at stated level), this may be due

to experimental issues with measuring low levels of translation in these genes, and since protein stability can be inferred from translation level (as shown previously [5]), these proteins may not be sufficiently steady-state. Alternatively, due to the low resolution of only having three time points (G1, S and G2/M), these labile proteins may be below the detection threshold at the time of measurement. To further check whether the presence or absence of translation measurements had an impact on corresponding model weights $\mathbf{w}$, we developed a simple linear mixed model (LMM) in the form:

$$\mathbf{p}_j^{(t)} = \mathbf{m}_j^{(t)}\mathbf{w} + Z_j u_j + \boldsymbol{\epsilon}_j \tag{3.1}$$

where $j = 1, \ldots, J$ represents the subgroup of containing-translation or not-containing-translation measurements, Zu represents the random effects over J groups. In this case Z_j is of size n_j and u is of size $J = 2$. This has the effect of adding a random intercept to each group. With u in the range of 0.05 across all t, we conclude that the separation between these gene groups does not have a significant effect on the coefficients, however one of the main LMM assumptions is that values within each subgroup are independent, which is not the case in this example.

Combined Linear Predictor Next, we developed simple linear models that mapped mRNA and/or translation abundance to the corresponding protein abundance at that cell cycle phase (see Table 3.2), with a naive protein abundance predictor with a bias term using just mRNA and translation levels $X_n^{(t)} = \{m_n^{(t)}, r_n^{(t)}\}$ in the form (Figure S8B, Figure S10):

$$p_n^{(t)} = w_0 + w_1 X_{0n}^{(t)} + w_2 X_{1n}^{(t)} + \epsilon_n \tag{3.2}$$

where $p_n^{(t)}$ is the target protein abundance at cell cycle phase t, with w_0 as bias and ϵ_n as residual error. This illustrates that once translation is known, mRNA levels become mostly redundant in protein abundance prediction as there is a negligible increase in r^2 (compared to Figure 3.2C and D). This is supported by a negligible decrease in Akaike's Information Criterion (AIC) between r_n and $m_n + r_n$ across all t. The values of $\{w_1, w_2\}$ also show a shift in weight from mRNA to translation. Parameter uncertainty is additionally quantified using Bayesian Linear Regression as:

$$\mathbf{p}^{(t)} \sim \mathcal{N}(\mathbf{p}^{(t)} | w_0 + w_1 \mathbf{m}^{(t)} + w_2 \mathbf{r}^{(t)}, \sigma^2) \tag{3.3}$$

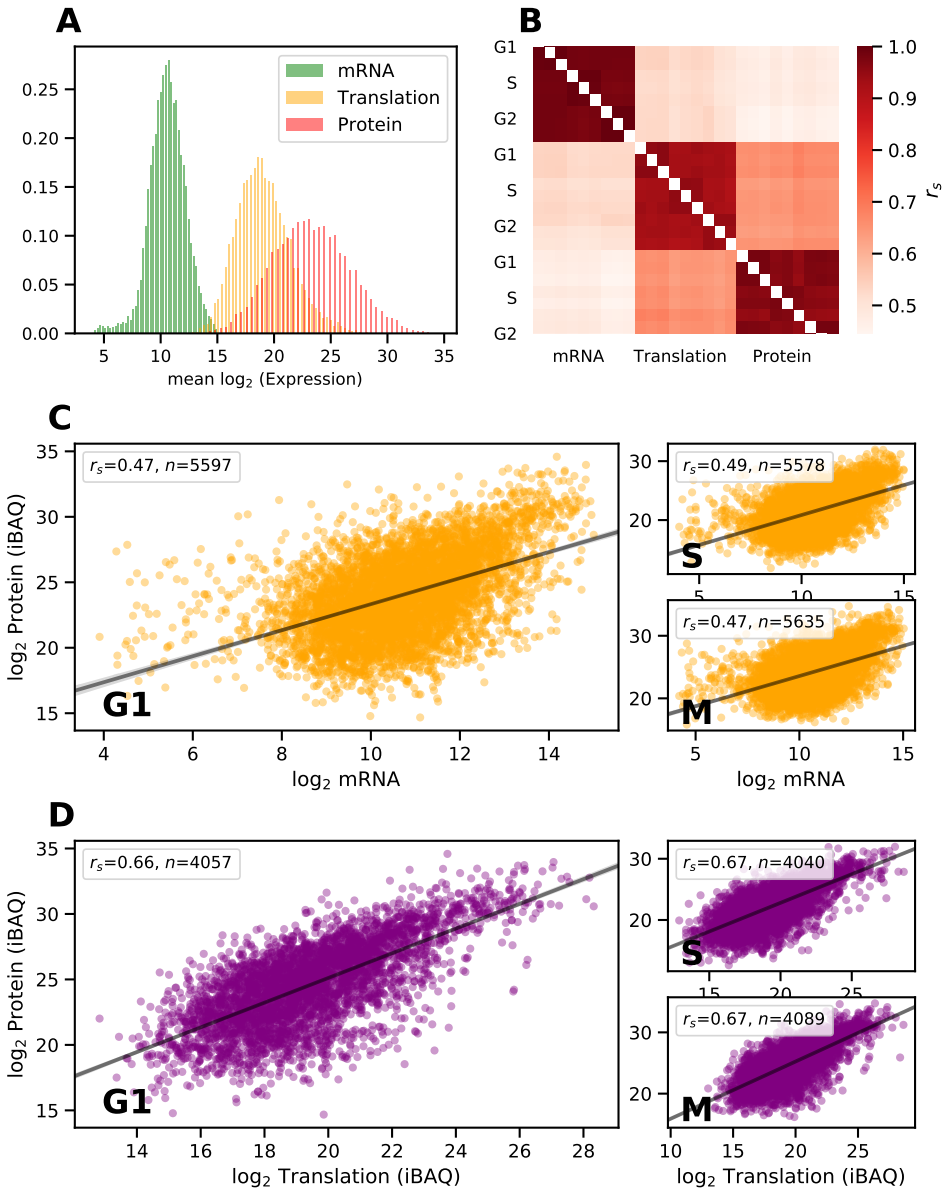


Figure 3.2: Distributions and Correlations in transcript, translation and protein levels. *Published data from [5], shown for completeness. (**A**) Histogram distributions of mean log_2 mRNA (microarray), translation (PUNCH-P) and protein (MS) levels for various cell cycle phases. (**B**) Hierarchical clustering of Spearman-rank (r_s) correlation matrix given expression levels. (**C-D**) Scatterplots of log_2 mRNA versus protein (C) and translation versus protein (D) per cell cycle phase. M equals G2/M phase. n refers to the number of samples.*

Input p ∼X	**Phase**	r^2_{adj}	N	**AIC**	w_0	w_1	w_2
mRNA	G1	0.23	5783	2.82e+04	13.4	0.99 ± 0.04	NA
	S	0.25	5763	2.79e+04	10.7	1.01 ± 0.03	
	G2/M	0.22	5822	2.82e+04	13.9	0.95 ± 0.04	
Translation	G1	0.47	4229	1.91e+04	6.4	0.93 ± 0.03	
	S	0.47	4214	1.9e+04	4.3	0.91 ± 0.02	
	G2/M	0.47	4267	1.91e+04	6.8	0.91 ± 0.02	
mRNA+ Translation	G1	0.49	4229	1.9e+04	5.0	0.31 ± 0.06	0.82 ± 0.03
	S	0.49	4124	1.9e+04	2.5	0.41 ± 0.06	0.78 ± 0.03
	G2/M	0.49	4267	1.9e+04	5.4	0.31 ± 0.05	0.81 ± 0.03

Table 3.2: Linear model parameters and results against protein abundance. *Comparison of 9 linear models of the form* $p_n^{(t)} \sim m_n^{(t)}$, $p_n^{(t)} \sim r_n^{(t)}$ *and* $p_n^{(t)} \sim m_n^{(t)} + r_n^{(t)}$, *respectively. Inputs undergo no standardization.*

using priors:

$$\sigma \sim \text{HalfCauchy}(\beta) \tag{3.4}$$
$$\mathbf{w} \sim \mathcal{N}(0, \sigma_w^2) \tag{3.5}$$

using uninformative hyperpriors $\beta = 10, \sigma_w^2 = 10$. Further details to the Bayesian treatment are considered In Appendix 6.2. Additional model scenarios we explore are:

- **Interaction terms**: By considering the interaction of mRNA with translation level $w_1 m_n^{(t)} r_n^{(t)}$, it may possibly explain more in the relationship with respect to protein (see Figure 3.3A). However r^2 is only increased by 0.01, no significant change in AIC, with very low weighting to the interaction term. Problems also arise with matrix conditioning, with regards to multicollinearity.

- **Polynomial terms**: For mRNA-protein relationships, a second-order term $w_2 m_{nt}^2$ does significantly improve r^2 by 5%, where a curved line of best fit better models lower-expressed mRNA abundances (see Figures 3.3B-C). However this is not to say that log mRNA-protein relationships are quadratically related, previous studies have also modelled this as a piecewise-linear [6] which makes more intuitive sense; separating lower and higher-expressed mRNAs using a hinge function. Additional polynomial terms $K > 2$ yielded no significant change in r^2 or AIC.

- **Lagging terms**: Given the time-series nature of the data, we can compose models in the form:

$$p_n^{(t)} = w_0 + w_1 m_n^{(t)} + w_2 m_n^{(t-1)} + \epsilon_n \tag{3.6}$$

  relying on a lag-effect between mRNA and protein production. In general we find a 1% increase in r^2, which isn't surprising as we would not expect this lag-effect to be present nor significant in a majority of proteins.

- **Piecewise-linear**: Following from Vogel [6], we deployed Friedman's MARS model [102] to model the mRNA-protein relationship (see Figure 3.3D). MARS identifies two non-pruned hinge-points at log2 mRNA

abundance values of 9.15 and 11.72, which separates the abundance domain roughly into 'low-expressed mRNA', 'medium-expressed mRNA' and 'highly-expressed mRNA' with corresponding slopes. MARS also gives one of the best r^2 at 0.28; albeit relatively unimpressive compared to *S cerevisiae* model correlations.

Aviner's [5] subsequent analysis focused on the fold-change differences across mRNA, translation and protein:

$$\Delta p_n^{(t)} = \mathcal{Z}(p_n^{(t+1)} - p_n^{(t)}) \qquad \forall n \tag{3.7}$$

where $\mathcal{Z}$ is z-score transformation, or standardization and t represents the current time-step or cell cycle phase, being G1, S or G2/M. This process occurs for mRNA m_n and translation r_n in addition to protein levels p_n (see Appendix S9). As expected, most genes do not change significantly with respect to abundance across the cell cycle, with the vast majority of fold changes being no more than 2-fold. There is a notable increase in correlation between changes in mRNA and translation; that is to say that when mRNA levels change, translation levels are more likely to correspondingly change in similar fashion ($r_s \sim 0.25$), compared to no significant correlation between mRNA-protein or translation-protein ($r_s \sim 0.03$). Whilst the mean fold change is normalized, there is a 2-fold global increase from S phase to G2/M phase, and a corresponding 2-fold global decrease from G2/M to G1; indicating that the vast majority of abundances double in quantity in the run-up to cell division.

At this point, the limit of what could be understood from a statistical point of view had been reached, hence to continue on this path, we needed to extract additional information about each gene and protein, as inspired by Gunawardana [3, 93], Tuller [137, 53] and others [6]. This information we describe as sequence-derived features (SDFs), and we will expand on the detail, assumptions and limitations of SDFs in the next chapter. However for now we will continue to develop a more sophisticated model by incorporating an additional 30 or so features into our subsequent analysis.

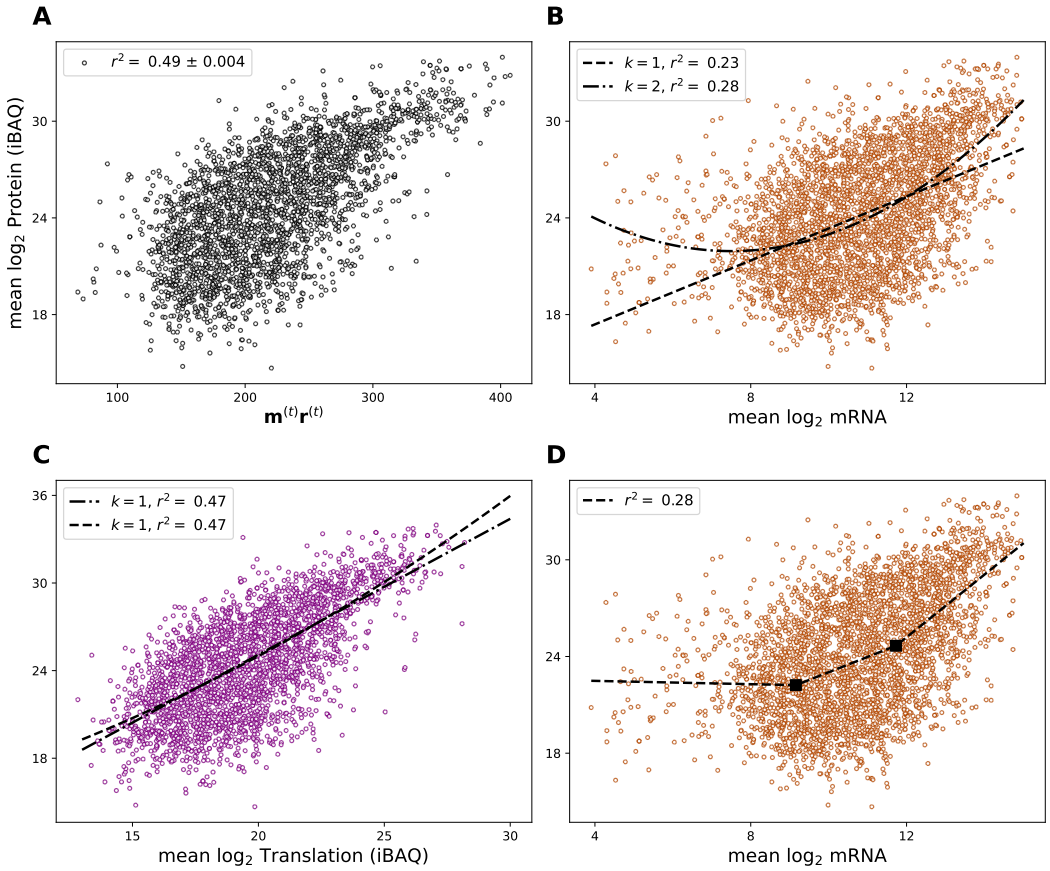


Figure 3.3: Modelling of G1 mRNA-protein relationships under different scenarios. *Scatterplots of A) Interaction between mRNA-translation $m^{(t)}r^{(t)}$ against protein $p^{(t)}$, B-C) Polynomial models $k = 1, 2$ including quadratic terms for mRNA-protein m_n^2 (orange) and translation-protein r_n^2 (purple), D) MARS model for mRNA-protein, with 3 non-pruned coefficients inc. intercept. Hinge points indicated as squares.*

3.2.2 Sequence-based Features Cumulatively Improve Prediction, But Individually Correlate Weakly

We mined for features primarily from curated RefSeq mRNA transcripts and associated amino-acid sequences (beginning with NM_ or NP_) from the NCBI Entrez database [140] using HGNC gene names [143]. A number of SDFs were extracted from the underlying mRNA or coding sequence (CDS), in addition to frequency-based features that are identified in the Genbank feature table, and are described here (see Appendix S1). Next, we explore pairwise correlations between all the features, as well as their correlations to the target protein concentrations as a clustered intensity plot (Fig 3.4), with translation, mRNA levels, sequence-length/protein molecular weight (PMw) and CUB with the largest absolute Spearman-rank correlations to protein level (r_s = 0.66, 0.47, −0.4, 0.37 respectively). Interestingly the negative correlation between Length/PMw to protein level would suggest that larger proteins are more likely to have lower abundance across all phases. Indeed we would expect enzymatic proteins, known to be smaller; to be higher in abundance than larger proteins which predominantly involve structural interactions.

Further to this, the comparatively small correlation of tAI and CAI with respect to protein with regards to previous authors [3, 53, 6] may be due to differences in gene regulation complexity between humans/yeast. However, the correlation matrix does not inform on how features will cumulatively interact with each other in any subsequent models, therefore making it difficult to identify redundant features. To examine this effect, we performed Principle Component Analysis (PCA) on the input matrix (i.e all the features minus protein) to see how much explained variance can be in the largest eigenvalues (see Figure S11). Whilst there is noticeable dominance within the first six principle components, there is not a clear exponential decay in feature importance, indicating that there are small, cumulative factors at play in these features that may contribute independently useful information. In addition, the assumption of linearity required for PCA transformation use may not hold true in the biological system due to complex interactions between mRNA and protein *in vivo*. Further to this, we examined the scatterplots from t-distributed stochastic neighbor embedding (t-SNE) and observed uniform scattering/little structure in reduced dimensions. Due to these reasons, we used feature selection instead of PCA in downstream analyses.

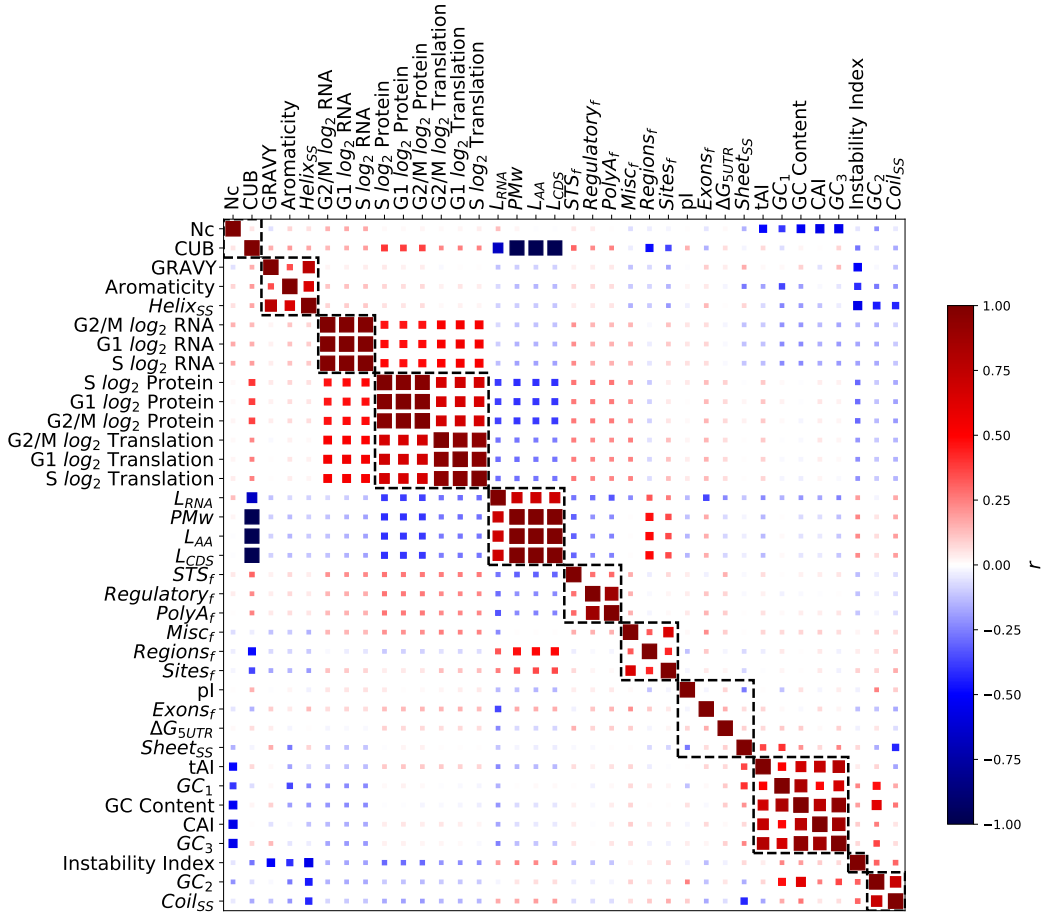
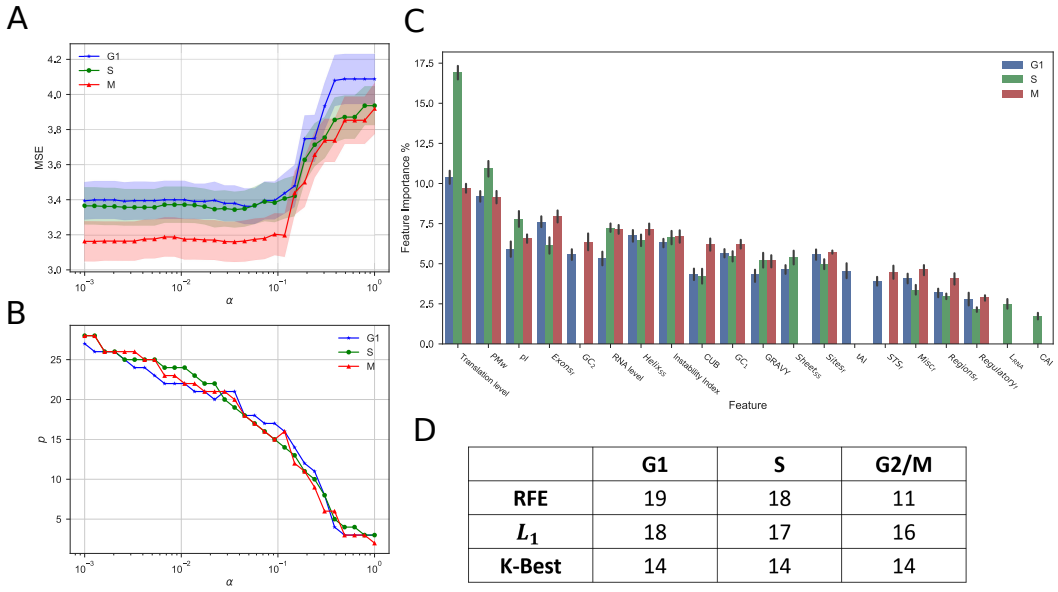


Figure 3.4: Insightful inter-correlations across sequence-derived and gene expression data. *Pairwise Spearman-rank (r_s) correlation matrix in Hinton format for all continuous input features including target (protein). Colour and square size refer to correlation. Features are sorted by agglomerative clustering,('single' linkage), groups are boxed in black dashed lines. See Table S1 for abbreviated labels.*

Analysis of Feature Selection Approaches To examine the potential of different computational methods on this dataset, we performed 10-fold cross validation on different regressors across all phases (see Figure S12), with gradient-boosted regression trees (GBRT) consistently providing marginally higher accuracy on out-of-sample data (r^2=0.64 ± 0.06) than other methods, and performing significantly better than using just mRNA and translation as inputs (r^2=0.49 ± 0.02). We note that GBRT is non-linear in its approach, and fairly robust to overfitting due to averaging over base tree estimators. It is interesting to observe the surprisingly good performance of simpler algorithms like OLS still achieving reasonable out-of-sample accuracies (r^2 =0.61 ± 0.06), confirming the robustness of the dataset and highlighting its case for continued use in future studies in protein prediction. Indeed, both Gunawardana [3] and Tuller [137] found non-linear models (such as neural networks) brought little benefit and even reduced correlations. In addition, both Gunawardana and Tuller got larger correlations from linear models ($r^2 = 0.86, 0.76$ respectively) but both developed models for steady-state yeast, not dynamic human cells. We do however observe marginal non-linearity in scatterplots (Fig 3.2C,D) at extrema thus supporting the use of a pseudo-linear method. However in the interests of reducing overestimation from correlations within related features, we deployed three different methods of feature selection as no method is known to be optimum:

1. Recursive Feature Elimination (RFE)
2. ℓ_1 sparsity-inducing regularization (LASSO)
3. Selecting k-Best (ANOVA)

For step-by-step details of the feature selection parameters used, see Supplementary Section 6.2 for more details. For inducing an appropriate amount of sparsity into the input matrix using ℓ_1 regularization, selecting the regularizing term α is crucial. We observe a dramatic increase in mean-squared error (MSE) rate with $\alpha > 0.1$ (Fig 3.5A) across all cell cycle phases, while the number of features remaining p falls linearly as α increases (Fig 3.5B), showing strong redundancy with at least half (14) of all features. Using the optimized α, we created a GBRT model (with 10-fold cross validation (CV)) using the regularized feature matrix generated from CV Lasso models, and describe the model coefficients as feature importances (Fig 3.5C). Unsurprisingly, translation level dominates as the most important feature across all



	G1	S	G2/M
RFE	19	18	11
L_1	18	17	16
K-Best	14	14	14

Figure 3.5: Impact of ℓ_1-regularization on reduced feature sets. *(**A-B**) Line-plots representing parameter tuning of regularizing term α against the mean-squared error (A, MSE) and the number of features remaining (B, p), across all cell cycle phases, where $\alpha \in [10^{-3}, 1]$. Error bars indicate $\pm$SD with 10-fold CV. (**C**) Bar plot representation of model coefficients (as importance) for each feature from the Gradient-boosted regression tree (GBRT), using optimized α, for each cell cycle phase. Error bars indicate $\pm$SD with 10-fold CV. (**D**) Table of number of selected features per method, per cell cycle phase.*

phases, but the remaining features mostly appear to have similar importance (5-8%), with amino-acid derived features such as PMw and pI, on average, performing better than traditionally used mRNA-based metrics like tAI or CAI. All 3 of the feature selectors reduced the most number of features from G2/M phase compared to G1 (Fig 3.5D), which may suggest G1 and S proteins may be affected by post-translational regulations.

Here we see divergence from work done on other model organisms (such as yeast and *E. coli*), which have shown strong correlation contributions from codon bias metrics like tAI and CAI [3, 137]. We suspect this is due to the increased presence of post-translational modifications (PTMs) within higher-order organisms like *H. sapiens*, causing fluctuations on protein abundance that act as noise to the correlation with these mRNA-based metrics. It is also a possible factor that tAI/CAI information value is simply absorbed into translation/PUNCH-P measurements rendering their contributions somewhat smaller when combined with translation. We note the increased skew of feature importances within S phase (significantly larger translation, PMw, pI), possibly indicating that these features are more active in predicting DNA replication/repair mechanisms associated with this phase. In the original work, Aviner et al. [5] also explored S phase regulation in more detail in their further analysis in relation to fold changes, therefore complexities in S phase may indicate more frequent post-translational modifications. However exploring the importance of each feature only begins to provide biological interpretation into the complex interplay between features - our primary interest is novelty detection in outliers with respect to a predictive model.

3.2.3 Overestimation In Majority Of Protein Outliers Indicates Post Translational Modification Or Degradation

Next, we incorporated reduced input from ℓ_1 regularizer sets into GBRT models for G1, S and G2/M cell cycle phases (see Figure 3.6, Table S1), using Leave-One-Out Cross Validation (LOOCV) for each predicted gene (Figure 3.7A), with significantly stronger Pearson product correlations (r_p=0.82, r^2=0.67) across all cell cycle phases than a naive predictor with just mRNA and translation inputs, therefore explaining two-thirds of protein variation.

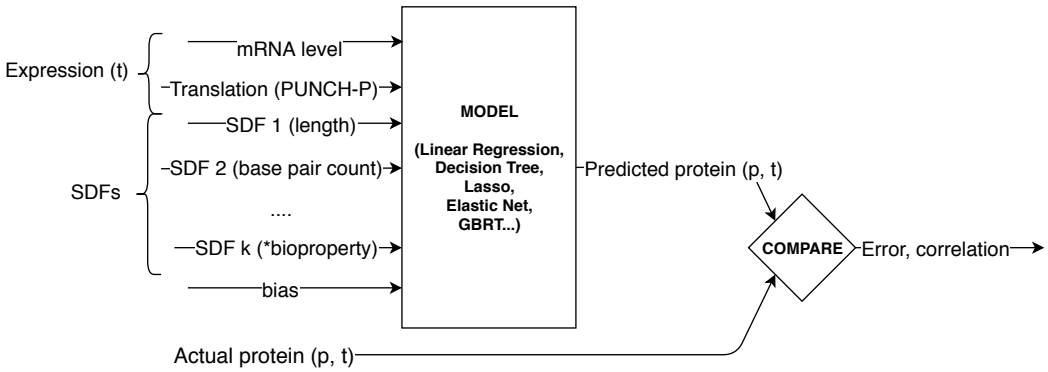


Figure 3.6: Model for mRNA, translation and SDF against protein abundance. *Flowchart diagram describing the model construction for a mRNA-translation-SDF predictor. Read supplementary 6.2 for model and feature selection.*

Vogel et al. [6] found similar findings, with features that focused on individual amino-acid frequencies, additional experimental data (such as mRNA decay rate) and codon-related features. They too found polyadenylation, GC content and codon bias index to be insignificant features, with strong negative correlations in coding sequence and 3'-UTR sequence length (refer back to Fig 7). Previous work has demonstrated that short mRNAs tend to be more stable than long mRNAs [150] and are more efficiently translated; with the addition that resulting short amino-acid chains may fold into their tertiary structure faster than their longer counterparts. Other arguments stem from decreased translation initiation in long sequences [15], due to an increase in mRNA secondary structures found in longer 5'-UTR regions.

With perfect prediction as $y = x$, outliers signify difficult-to-predict proteins that according to our hypothesis are involved in post-translational modifications/processes, which we characterise using different percentiles with respect to the squared-error (ϵ^2, red). Indeed across all phases and feature selectors, we notice at least a 2:1 ratio of outliers lying above the regression line to below, indicating that the global model trained on all proteins tends to overestimate the abundance of some proteins when in fact they should be lower. This ratio is lower than Gunawardana [3] where the ratio was 23:1 above/below conducted using steady-state yeast models, therefore for

this pattern to follow in a dynamic experiment is supportive of using novelty detection as a powerful theoretical principle. This would strongly suggest that post-translational modifications or degradation is taking place in these proteins which are not accounted for in our model input parameters. For proteins underestimated in abundance, this may be due to lack of resolution in only having three timesteps (six hours apart), detecting proteins without steady-state abundance, or time-lag concentration effects. Outlier overlap between feature selectors is reasonable see (Figure 3.7B), with roughly two-thirds of proteins identified as 90th percentile outliers across RFE, ℓ_1 and K-Best feature selectors, to improve robust identification of outlier proteins. In addition to this, there is surprising overlap between cell cycle phases (Figure 3.7C), with roughly one-quarter of proteins found to act as outliers across all 3 phases, with roughly double S-G2/M outliers compared to G1-S or G2/M-G1 outliers, across multiple percentiles. To see the full set of intersections between methods and phases, see Figure S13.

Across 90th percentile outlier proteins, ZNF687 and CTNNB1 (both above prediction line) occur in the top 5 outliers with highest ϵ across all 3 phases, with many proteins not fluctuating much in terms of ϵ across the cell cycle.

3.2.4 Evidence Of Post-Translational Modification/Degradation In Outliers Reveals New Insights

To contrast our hypothesis of post-translational modification (PTM) in outlier proteins, we generated structural site predictions of Acetylation, Methylation, Palmitoylation, Phosphorylation and Sumoylation for each amino-acid sequence. We then calculated the total number of PTMs for each protein and compared the outlier mean total PTM to 10000 mean total PTMs from randomly sub-sampled protein sets of the same size (see Figure S14). In all upper 90th percentile sets we examined, we found the vast majority of outlier sets to have a mean PTM score greater than the distribution μ, with S phase consistently lying furthest from the mean; thus indicating that outliers found in our regressors are more likely to have significantly more post-translational modification sites.

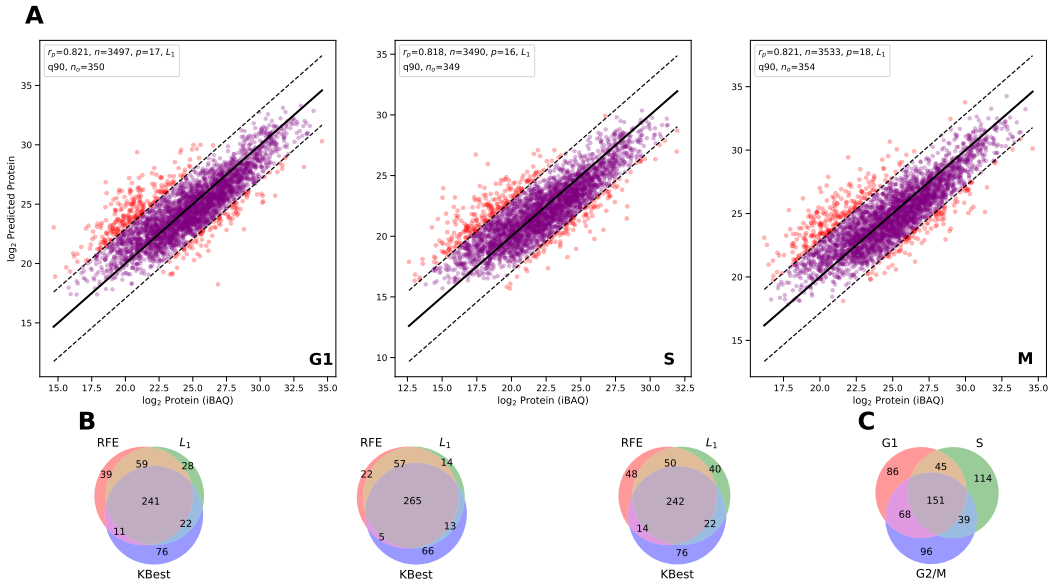


Figure 3.7: Biological interpretability in outliers between actual and predicted protein abundances. *A) Scatterplots of measured protein abundance (y) against ℓ_1-regularized predicted ($\hat{y}$), using GBRT with LOOCV for G1, S and G2/M cell cycle phases. 90th percentile outliers with respect to ϵ^2 highlighted in red. n_o refers to the number of outliers. B-C) Venn diagrams of outlier (red) overlap between RFE, ℓ_1 and K-Best feature selectors per cell cycle phase (B), and between G1, S and G2/M cell cycle phases per feature selector (C).*

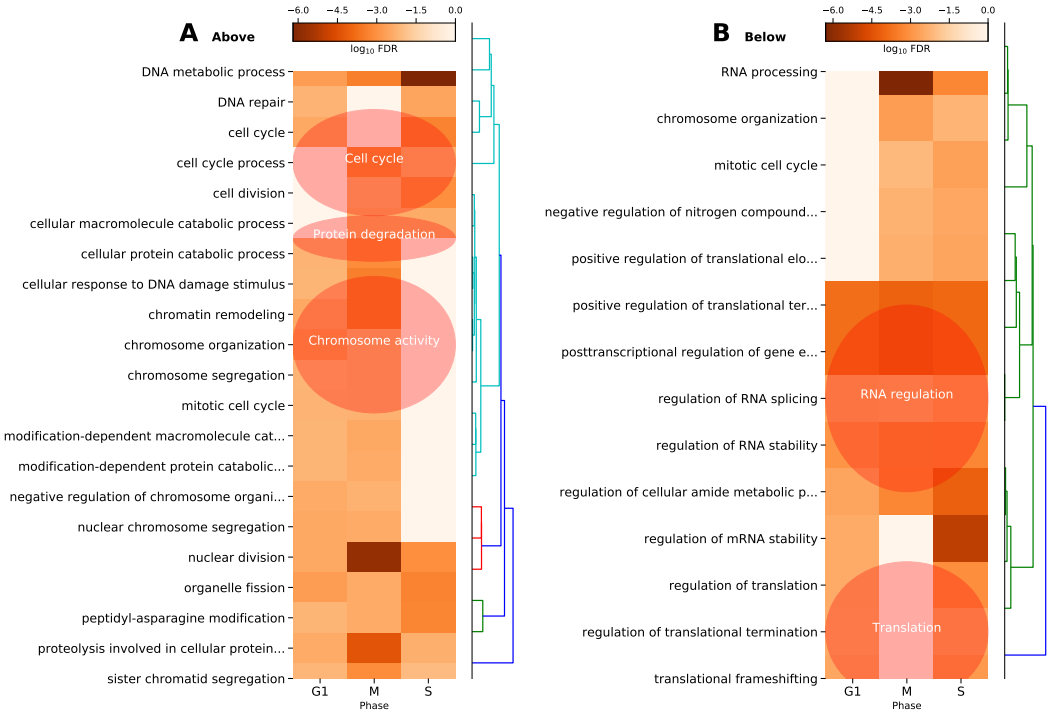


Figure 3.8: Gene Ontology (GO) analysis reveals separation of functionality between outlier groups. *Hierarchical clustering of selected Gene Ontology Biological Process (GOBP) terms in at least 2 cell-cycle phases using (log_{10}) p-value FDR with B&H correction ($p < 0.01$), in 90th percentile in the above-boundary outlier (**A**) and below-boundary outlier (**B**) groups.*

To consider deeper functional roles than just exploring the counts of PTM sites (considered our 'coarse level analysis'), we perform Gene Ontology Biological Process (GOBP) enrichment analysis on 90th percentile outliers for each cell cycle phase and clustered them in terms of their term significance/occurrence (Figure 3.8). We filtered for GOBP terms that had an FDR value < 0.01 across at least 2 cell stages. G2/M phase contained the largest number of significant terms identified, with strong evidence for post-translation degradation pathways found in protein catabolic process/ubiquitin-dependent catabolic process terms (bottom of cluster), across all 3 cell cycle stages. Alongside this, we also found strong significance in (negative regulation of) chromosome organization across all phases, suggesting a strong relationship between chromatin modelling and post-translational modifications/degradation with associated proteins. Indeed, we found strong presense of helicases (HEL-), ATAD2 and E2F4/5 in all outlier sets, known to have roles in DNA repair/chromatin-modifying proteins [151]. Further to this, the presense of many (regulation of) cell-cycle related terms between G2/M-to-G1 stages indicates that post-translational modification/degradation contributes significantly in robust control of cell cycle factors; perhaps more than previously expected. The gene regulation network within the yeast cell cycle have already been explored in detail [152], and highlights the fact that although over 800 yeast genes are involved in the overall process, a significantly smaller portion are responsible for regulating the core cell cycle itself.

We performed split enrichment analysis on outliers found above and below the regression line, wherein with above outliers; protein catabolic/proteolysis terms to exist only in M-G1 stages, with cell cycle/division/chromosome segregation across all 3 stages, with DNA repair/response to DNA damage found shared between G1-S. Contrasted to below outliers; we found dominance of post-transcriptional regulation terms and translational frameshifting across all 3 stages, with RNA/mRNA stability found in S-G2/M groups, and RNA processing/regulation of RNA splicing found in G1-S. Thus the key takeaway messages are: overestimated proteins tend to either be involved in the cell cycle, protein degradation or chromosome modification. Underestimated proteins tend to play a role in RNA regulation or the translation mechanism. We also looked at different percentile cut-off points; namely 95% and 99% percentiles in the 'above' group, and find particular emphasis on proteolysis and protein catabolic terms and their derivatives (see Figure S15) within G2/M, chromosome organization within S phase and cell cycle

processes within G1 and G2/M. However the 99% group only contained $N = 35$ genes, and so any statistical conclusions to be drawn should be taken with caution.

3.3 Discussion

Analysis of Time-Series Concentration With Sequence-Derived Features

In this chapter we have collated time-series concentrations of mRNA, translation and protein from Aviner [5] and sequence-derived features from other sources [140, 153]. Consistent with previous authors, our data shows that mRNA and translation go some way in explaining protein variation (r^2=0.23 and 0.45).This diverges from previous similar work by Schwanhäusser et al [139], where protein translation is calculated using a mathematical model of mRNA and protein rates, rather than measured directly; and where sequence derived features are not factored in their analysis. Our data establishes the redundancy of using mRNA level as a proxy to protein level with the introduction of translation measurements via PUNCH-P [4], likely due to factoring in post-transcriptional controls as translation occurs after mRNA processing. The remaining discordance in correlation between translation and protein is therefore mostly associated with post-translational regulation of protein abundance once synthesised.

To improve predictive power, we extracted features about physical properties associated with the underlying mRNA/amino acid sequence such as CAI, tAI and gene length. Clustered inter-correlation analysis between features showed groupings of features usually by function (i.e strong correlation between mRNA and amino-acid length). Negative correlations between sequence length and protein level have been similarly reported in studies of other organisms [6], and is theoretically supported. However codon bias correlations (CAI, tAI) to protein are noticeably smaller than in previous studies [3, 61], which may be due to further robustness of the gene regulatory framework in *H. sapiens* compared to *S. cerevisiae*, or due to recording dynamic time-series nature of the data rather than a steady snapshot. Feature selection techniques were explored (see Supplementary 6.2) to find the most appropriate extracted features, of which several were identified as important

across different techniques; including instability index, protein weight and CUB.

To simplify the model (and prevent overfitting), we considered unsupervised learning techniques, particularly PCA and t-SNE which underperformed, due to the complex interactions occurring between the features. Whilst other applications for dimensionality reduction often have significantly higher dimensions p, such as image or natural language processing; we found many features contributing a small but significantly cumulative reduction in model error. This highlights the diverse low-impact optimizations that exist in the cellular framework for self-modulation, whether by sequence length, codon bias, translational efficiency or other pre-translational methods in each associated mRNA.

Predicted Outliers Indicate Post-Translational Regulation

Supervised learning on the input features enabled a linear comparison between actual and predicted protein concentrations, where we inferred that proteins furthest from the linear model are involved in biological processes which are primarily regulated post-translation. Choosing the most appropriate percentile to identify outliers is not clear; Gunawardana et al. [3] chose a 2.5% cutoff, but had a small number of outliers ($<= 50$). We chose a 10% (90th) cutoff in order to improve the significance of subsequent GO analyses, at the cost of possibly including proteins that may not be deemed as outliers. Modest overlap (25-40%) between outlier proteins across the cell cycle shows a core group of proteins that the model fails to predict consistently, which is enriched for catabolic processes. In relation to effects from time-delayed mRNA expression, we found that it partially affects 10-12% of proteins we've sampled by bootstrapping, but due to low time-resolution with only three steps in the cell cycle, this conclusion is drawn with caution as a 6-hour time delay window is more than sufficient for mRNA expression levels to change aberrantly.

Expanded Future Role for Sequence-Derived Features

One surprising takeaway from this initial research phase was the power of SDFs in model prediction, despite the fact that we would assume there would

be little useful information within the DNA sequence to actively predict something as complex as dynamic protein abundance. Following this line of thinking, we returned to the database sources and extracted new sequence-based and frequency-based features known before protein synthesis to use as inputs for a machine learning predictor model. We believed that this might have the effect of exaggerating interesting outliers even further. Our downstream analysis develops this to expand the original dataset considerably to discover new insights across the cell cycle, and indeed in other *H. sapiens* cell lines.

Chapter 4

Multi-context sequence-derived features for general application

When working with data from the cell cycle, it quickly became apparent that expression data alone would be insufficient to properly expand our understanding of the cell cycle. Thus, based on previous studies [137, 3], we initially expanded our dataset with around 30 additional features based on sequence-derived features (SDFs) engineered from the RNA/amino acid sequence. This work was published in BMC Bioinformatics [1]. In this chapter, we're going to conduct a full-fledged analysis of SDFs, how they depend on each other with linear and non-linear dependency metrics, with techniques for dimensionality reduction whilst retaining the maximum interpretability. Parts of the research in this chapter is under review with Nucleic Acids Research (NAR), with additional supplementary material and tangents.

4.1 Data Preparation

Here we describe the processes of generating the enlarged SDF dataset of over 200 features. Some of this will be reminiscent of Chapter 1, but there are noticeable differences.

SDF Extraction mRNA transcript variants were extracted from NCBI Entrez Direct [140, 141] via Biopython v1.7 [142] package (Python 3.6). Unique gene names (HGNC) [143] were mapped to curated Refseq accession numbers, obtaining GenBank files for all *H. sapiens* mRNA transcripts.

Exon data and elements from feature table were extracted and counted. We filtered for mRNA transcripts whose Refseq ID began with "NM_". *H. sapiens* amino acid (AA) sequences were taken directly from Uniprot/Swissprot, selecting Proteome UP000005640. The mRNA sequence is subsequently split into coding sequence (CDS), 5'UTR and 3'UTR, whereby a number of features are counted such as exons, sequence-tagged sites (STS) and more. Numbers of exons, sequence-tagged sites (STS), misc features, regulatory regions and poly-adenylated tails in the mRNA transcript are counted. Protein sites, regions, molecular weight (PMw) and more in the AA sequence are counted. We also count mono and di- nucleotide frequency for mRNA, CDS, 5'UTR and 3'UTR transcripts. Amino acid frequencies are calculated for the corresponding amino acid transcripts. We extracted CAI and 'the effective number of codons' (Nc) using CAIcal [144] server using CDS sequence as input in conjunction with the Human Codon Usage table as frequencies per thousand from the Ensembl database (release 57). We used ExPASy's ProtParam [145] module in Biopython to predict pI, Aromaticity, Instability Index, GRAVY and protein secondary structure. tAI values are calculated using stAIcalc by Sabi et al [62], using the offline version with human tRNA gene copy numbers taken from GtRNAdb [63] for hg19 (NCBI build 37.1 Feb 2009). Codon Usage Bias is calculated following the method from Roymondal et al [61], requiring no reference codon usage table. Changes in Gibbs Free folding energy ΔG for 5'UTR, a proxy mRNA secondary structure, is predicted using RNAstructure EnsembleEnergy algorithm [146], using window sizes of $\{L$,10,20,30,40,60,100$\}$. PEST regions for amino acid transcripts are calculated using the Emboss suite of the European Bioinformatics Institute (EBI) using a window size of 10. Post-translational modification (PTM) features were taken from experimental studies as collated by PhosphoSitePlus [154] and counted. Gene Ontology terms were taken from the Gene Ontology Consortium [155], where labels to other related gene information were obtained via the Biomart portal [156]. See Appendix 6.2 for full text details of each text-mined SDF from Refseq and Uniprot/Swissprot.

Notation In this chapter, we will refer to a generic sequence-derived feature vector as $\mathbf{x}_p$, where $p = (1, \ldots, P)$. Each $\mathbf{x}_p$ belongs to a gene region $R_p \in \mathcal{R}$, which can be the mRNA, CDS, amino acid sequence and so on. Furthermore, each $\mathbf{x}_p$ also has a data group $D_p \in \mathcal{G}$, which represents whether the feature is a mononucleotide frequency, codon bias, text-mined feature

and so on. To illustrate different normalization techniques to preprocess $\mathbf{X}$, we will use the notation $\mathbf{X}^{(0)}$ to illustrate unscaled, $\mathbf{X}^{(1)}$ for normalization 1, and so on. Where correlations contain an asterisk r_*, this indicates that different ways to calculate coefficients may be grouped together, such as Pearson, Spearman and/or Biserial coefficient depending on the pairwise combination and the data type therein. η is used as an arbitrary threshold or tolerance parameter. K within section 3 of this chapter refers to the reduced dimensionality of $\mathbf{X}_P$.

SDF Preprocessing As a precursor step, for each feature $\mathbf{x}_p$ we eliminated certain features by the following liberal criteria in order:

1. Greater than 50% missing values.
2. Very low variance and noninformative, i.e $\sigma^2(\mathbf{x}_p) < 10^{-7}$

Secondly, given that many of the features are dependent on the length of the gene, mRNA transcript or amino acid sequence, depending on the source, our first approach was to normalize by length, or by other intuitive factors given various inputs. For details as to which normalization applies to which feature group, see Table 4.1 for details. For normalization 1, let $\mathbf{x}_p$ be the vector of counts for feature p, then the count frequency is given simply as:

$$\hat{\mathbf{x}}_p^{(1)} = \frac{\mathbf{x}_p}{\mathbf{L}_R}, \qquad \forall p \tag{4.1}$$

where $\mathbf{L}_R$ is the vector gene lengths, with $R \in \mathcal{R}$ referring to the data region, whether that be mRNA, CDS or AA etc. An alternative approach would be to view frequencies of bases with respect to some expected frequency, for instance since there are 4 DNA bases, we could assume uniform distribution between all four DNA bases, and hence the expected value for all bases j is the expected reciprocal $\mathbb{E}_j[1/T]$, where T is the total number of unique bases (4) or di-bases (16), depending on the sequence type. Note that for dinucleotide expected values, it is $\mathbb{E}^{\mathrm{DN}}[1/2T]$. Then the relative frequency for feature p becomes:

$$\hat{\mathbf{x}}_p^{(2)} = \frac{\mathbf{x}_p}{L_R\,\mathbb{E}[1/T_R]} - 1, \qquad \forall p \tag{4.2}$$

Note that in the amino acid case, this is a little more complicated as each amino acid is encoded by trinucleotide patterns within the mRNA. To

Feature (Group)	**Normalization 1**	**Normalization 2**
Mononucleotide count (mRNA)	See equation 4.1	See equation 4.2
Dinucleotide count (mRNA)	See equation 4.1	See equation 4.2
Amino acid count	See equation 4.1	See equation 4.2
Isoelectric point	$x - 7$	
Instability Index	$x - 40$	
Kozak sequence	L_{CDS}	
Text-mined features (mRNA)	L_{mRNA}	
PTM	L_{AA}	
Gene length	$\log(x + 1)$	

Table 4.1: Summary of normalization strategies for each SDF feature group $\mathcal{G}$. *Base counts undergo differing strategies for comparison, L refers to gene length of respective source. Where a length is specified, it is scaled by as* $\mathbf{x}_p/L_p$.

account for this, $\mathbb{E}^{\text{AA}} \neq 1/20$, but rather the number of synonymous codons for amino acid j, normalized by the total number of trinucleotides, which is 64. What this means is that values $\mathbb{E}[\hat{x}_{np}^{(2)}] = 0$ meet the expected frequency, where divergent values indicating a departure from this frequency. In the next section, we always use the 'Normalization 2' methods unless otherwise stated since it nearly always centers the distribution around zero whilst maintaining a keen interpretability.

Dataset Summary Over 200 sequence-derived features have been extracted, of which 112 are derived from the NCBI Refseq repository [140] of 50007 human mRNA curated transcripts, with the remaining 78 from 20395 Uniprot/Swissprot human AA curated transcripts, with associated PTM data from PhosphoSitePlus [154], deriving estimate PTMs from AA sequences.

Figure 4.1: The Breakdown of Sequence-Derived Features and their sources. *A pie chart detailing the number and proportion of features by database source (inner ring) and broad Data-Grouping $\mathcal{G}$ (outer ring). mRNA/Ensembl/Refseq-like features (54%) are denoted in shades of green, amino-acid/Uniprot/Swissprot-like features (33%) are denoted in shades of red. Post-Translational Modification (4%) features are extracted separately (in purple). HGNC labels (7%) are not included in the machine-learning algorithms but are shown here for completeness (in blue).*

4.2 Results

The majority of engineered features are a form of count feature measuring a biological phenomenon, whereby a majority are derived from the frequency of mono-, di- or tri-nucleotide occurrence. A breakdown of sequence-derived features by broad affiliation and database source (Figure 4.1) shows a majority of features derived from mRNA (55.3%), a third from amino acid (33.4%) and the remaining PTMs constituting 4% of the features. HGNC features [143] consist of identifiers only and is therefore not used in downstream analyses. Note here that we do not include expression or halflife data that we will use later on for further abundance regression analysis. The imbalance between mRNA/amino acid features is partly due to the richer GenBank format which stores larger amounts of meta information regarding the mRNA transcripts.

The high prevalence of nucleotide frequency features in mRNA (71.9%) and minority amino acid frequency in proteins (29.9%) lead to a heavy length-based dependency, which we correct for by scaling by the appropriate length, or as a ratio to the expected frequency. The frequency of bases, and indeed the off-frequency of particular codons, otherwise known as *codon bias*, is known to correlate with post-transcriptional regulation, mRNA decay and influences translation [6, 53, 157, 158]. Continuous feature groups include estimations of the Minimum Free Energy (MFE) of the 5'UTR region (using various window sizes) and amino acid biophysical properties, such as Isoelectric point [159] and GRAVY [160].

4.2.1 Sub-Analysis of Sequence-Derived Features and Derivation

In this section we're going to begin by analysing the breakdown of derived features into their interesting sub-regions and groups. These groups are defined in Table 4.2 in association with their gene region, database source and number of features P. As mentioned previously, a significant fraction of the features are base (80) and amino-acid frequencies (20), similar to Vogel et al [6], with a large corpus of amino-acid text-features (42) representing amino-acid interactions.

Data Group $\mathcal{G}$	**Data Region $\mathcal{R}$**	**Source**	P
Special Frequency	mRNA	Refseq	1
Mononucleotide Frequency	not protein	Refseq	16
Dinucleotide Frequency	not protein	Refseq	64
Sequence Length	All	Both	5
Gene profile	All	Refseq	5
Sequence Entropy	All	Refseq	5
Proportional Length	sub mRNA	Refseq	3
Translation Frequency	CDS/5'UTR	Refseq	2
Minimum Free Energy	5'UTR	Refseq	7
RNA misc.	mRNA	Refseq	9
Codon Bias	CDS/5'UTR	Refseq	5
Amino acid Frequency	protein	Uniprot	20
Amino acid Biophysical	protein	Uniprot	4
Amino acid text-feature	protein	Uniprot	42
PTM	PTM source	PhosphoSitePlus	8

Table 4.2: Breakdown of SDFs by data group, region and database source. *P represents the number of features in each data group. sub-mRNA in this case means any of the sub-parts of an mRNA strand, meaning the 5'UTR, 3'UTR and CDS regions.*

Genome base profiles and entropy We begin with a genome-wide analysis of base sequence profiles to discover interesting characteristics with respect to mononucleotide and dinucleotide frequency. If every gene has an mRNA sequence s_n of length L_n, then let the count of base i at position $j = 1, \ldots, L_n$ be b_{ij}. To obtain a proportion/probability we marginalize over the bases to give:

$$\hat{b}_{ij} = \frac{b_{ij}}{\sum_i b_{ij}} \tag{4.3}$$

note that the marginalization takes account of the gene length and hence we require no division in subsequent steps. Then if we assume a uniform expected distribution of bases across L_n, the observed/expected (O/E) ratio is $\chi_{ij} = (\hat{b}_{ij}/T^{-1}) = T\hat{b}_{ij}$, where T is the number of unique bases and values $\chi_{ij} = 1$ represent the expected frequency.

Figure 4.2 shows O/E across the CDS, 5'UTR and 3'UTR mRNA nucleotide (nt) positions for all protein-coding mRNAs. Here we plot the first raw 10 nt positions (A,D,G) along with the 5th to 200th nt position using a rolling average (window=10, [B,E,H]) and 200th to 5000th nt using the same rolling average (C,F,I). As expected we find ATG as the starting codon (Appendix 4.2A) for nearly all CDS sequences, with a 2:1 over-representation of G at position 3 (0-start index), +50% C at position 4 and so on. As a general trend, Thymine (T) is under-represented across the entire CDS (-10%, Appendix 4.2B), with cytosine and guanine over-represented at +10% and +15%, respectively. In regards to the 5'UTR region, C/G are over-represented at +20/30% with A/T globally under-represented, with the reverse being true for the 3'UTR region. This may be accounted by the high density of CG-rich islands that pre-dominate in the 5'UTR, known to be associated with DNA methylation. In particular, the highest C/G regions for 5'UTR are at the start of the sequence, and gradually deteriorates until reaching the CDS, whereas the A/T region climbs in strength the further away from the CDS it gets (Appendices 4.2D+). The dominance of A/C in the positions immediately preceding the TSS (see Figure 4.3B) are well known in the literature, and associated with translation initiation by aiding or hindering the ribosome subunits from binding.

From these profiles we derive two interesting features for each region $\mathcal{R}$, these are:

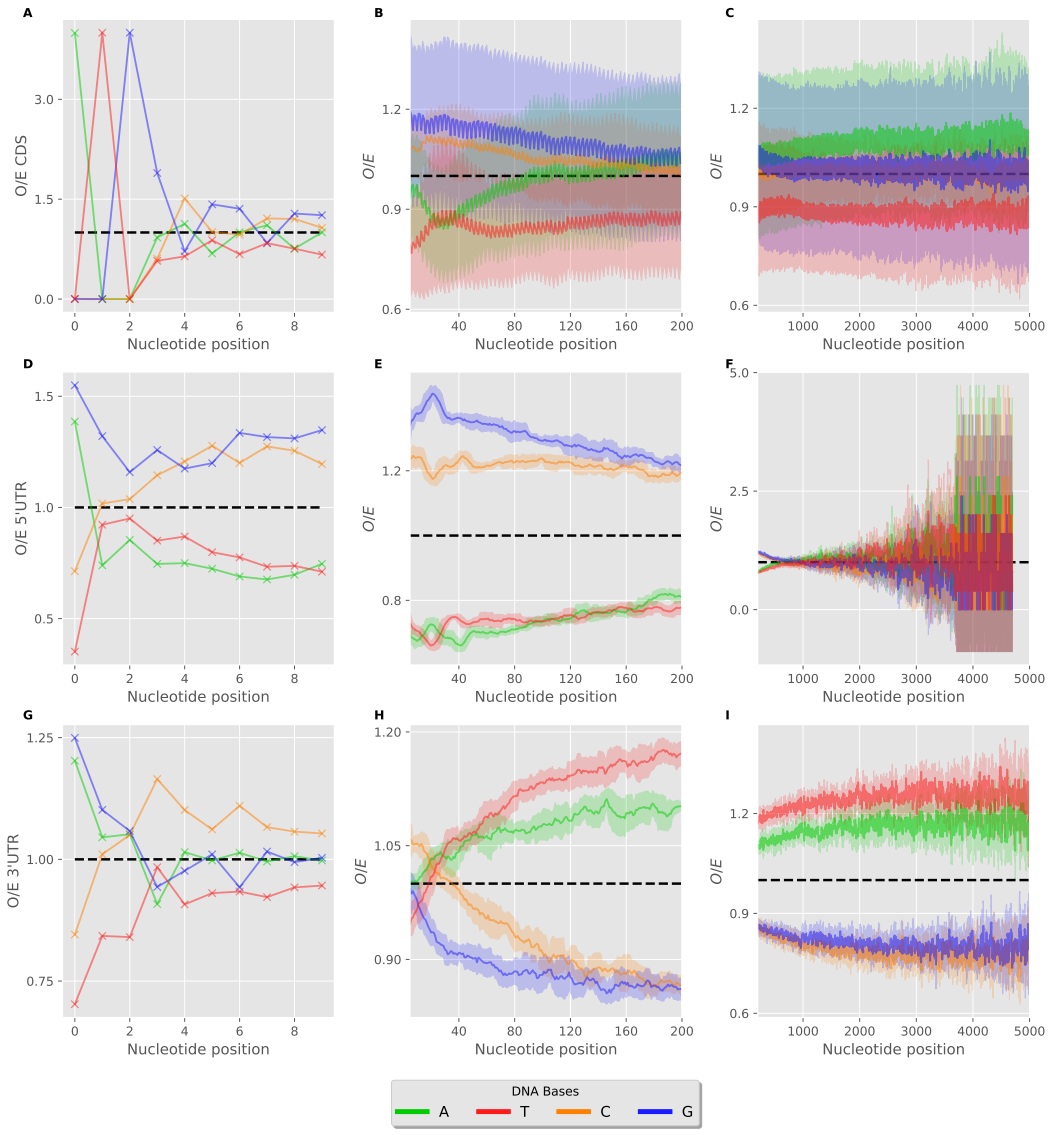


Figure 4.2: Mononucleotide observed/expected ratios by position. *Line plots of observed-over-expected mononucleotide frequencies across the human transcriptome for coding-only genes. A-C) CDS, D-F) 5'UTR and G-I) 3'UTR. Column 1 - first 10 nucleotide positions, Column 2 - 5-200 nt position with rolling mean (window=10) and rolling std, Column 3 - 200-5000 nt position with rolling mean (window=10) and rolling std.*

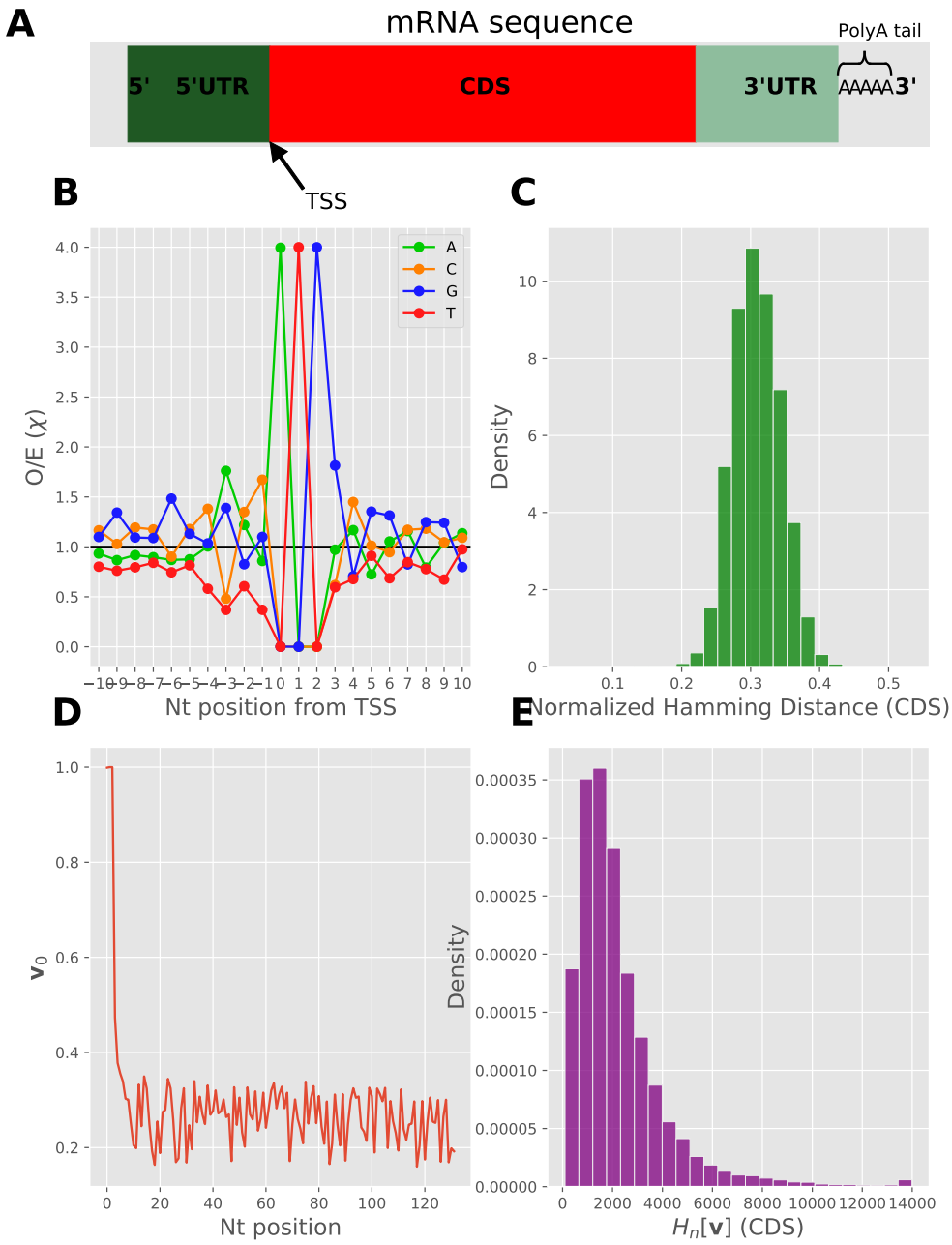


Figure 4.3: Gene profiles from mRNA sequence and associated derived metrics. *A) Overview of mRNA sequence and components. B) Lineplot of χ_{ij} by the nucleotide position distance from TSS. C) Derived Hamming distances between each sequence s_n and expected sequence $\hat{s}$ across CDS sequences. D) Proportion vector profile v_j for example CDS sequence 0. E) Sequence 'entropy' $H[v]$ distribution across CDS sequences.*

- **Normalized hamming distances**: We can create the most 'likely' or probable sequence by selecting the maximum probability at each position:

$$\hat{s}_j = \arg\max_i \hat{b}_{ij} \tag{4.4}$$

over domain $\mathcal{R}$. We can then compute the normalized Hamming distance between s_n and $\hat{s}$, where we only select up to L characters in s_n to ensure equal length. See Figure 4.3C for distribution over CDS.

- **Sequence entropy**: Using probabilities $\hat{b}_{ij}$ and sequences s_n, we can create a proportion vector profile $v_{nj} = \hat{b}_{ij}$ where $i = \arg s_{nj}$ is the selected character from the sequence. See Figure 4.3D for an example profile. We can think of the 'entropy' of this profile as the product of the probabilities over each position:

$$H[\mathbf{v}]_n = -\sum_j \log v_{nj} \tag{4.5}$$

where we compute the log-sum to avoid floating-point precision errors. See Figure 4.3E for sequence entropy distribution over CDS. We can trivially transform this to a normal distribution by taking the log-transform when needed. We can think of this metric as a measurement of how much each gene's sequence conforms to the most frequent sequence, which may aid in identifying outlier proteins.

Biophysical properties A number of significant amino acid chain properties have been studied over the years, usually derived by a combination of certain amino acids. For example, the Isoelectric point (pI) defines the pH; whether an amino acid chain has a net electrical charge. Given that certain amino acids are known to be positively/negatively charged [161], estimates of pI can be obtained assuming an immobile pH gradient. Simpler biophysical properties are metrics such as Aromaticity, which are counts of amino acids containing aromatic rings such as Cysteine and Phenylalanine, GRAVY (Grand Average of Hydropathicity) [160]; a weighted summation of hydropathic amino acids, and PEST regions which are rich in Proline, Glutamic Acid (E), Serine and T)hreonine, and whose sub-sequences are associated with protein degradation.

Codon bias Following from previous studies which found expression-codon bias relationships of interest, using the techniques discussed in section 2.1.3, we calculated various Codon bias metrics across the entire human transcriptome (see Appendix S16). Unlike the relationship between CAI-tAI within *S. cerevisiae*, codon bias inter-correlations appear significantly lower in *H.sapiens* and in some cases non-linear (see w.r.t uORF 5'UTR). However we cannot discount the possibility that increased codon bias correlations may be related to the unintentional selection sampling that occurs in smaller datasets; highly expressed proteins are much more likely to be measured in an experimental study than otherwise, and the codon-bias expression relationship becomes stronger in these samples.

Minimum Free Energy Calculations The 5'UTR mRNA secondary structure has been increasingly shown to play an important role in modulating translation initiation and elongation, amongst other roles. Methods to quantify this structure can be achieved by calculating the change in Gibbs-free folding energy, known as ΔG. More negative values indicate an increase in *in silico* stability, whereas values $\rightarrow 0$ may indicate structural instability. Given that ΔG is calculated on a base-pair basis, scaling by mRNA length is essential for multi-mRNA comparisons, see Appendix S17 for ΔG distributions. Initially we just calculated ΔG over the 5'UTR domain, but then we recognised that regions closer to the TSS would likely have more impact on translation regulation. Hence we re-calculated ΔG_w for differing window sizes w, where w is the distance from TSS. The distributions follow the Central Limit Theorem and converge to a Gaussian distribution as w increases. Note that for normalization by sequence length we take $-\Delta G/w$ as displayed in Appendix S17. We could've spent more time making more rich use of the predicted mRNA secondary structure, and this may be an appropriate use of time for future research projects, however we limited ourselves to energy calculations for this thesis.

Post-translational modifications Our PTMs are calculated from PhosphoSitePlus, an online library for mammalian PTMs [154] with over 95% of PTM sites are verified using MS experiments, with site assignments that score with a p-value greater than 0.05 are automatically filtered. The vast majority of sites discovered are Phosphorylation sites, with smaller but not insignificant amounts of Ubiquitination, Acetylation and Methylation.

4.2.2 SDF Intercorrelations Exhibit Widespread Multicollinearity

To begin understanding the possible impact of these derived features and their relative value, we compute the pairwise correlations $r_*(\mathbf{x}, \mathbf{y})$ between each SDF, however due to the heterogeneity of data type, we generalize the correlation metric to where:

- Both vectors are real $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{N\times 1}$: use Spearman-rank r_s since we cannot assume linearity.
- One real vector $\mathbf{x} \in \mathbb{R}^{N\times 1}$, one dichotomous vector $\mathbf{y} \in \mathbb{D}^{N\times 1}$: use Biserial correlation (r_{bs}, see Section 2.3.4).
- Both vectors are binary/dichotomous $\mathbf{x}, \mathbf{y} \in \mathbb{D}^{N\times 1}$: use Spearman-rank r_s.

where not stated, p-values $< 1e^{-16}$, where sample sizes $N > 1000$, and often much higher. As many of these SDFs are not entirely independent of each other, we would expect large levels of inter-correlation between features of the same type and source (see Figure 4.4); particularly prevalent are the strong positive correlations between CG-containing mRNA base ratios, and likewise for AT-containing ratios. The strongest correlations are between mono/di-nucleotide ratios by mRNA, 3'UTR and CDS regions for each gene, as well as the MFE calculations in the 5'UTR region (as the window size is all that changes). Due to the need to eliminate length bias for count-based features, normalizing by length introduces intra-correlation between features as a common factor. Furthermore, the majority of large spurious off-diagonal correlations relate to cross-talk between the mRNA/Uniprot datasets having count features with the same description: for example the 'signal peptide count' ($r_{bs} = 0.69$), 'peptide' ($r_{bs} = 0.5$) and 'transit-peptide' ($r_{bs} = 0.63$) count features had an equivalently named-feature in Uniprot. Future subsections within this chapter will attempt to tackle this induced multicollinearity using feature selection and/or dimensionality reduction methods. In general, the relationship between CDS and amino-acid derived features was very strong, this is largely because of the linear 3:1 mapping between codons and amino acids as determined by the genetic code; biophysical properties also correlated well with amino-acid/CDS frequency ratios. These findings are consistent to previous work done by Vogel et al. [6], albeit with a slightly

more varied feature set; feature correlation overall tends to cluster by gene region with the exception of base frequencies and favours features that are biologically closer to it, as we successfully demonstrate (see Figure S18). However, Vogel's group did not consider features derived from text-based information or make use of meta information provided in these databases, as many of them were still under development at the time of research.

The magnitude of each feature to correlate with all of the others is calculated as the ℓ_1-norm of each correlation matrix row $j \in P$:

$$||r_*^{(p)}||_1 = \sum_{j=1}^{J} |r_{pj}| \tag{4.6}$$

where large $||r||_1$ indicates redundancy, but low $||r||_1$ may be non-linearly related, irrelevant or relevant to expression but not to other SDF features (see Figure S19). Generally, mRNA, CDS and 3'UTR dinucleotide features have high intra-correlation, particularly base frequencies that consist of GC/AT-only bases. Of codon biases, which historically have been used extensively in previous studies as effective mRNA proxies, Codon Adaptation Index (CAI) scores highly, but other metrics do not.

Non-linear relationships between SDFs The most popular measures of correlation include Pearson and Spearman-rank correlation, but both of these methods depict either linear or monotonic relationships respectively, and struggle to model non-linear relationships. Therefore to capture any potential nonlinear relationships, we computed the continuous Mutual Information (MI) metric between all SDFs, treating each vector pair as random $i.i.d$ variables X and Y:

$$\mathbb{I}(X;Y) = \mathbb{KL}\left[p(X,Y)\,||\,p(X)p(Y)\right] \tag{4.7}$$

where $\mathbb{KL}$ is the Kullback-Leibler (KL) divergence, also known as relative entropy. KL divergence can be computed via estimates of the Shannon entropy $H(\cdot)$ as:

$$\mathbb{I}(X;Y) = H[X] + H[Y] - H[X,Y] \tag{4.8}$$

where $H[X]$ is the marginal entropy for X and $H[X,Y]$ is the joint entropy over X, Y. Estimates of entropy in practice requires discretizing continuous random variables X and Y into bins k and evaluating the density of each

Figure 4.4: Large-scale interdependencies between SDFs reveal source dependency. *Spearman-rank (r_s)/Biserial correlation (r_{bs})-mixed correlation matrix between sequence-derived features (SDFs). See Methods for details on correlation method. Both axes indicate the direction of molecular biology (from DNA to post-protein). mRNA/RNA features are denoted in green shades, amino-acid features are denoted in red shades, PTM features are denoted in purple. Off-diagonal elements indicate a correlation (r_s—r_{bs}) between two features (red: positive, blue: negative).*

bin (here we use $k = 20$). Using this metric, plot the pairwise 'mutual information matrix' between each continuous SDF feature (see Figure S20), but similar to wholesale correlation matrices with large P, it is difficult to ascertain interesting relationships beyond the general clustering of mRNA-base count-like features. To account for feature pairs that may contain a non-linear component not picked up by correlation metrics, we developed a quadratic model ($r^2 = 0.91$) between mutual information and correlation $\mathcal{E}_{\mathbb{I}(X;Y)|r_*}$ for all $X \neq Y$:

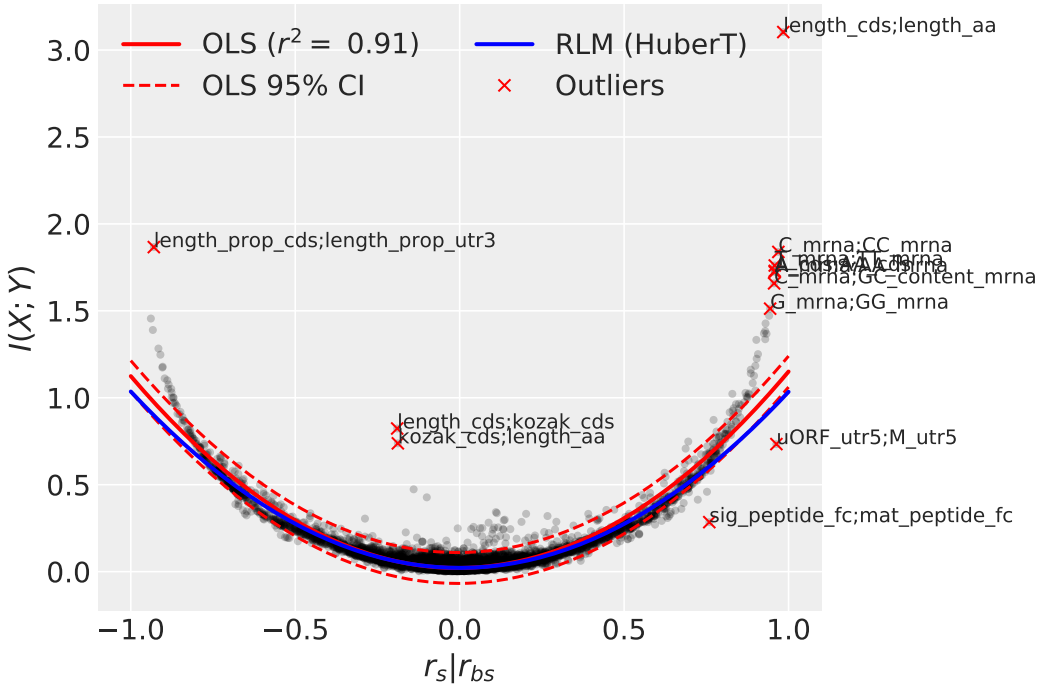


Figure 4.5: Quadratic modelling of linear/non-linear dependence between SDF pairs. *Scatterplots of Spearman-rank (r_s)/Biserial (r_{bs})-to-Mutual Information ($I(X;Y)$) pairs. Ordinary least squares (OLS) modelling as eqn. 4.9 in red with 95% confidence intervals (CI), against Robust Linear Model (RLM) using robust norm (Huber-T) in blue. Selected outliers (top 10 large plus 2 smallest) marked in red crosses and labelled.*

$$I(X;Y) \approx w_0 + w_1 \mathbf{r}_*(X,Y) + w_2 \mathbf{r}_*^2(X,Y) + \boldsymbol{\epsilon} \tag{4.9}$$

hence pairs that have $\epsilon_n \to 0$ are equally well explained by linear metrics as non-linear metrics, with points with large ϵ_n as outlier correlations of interest (see Figure 4.5). The vast majority of outliers are cases where r_* is underestimated in value. In particular, length-to-length features such as amino acid length and CDS length appear to strongly correlate both in linear and non-linear metrics; a surprise was discovering kozak sequences appearing more strongly as non-linear association to length which was not picked up by correlation metrics ($r_s \approx$ -0.2). We also sampled two outliers that were slightly overestimated by r_*: one case of methionine frequency to uORFs in 5'UTR region, and signal peptide to mature peptide. In both cases the strong dependency is obvious as both contextual domains heavily overlap. We further check to see if features divide by gene region $\mathcal{R}$ (mRNA, 5'UTR etc.) across the $\{r_*, I(X;Y)\}$ domain. We can compare groups by taking the kernel-density estimate (KDE) of each regions' Gaussian PDF in the form $r_*(\mathbf{x}_p, \mathbf{x}_q) \in R_{pq}$ (see Figure S21). The density of SDF pairs from the same region $R_p = R_q$ tend to have a significantly flatter KDE estimate, indicating significantly higher positive/negative correlations among features from a similar region. Most protein KDEs exhibit the lowest correlation PDFs, which may be a reflection of the large body of dichotomous text-mined features from the amino acid sequence which tend not to correlate with many other features (see correlation matrix in Figure 4.4), and the relative distance these features have to mRNA and sub-mRNA feature regions.

Comparison to Vogels' feature set To validate the derived features and to benchmark as a comparison, we extracted the mRNA sequences and SDFs as provided by Vogel's work [6], see Supplementary 6.2 for details on the methodology. In the next section we'll also extend this comparison when exploring the impact on expression levels. To achieve this, we used two major forms of analysis for feature comparison:

1. **Sequence alignment**: To check whether we were actually using the same sequences to derive information from, we performed pairwise local sequence alignment using Biopython [142] across CDS, 5'UTR and 3'UTR sequences; we then sort mean pairwise alignment scores A_n across the data percentiles (see Figure 4.6A). We approximated the

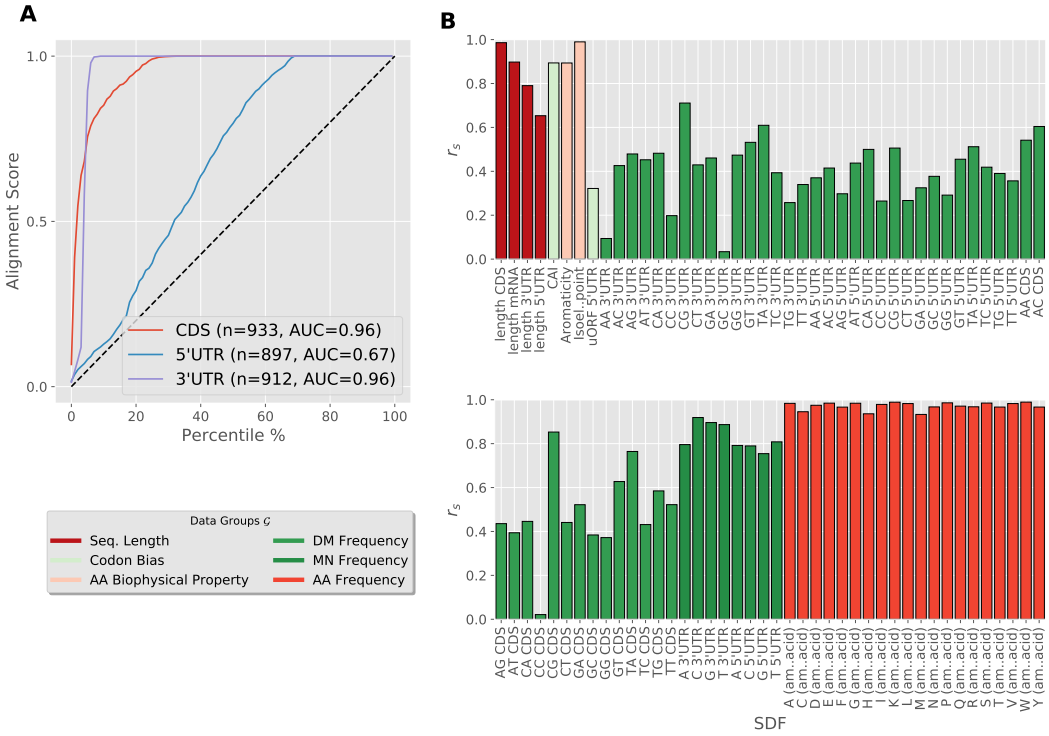


Figure 4.6: Sequence and SDF comparisons to Vogel et al. [6]. *A) Lineplot of percentile pairwise sequence alignment scores across differing mRNA sub-regions. B) Spearman-rank r_s correlations between features extracted by Parkes and Vogel ($P = 84$).*

area-under-curve integral (AUC) using the composite trapezoidal rule with K samples/percentiles and fixed step size $\Delta A = 1$ as:

$$\int_0^1 f(A)\mathrm{d}A \approx \sum_{k=1}^{K} \frac{f(A_{k-1}) + f(A_k)}{2} \tag{4.10}$$

where $f(\cdot)$ can be any necessary transforming function and $K = 100$. The CDS and 3'UTR regions have a 96% area-under-curve (AUC), but 5'UTR only has 67% AUC. We found using global alignment that around 10% of 3'UTR sequences contained a sub-alignment where one sequence had a large addition that would lead to significant negative scores (around -0.5/8). Hence we decided to use local alignment to ignore these issues and focus on checking whether any alignment was possible. There were very few significant mismatches in the CDS sample - with the lowest 20% percentile containing at least an 80% match; this could be attributed to modifications within the database between the periods of our research and the research conducted by Vogel [6]. Somewhat more surprising was the increased lack of agreement in the 5'UTR sequences; in most cases there is a large disparity in sequence length between databases; and indeed in the majority of discrepancies, our sequence lengths are larger than Vogel's (see Appendix S23). There are a number of possible explanations for this including 1) unknown selection or reduction procedures applied to the sequences within Vogel's research, 2) discrepancies between Refseq and Ensembl databases, 3) changes to the underlying sequence database across time, 4) errors in our calculation with regards gene region boundaries, or 5) an meaningless artefact of working with a substantial subset of the whole transcriptome (50k). It is likely that many or all of these explanations contribute some small part to the discrepancy in sequence.

2. **Derived-feature correlations**: To compare SDFs, we computed the pairwise Spearman-rank r_s correlation between each Parkes-Vogel feature where the feature was deemed the same (see Figure 4.6B, $p = 84$). Similarly to the results from sequence alignment, the amino acid-based features have very high correlations, but mono and dinucleotide frequencies are somewhat poorer, with CDS correlations performing the best owing to having the most similar sequences. To overcome the possibility of having various non-linear transformations to one of the

features skewing the correlation metric, we used three different transformation techniques (see Supplementary 6.2) and then selected $\max r_s$ over the transformation types. Once again, with a relatively small sample $n = 453$, it is unclear whether these correlations would be representative of the wider mRNA population.

We also performed some systematic analysis of pairwise SDFs in the form of Canonical Correlation Analysis (CCA) between the two datasets (see Figure S22). We discovered there was very limited overlap in the feature space, which we suspect is due to the time difference between the studies, and thus the evolution in the underlying genetic databases which form the basis on which SDFs are extracted. This remains true when we filter for just the subset of genes chosen in Vogel's sequence feature set. It is wholly possible however that, as a number of features break inter-independency assumptions, that statistical insight from this check may be questionable. We have in this previous study a relatively small but useful benchmark to compare our features against; the value of which won't be particularly clear until they are used for predicting useful biological properties such as expression level, gene function, interaction and so on. Our subsequent analysis is to tackle the problems arising from multicollinearity using an array of feature selection and unsupervised learning techniques, and the trade-offs therein.

4.2.3 Systematic Unsupervised Learning Approaches Trade-Off SDF Performance Against Interpretability

Many approaches to modelling biological and other problem domains involves creating a simple model and slowly adding relevant features until some saturation point is reached, which we can think of as a *bottom-up* approach. Our methodology is to create and design as many features as possible, build a complex model and then prune back the feature set to prevent overfitting and multicollinearity, which is a *top-down* approach. To address the concerns regarding feature viability, multicollinearity and overfitting, we devise a systematic pipeline to transform feature inputs $\mathbf{X}_P \rightarrow \mathbf{X}_K$ into a smaller subspace that is suitable to ML modelling. See Supplementary Table S2 for an overview of a large body of feature selection and dimensionality reduction techniques. For subsequent analysis we assume we are working with $\mathbf{X}^{(2)}$, i.e the normalization 2 input matrix, unless specified.

Value Filters The simplest method to drop unhelpful features is simply to apply some criterion, such as the number of missing values, variance σ^2 or correlation r_* and filter columns by this metric. Unlike most large datasets of this kind, our SDFs have significantly low numbers of missing values and only tAI has more than 5% missing values among a feature set of $P = 194$. One could use imputation to fill the remaining values, but we leave as-is to allow flexibility for future application. In terms of variance $\sigma^2 < \eta$, a number of *feature counts* associated with mRNA have very low variance $< 1e^{-6}$, mainly due to normalization by gene length often giving very low values, eliminating around 6 features. We briefly explore correlation filtering, but since there is heavy overlap between the principles in this filter and more advanced dimensionality reduction techniques, we do not consider them in detail. Similarly, we do not cover ANOVA as the F-value is a simple conversion from a correlation between each input $\mathbf{x}_p$ and a target $\mathbf{y}$; ANOVA also requires strong linearity assumptions and Gaussian-distributed features.

Feature Selection Rather than drop unhelpful features using various criteria, we can take a more active approach by iteratively developing more complex models which we then evaluate to minimize the loss. This approach has the significant drawback of requiring a target variable $\mathbf{y}$, i.e a supervised approach, and hence must be performed on a per-application basis. The key advantages are the substantial reduction in $P \rightarrow K$, flexibility and interpretability of results. In the following examples, we use as an example the target protein expression $\mathbf{p}$ from Vogel et al [6]. For instance, we could use the feature selection inherent within ℓ_1-norm based models such as LASSO to select for non-zero feature coefficients (see Appendix S24A-B). In these models we firstly select an appropriate regularizing hyperparameter $\hat{\alpha}$ by minimizing equation 2.19 to find a local minima, and verified this using cross-validation over an appropriate parameter space. In this case, we found 41 non-zero coefficients out of a possible 194 features. As previously mentioned, not only does this require a target, but the feature selection can be rather inconsistent - this is especially true if the data is slightly perturbed. One common method around this is to undergo bootstrap sampling using different data subsets and estimate feature inclusion frequency. Furthermore, if multiple variables are highly correlated as is the case with our dataset, LASSO tends to select only one of them arbitrarily. We considered using a Group Lasso model which imposes hierarchical structure on the features, but it is not clear what

is the most appropriate way to group our sequence-based features. Other techniques that are more intensive are Recursive Feature Elimination (RFE) algorithms that recursively prunes the worst features with each model fitting (see Appendix S24C). Here we compare against OLS, Ridge and LASSO models, where $\hat{\alpha}$ is determined by minimizing the respective objective functions (eqs 2.17, 2.19) respectively. The general consensus across all three models is in the range $P = [28, 32]$. We also considered the Greedy Forward Feature Selection (GFFS) approach as done by Gunawardana et al [3], which additively adds features based on the cross-validated RMSE at each iteration; this yields similar results to RFE.

PCA Here we describe PCA within a probabilistic framework by describing the latent variables $\mathbf{z}$ as conditioning the centered inputs $\mathbf{x}$ in the following likelihood function:

$$p(\mathbf{x}_n|\mathbf{z}_n) = \mathcal{N}(\mathbf{x}_n|\mathbf{W}\mathbf{z}_n, \sigma^2\mathbf{I}) \tag{4.11}$$

where we define the prior of $\mathbf{z}$ to be $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I})$, with noise modelled as a single parameter $\sigma^2\mathbf{I}$; see section 2.3.5 for more details. As first instance we compute $\mathbf{W}$ using the whole dataset $\mathbf{X}^{(2)}$, and then compute the explained variance ratios (EVR), which are simply normalized singular values λ_i as computed by SVD. Here we plot the cumulative EVR across K (see Figure 4.7A). One striking aspect is the relatively slow pace of variance compression within the dataset; contrasted to for instance image processing where variance is preserved in significantly fewer dimensions. We only reach 80% total variance at 46 (principle) components and 95% variance with 93 components, with an AUC as estimated by the composite trapezoidal rule (eqn 4.10) of 87%. Based on the performance, we highlight the 80-99% region (blue box) as reasonable candidates to select optimal K, the trade-off of which we will consider in later chapters. PCA does not handle categorical/binary features very well, as shown in Figure 4.7B; all outliers belong to the amino-acid meta information category across the first two components, and this trend holds across most eigenvectors (Figure 4.7D). To model the outlier eigenvectors, we used a t-distribution to estimate $\mathbb{E}[\mathbf{W}_k]$ for all $k \in K$ (green lines), along with 1% and 99% percentiles (see example eigenvector $\mathbf{W}_1$ in Figure 4.7C). As we see a normal distribution is inappropriate due to the undue influence of outliers in most $\mathbf{W}_k$. One of the main reasons for model failure in this instance is the assumption that the noise variance σ^2

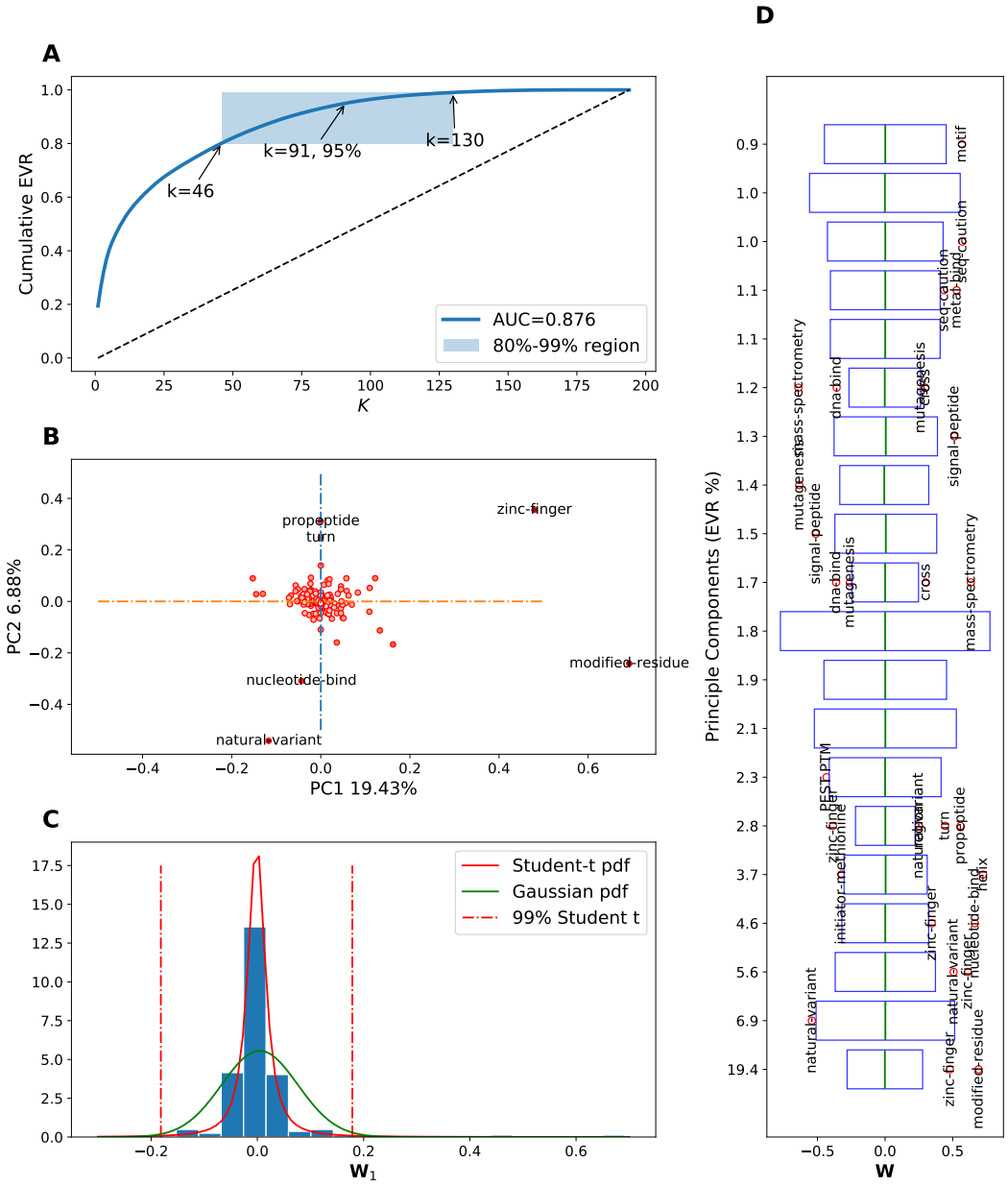


Figure 4.7: PCA analysis on whole SDF dataset. *A) K components against cumulative explained variance ratio (EVR), representing normalized singular values λ_i. Box selection indicates likely region to select optimal K. B) First two principle components* $\mathbf{W}$ *and coloured based on magnitude from origin. C) Histogram of* $\mathbf{W}_0$ *with fitted Gaussian and Student-t distributions. D) First 20 principle components in pseudo-box plot form, with bars representing 99.5% percentiles of all data. Outliers w.r.t percentiles are drawn as scatter points.*

is the same for every feature; this is not true among continuous features, let alone the categorical ones. Furthermore, the removal of interpretability in components makes wholesale PCA a poor option in terms of gleaning meaningful insights into SDF feature value. However one important lesson stands out from this: the compression of biological SDFs is a non-standard problem, indicative of the nonlinear, interactive components that describe the cellular environment which we seek to encapsulate. We can solve one of these problems by considering Factor Analysis (FA) by allowing for heteroscedastic noise.

Factor Analysis (FA) Recall that for FA we model the centered SDF matrix $\hat{\mathbf{X}}$ as containing a hidden optimal subspace using latent variables $\mathbf{z}_n \in \mathbb{R}^L$ with added Gaussian noise, giving the following likelihood function [162]:

$$p(\mathbf{x}_n|\mathbf{z}_n) = \mathcal{N}(\mathbf{x}_n|\mathbf{W}\mathbf{z}_n, \mathbf{\Psi}) \tag{4.12}$$

We can think of FA as a low-rank parameterisation of a Multivariate Gaussian, requiring that all features correspond to a normal distribution; this means that FA can only be applied to our continuous features. We can compare how each feature $\mathbf{x}_p \to \mathcal{N}(\mu_p, \sigma_p^2)$ maps to a normal distribution by computing the normal theoretical quantiles, also known as Quantile-Quantile (QQ) plot. Graphically sampled theoretical quantiles will lie on the $y = x$ plane, leading naturally to use Pearson's correlation coefficient r_p as a measure of similarity. We choose $r_p > 0.95$ to select 110 features which are deemed as *normally distributed* out of 150 continuous features.

The first latent factor corresponds to the divergence between CG-rich (positive) and AT-rich (negative) SDFs (see Figure 4.8A), which form strongly co-correlated clusters that require elimination. The second latent factor primarily contains strong negative weighting to length and entropy-based features, with the third focusing on proportion length features, fourth for amino acid frequency and biophysical properties and fifth MFE energy calculations (see Figure 4.8B). Factors in $\mathbf{W}_k$ are grouped by their data group $\mathcal{G}$, and we compute the scaled ℓ_1-norm $||\mathbf{W}_k^{\mathcal{G}}||_1$, where we use the absolute to ignore factor direction to prevent negative and positive factors within the same group averaging to zero and obscuring the magnitude. For Figure 4.8C we use the same schema for data regions $r = 1, \ldots, R \in \mathcal{R}$. Apart from in Factor 1

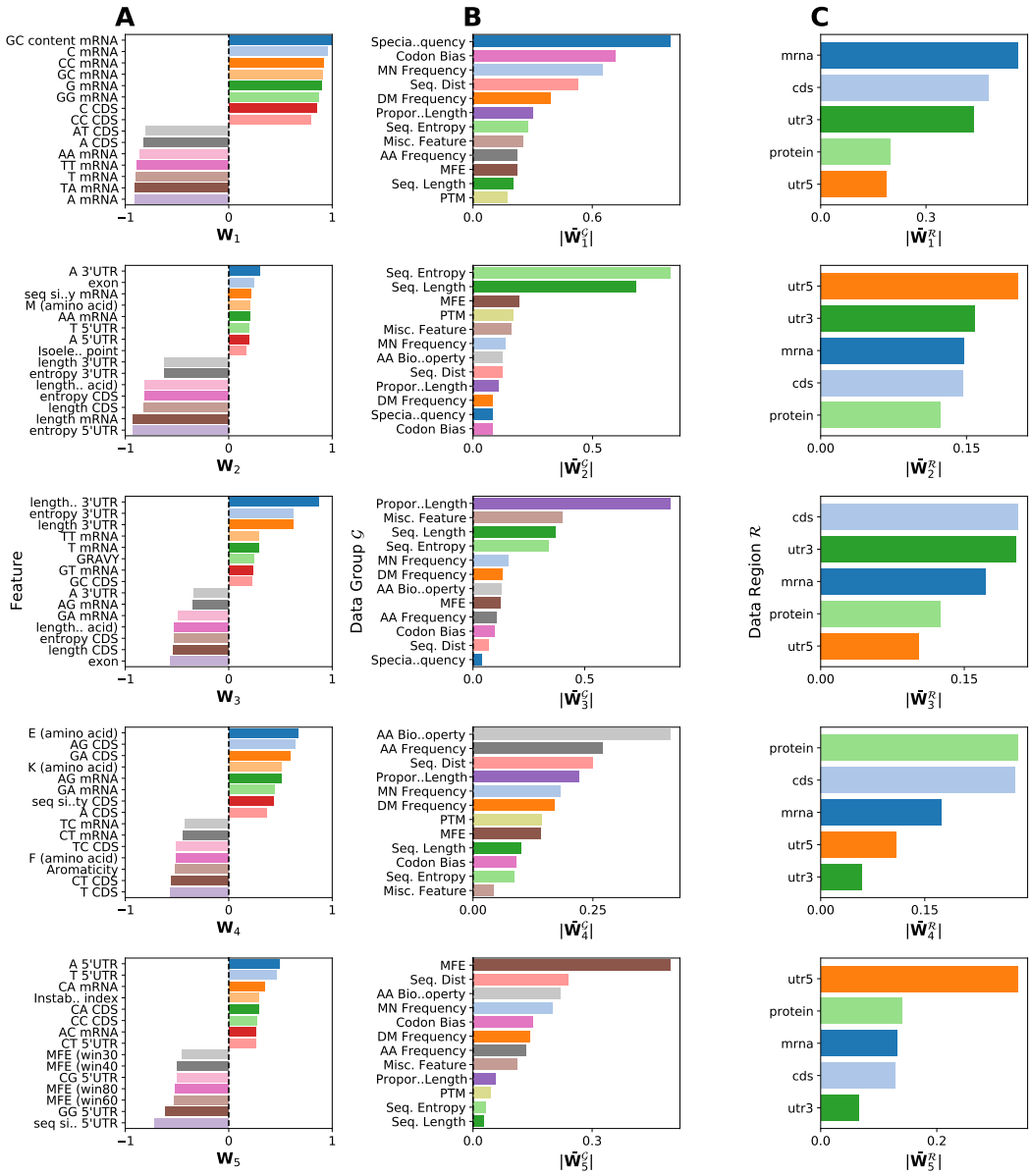


Figure 4.8: Factor Analysis of SDF feature matrix. *Barplots of A) Highest scoring features with factors* $\mathbf{W}_k$ *(1-5) for first 10 features. B) Absolute Grouped factors* $\frac{1}{G}\sum_g^G |\mathbf{W}_k|$ *by data group* $\mathcal{G}$*. C) Absolute grouped regions* $|\sum_r^{\mathcal{R}} \mathbf{W}_k|$ *by data region* $\mathcal{R}$*.*

where mRNA-based features are more selected for, we do not see many differences in terms of how each data region is treated by FA. Although one has to treat interpretation of $\mathbf{W}$ with caution, as the rotation matrix $\mathbf{R}$ applied to $\mathbf{W}$ to induce factor identifiability can be arbitrary (in this case we use *varimax*). Similarly, the features $\mathbf{\Psi}$ with most noise as determined by FA are mostly protein-related features, such as PTMs (0.59) and AA frequency (0.37), with mRNA misc. features (0.57) also scoring highly. Whilst the multivariate normal (MVN) assumptions required for FA significantly reduce K, the representation of noise variance for each feature in $\mathbf{\Psi}$ allows for heteroscedastic noise modelling, which leads to significant reductions in K while preserving maximal variance.

As described by Bishop and Tipping [122], we can calculate the negative log-likelihood for samples by drawing from the posterior distribution of the latent variables $\mathbf{z}_n$ to get:

$$\text{NLL} = -\sum_{n=1}^{N} \log \mathbb{E}[\mathbf{z}_n] \tag{4.13}$$

Comparing FA against PPCA, we see that the negative log-likelihood (NLL) across samples is substantially higher for FA than PPCA (see Figure 4.9A), with optimal K reached significantly faster than PPCA. By comparison, we also compute NLL for covariance matrices with shrinkage by Ledoit-Wolf (L-W) [117]. Furthermore, the linear increase in NLL for PCA compared to non-linear increases with FA further emphasises the importance of heteroscedastic noise modelling as a means for effective dimensionality reduction.

Stratified PCA (sPCA) Instead of performing PCA on $\mathbf{X}$, we stratify features P into G groups according to data groups $\mathcal{G}$ or regions $\mathcal{R}$. We can then perform normal PCA on each subgroup and combine the results together. For each PCA we use maximum likelihood using Minka's estimate [125] for automatic selection of K, or manual selection via an appropriate variance threshold η over the explained variance ratio (EVR) (see Figure 4.9B), where in general for low thresholds, one feature for each group is preserved, leading to the offset in K and the increased conservative nature of sPCA. If we break down EVR by data groups $\mathcal{G}$, there is a variety in the ability of PCA to obtain near to 100% variance in less than the maximum di-

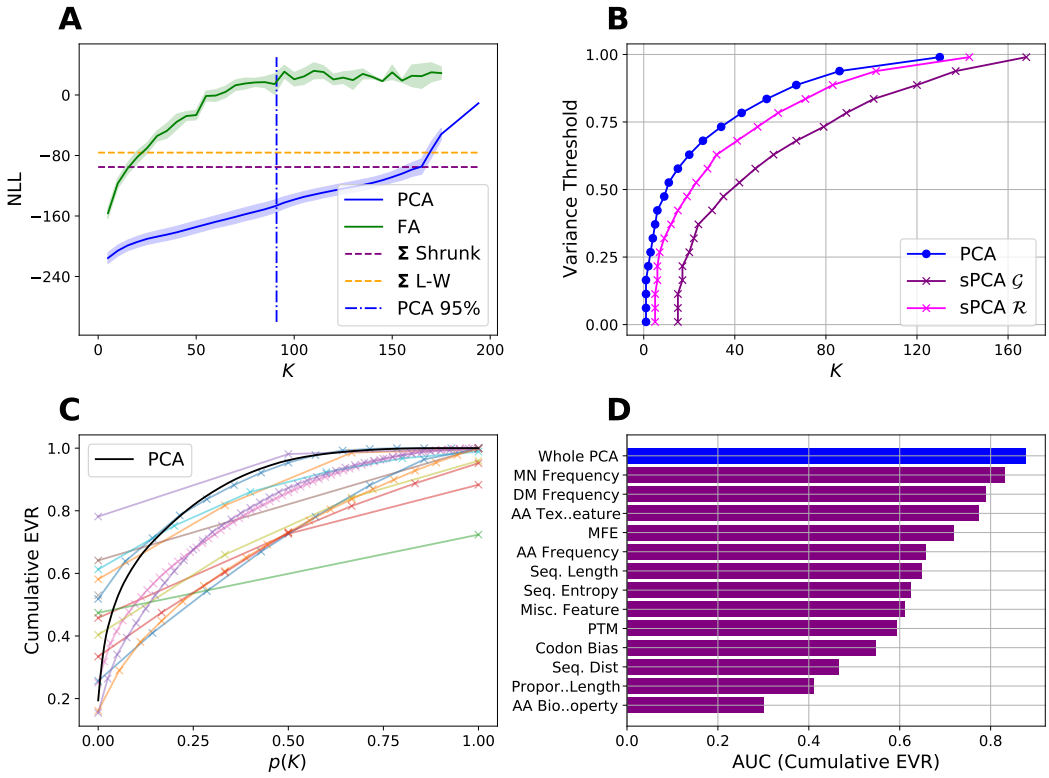


Figure 4.9: Stratified PCA analysis on SDF features. *A) Comparison between PCA and FA on log-likelihood (NLL) for different selected K, with fixed covariance and Ledoit-Wolf [117]. B) Changes in K as variance threshold between [0..1] for PCA and sPCA. C) sPCA (by data groups $\mathcal{G}$) explained variance ratios (EVR) as a proportion of K with whole PCA in black. D) AUC of cumulative EVR for whole PCA (blue) and sPCA (purple) by data groups $\mathcal{G}$.*

mensionality (see Figure 4.9C-D). In most of these cases we do see whole PCA do better than sPCA in performance, but we are trading off performance for a significant boost in interpretability, as log-likelihood samples, AUC and EVR can all be applied on a per-group basis rather than on abstract principle components. Groups such as mono- and dinucleotide frequency have the highest AUC scores when computed on cumulative EVR, illustrating a successful compression of these feature groups into a more compact domain. However certain groups such as amino acid biophysical properties and codon bias contain multiple different types of features within the same group, leading to a significant drop in compression. It is also the case that manual selection of groups $\mathcal{G}$ may mean that highly correlated features are not always grouped together, leading to some residual inter-group collinearity.

Multiple correspondence analysis MCA can be thought of as categorical PCA; and hence for working with large-scale categorical datasets [163]. Here we deploy MCA use for non-continuous variables by constructing dummy variables using one-hot encoding on each categorical feature prior to fitting.

Other We also perform exploratory analysis on manifold techniques such as t-distributed stochastic neighbour embedding (t-SNE) and Isomap. Evidence from the literature indicates that these methods are primarily used for visualization within two dimensions, and their complexity and non-linear mapping make them undesirable for preserving interpretability in subsequent analyses. Hence we do not consider their role any further within this research.

Overview Here from the above methods we construct the following numbered examples of reduced subsets, as $\mathbf{X}_P \rightarrow \mathbf{X}_K$, and following from previous notation, we will describe the transformed matrix $\mathbf{\Phi}^{(m)}$ of size K where ^{m} denotes the following:

1. Whole PCA with $P = 194$ using a variance threshold η of 95%.
2. FA with $P = 111$ selecting normally distributed (MVN) features, using varimax rotation.
3. sPCA using data groups $\mathcal{G}$, with each PCA using $\eta = 95\%$.
4. sPCA using data regions $\mathcal{R}$, with each PCA using $\eta = 95\%$.

5. PCA using $P_c = 151$ continuous features with $\eta = 95\%$, coupled with MCA $P_d = 43$ discrete features.

6. FA model from (2) using MCA model from (5).

7. sPCA model from (3) using MCA model from (5).

Each of these models preserves just over 15k samples, with varying degrees of K (see Appendix S25). We will refer to these examples in the next chapter.

4.3 Discussion

Sequence-Derived Feature Extraction and Formation

In this chapter we have devoted a significant amount of resources, attention and focus of this thesis to the discovery, curation and collation of sequence-derived features from mRNA and amino-acid transcripts into a more friendly tabular format [1]. We consider these features in this form a contribution to the body of knowledge given the complete coverage of the human genome and their versatility in application to a number of potential problems. Since the use of these features are not equivalent to deploying pre-trained models, many of the false assumptions that would lead to concern using pre-trained models can be disregarded in this instance, allowing for future research to be carried out with increased certainty. The disadvantage of this approach is the increased complexity required when performing the modifications necessary to align the dataset to a custom application. The extraction of these features from various databases and formats was non-trivial, with assumptions made particularly when it came to integrating across the mRNA-protein domain. A particular challenge when performing multiple intersection set operations across various database labels (Uniprot IDs, Refseq IDs, HGNC labels) is the preservation of large N; essential to maintaining significant coverage over the 'omic domains and retaining statistical significance. This proved difficult due to inconsistent naming conventions and version numbers attached to the names, though alias names are sometimes provided. These assumptions are documented within the relevant Jupyter notebooks for future readers. Furthermore, can lead to further problems from a future research perspective, as highly-cited proteins and mRNAs tend to have more consistent naming conventions and integrate better into multi-'omic datasets; leading towards

a bias that skews towards less novel/interesting protein analysis.

Another major assumption when extracting SDFs from various databases within a multi-'omics context is the handling of sequence variants. For example, a particular gene may have one or more mRNA transcript variants where a couple of base pairs are altered. These changes may lead to negligible differences statistically, but in practice small changes in the coding region or 5' untranslated region can have a dramatic impact on gene expressivity. Our approach was to always take the longest curated transcript wherever multiple of such transcripts existed, as many other authors have done, but theoretically there may be valuable information lost when making this assumption. Practically however, the potential explosion in data size particularly when attempting to integrate with amino acid sequences and protein expressions leaves little room to adopt a more variant-friendly approach; however future research potential with more computational resources are available could consider a more complete systems-wide approach that did not remove alternative variants.

Top-Down and Bottom-Up Modelling Perspectives

Simulation is often described as the third pillar of science (after observation and experimentation), coming with its own philosophical underpinnings and assumptions that we must consider to fully utilise a sequence-derived feature modelling approach. Whilst this field is rather vast, here we will focus between the two diametrical modelling approaches; bottom-up and top-down modelling.

Bottom-up In this approach, models are sequentially built upwards from simpler to more complex ones. This approach is common with researchers from a strong mathematical or engineering background. due to the problems associated with *curse of dimensionality* as the parameter/variable space expands. Practically, a classic example would be a greedy forward feature selection algorithm which greedily adds the best performing feature at each time step t to a linear regression model. Another example would be constructing bottom-up a probabilistic Bayesian model around some data $\mathcal{D}$, due to the cost of integrating out all other variables to compute each marginal probability. This approach provides advantages in aligning with the Occam's razor principle, better interpretability and mathematical robustness. Some of the

drawbacks are usually lower model performance compared to top-down in practical applications, poor selection in features and the assumption that just a few features are powerful in predicting a suitable target variable.

Top-down Alternatively, complex models can be initially constructed from many features and then pruned significantly down to a simpler model. This is one of our key *a priori* assumptions; that a suitably complex feature set can be fitted and efficiently pruned down to a generalizable model that can predict protein abundance or function. The main advantages to this is better model performance and better feature space coverage, with the drawbacks being more difficulty in model interpretation and risks of overfitting. In this section we reduce overfitting risks using an adjusted scoring metric for r^2, using dimensionality reduction and hold-out sets. We reduce the interpretation risk by considering different dimensionality approaches and trading off accuracy versus verbosity. Some of the models we use (such as MARS [103]) deploy both bottom-up and top-down strategies to iteratively build up and prune away excess within the coefficient space.

Unsupervised Learning on Sequence Information And Considered Trade-Offs

The tradeoffs between computational performance/dataset size and interpretability with respect to dimensionality reduction are well documented, and biological sequence-based features are no exception. Unlike embarrassingly large dimensionality domains such as image processing with huge redundancy, we see no such pattern within DNA sequence information - in fact complex models such as gradient-boosting regression trees actively prefer raw sequence data as opposed to a preprocessed matrix that has underwent dimensionality reduction. We argue within this Thesis and first published paper [1] that the small, cumulative impact of many features is required for decent model score performance; given the regulatory complexity that affects a given proteins' abundance.

Chapter 5

Multilevel modelling of protein abundance by sequence-derived features

In application to the problem of protein prediction via mRNA proxy, an interesting question now arises as to whether information derived from static sources (i.e the DNA sequence) can bare any relationship to experimental expression datasets, with the attributable noise that comes from space-time measurements. In this chapter, we focus once again on protein abundance prediction, except this time we perform full-pipeline model and feature selection, and determining the most important features within the SDF dataset. We consider biological explanations via Gene Ontology analysis as to which sequence-based features are important; and in particular explore the interesting relationship between sequence information and translation. Finally, we expand the sphere of knowledge to include protein-protein interactions and steady-state half-life of mRNA and protein expression as useful feature inputs to consider in conjunction with SDFs. Parts of the research in this chapter is under review with Nucleic Acids Research (NAR), with additional supplementary material and tangents.

5.1 Data Preparation

Here we describe the collection of mRNA-protein expression datasets used in this analysis alongside those previously used by Aviner [5]. Here we do not

Cell Line	Tissue	Source
U2OS	Bone Osteosarcoma Epithelial	Lundberg et al. [164]
U251MG	Glioblastoma	Lundberg et al. [164]
A431	Epidermoid Carcinoma	Lundberg et al. [164]
HeLa	Cervical Carcinoma	Aviner et al. [4, 5]
Daoy	Primary Medulloblastoma	Vogel et al. [6]

Table 5.1: Cell Lines and associated tissues.

describe extraction/preprocessing steps with respect to SDF features, as this is detailed in the previous chapter, and within Supplementary Material.

Expression Datasets Human HeLa cell cycle data was taken from Aviner et al. [5] with triplicative expression measurements for mRNA, translation and protein. Microarray data is taken from the Gene Expression Omnibus (GSE26922), parsed using the GEOparse package. Protein levels for HeLa are pre-normalized using intensity-based absolute quantification (iBAQ) [139]. U2OS, A431 and U251MG cell line expression data for mRNA and protein expression is taken from Lundberg et al. [164], whose RNA expression is estimated using RNA-seq, whereby RPKM values [165] were calculated for each RNA (see NCBI short-read archive with accession number SRA012517). Daoy cell line expression data and sequence-derived features are taken from Vogel et al. [6], whose cell line is cultured, collected and described previously [166]. Gene expression values are estimated using Robust Multi-Array (RMA) analysis [167]. Protein expression for all cell lines is estimated using MS/MS (Aviner, Lundberg) or LC-MS/MS (Vogel). We also make use of the PAXDB protein database [168] which provides among other things a consensus global protein abundance across the *H.sapiens* proteome which averages over many cell lines. PAXDB processes protein abundance in parts-per-million (PPM) to allow for inter-species comparison. See Table 5.1 for a summary of the cell lines used in this study.

Halflife Data To examine the impact of mRNA and protein degradation, we take HeLa cell mRNA half-life data from Tani et al. [169], removing missing values and any values >24h. We also split mRNAs into protein-coding and non-coding subsets. HeLa cell Protein halflives and k_{deg} decay

constants are taken from Cambridge et al. [170] and minimally preprocessed.

Protein-Protein Interactions Information on protein-protein interactions (PPI) was obtained from STRING database [171], where we downloaded all 11.7 million *H. sapiens* interactions using the full links option. We also access protein information such as display names and descriptions. We used the NetworkX package [172] to build a node-edges graph and calculate metrics such as degree and centrality. Note that we do not model the interaction network directly, only to extract tabular metrics for each protein.

Notation Recall that for an unnormalized design matrix $\mathbf{X}$ we used the notation $\mathbf{X}^1$ to signify normalization 1 from the previous chapter. Now using dimensionality technique m we reduce $\mathbf{X} \to \mathbf{\Phi}$, where $\mathbf{\Phi}^{(m)}$ refers to reduced SDF matrix using method m as listed at the end of section 4.2.3. Cell line datasets containing mRNA $r_n^{(c)}$ and protein abundance $p_n^{(c)}$ belong to cell line datasets $c \in \mathcal{C}$. We use $\mathbf{X}$ and $\mathbf{\Phi}$ interchangeably if the reduction is irrelevant to the narrative. We may also use r^2 and r^2_{adj} interchangeably, assume the latter is always being used. In certain subsections and supplementary we use root-mean-squared-error (RMSE) as the scoring metric rather than r^2, these can be viewed in a similar light, except RMSE is minimized instead of maximized.

5.2 Results

In this section we will consider the impact of steady-state and dynamic protein abundance datasets as a useful target for SDF prediction. We will begin with model selection, considering the previous dimensionality reduction datasets we generated in the previous chapter.

5.2.1 Model Selection of SDFs Against Expression Level Emphasises Feature Adaptability

In this subsection we will cover the detailed analysis in selecting appropriate models for determining protein abundance across a number of selected cell lines. These cell line studies as introduced in section 5.1 were selected based on the following three factors:

1. Study in question contained both mRNA and protein abundances.

2. Sufficiently large sample size $N > 10^3$ for statistical power.

3. Both steady-state (Lundberg et al., Vogel et al.) and time-dynamic (Aviner et al.) experimental set-ups for expression data.

Firstly, we analyse which regression technique performed best across most of the cell lines, secondly we compare by cell line and unsupervised learning preprocesing technique to generate $\mathbf{\Phi}$. Finally, we choose best $\mathbf{\Phi}^{(m)}$ and re-fit using the best regression model to determine coefficient importance.

Fitting by regression model Given normalized or reduced SDF data matrix $\mathbf{\Phi}^{(m)}$, we first associate the target data $\mathbf{y}$ using shared labels provided by Biomart/HGNC, using the set intersection operation. Then following basic filtering of low variance and missing values, we split the data into training/testing and validation subsets in a 4-to-1 ratio. We then designed a parameter grid space of varying linear and non-linear regression models, such as MARS (Earth) [103], Random Forest, Gradient-boosted regression in the form of XGBoost [112], SVM, ElasticNet [108] and others. We also consider an ensemble approach in the form of a Voting regression model; composed of Ridge ℓ_2, ElasticNet and SVM with uniform weighting. This is performed across each *H. sapiens* cell line, using the training subset. The aforementioned transformations such as z-score mean that differences in standards between the various mRNA/protein technologies can be mostly ignored. For details on the exact process that went into model selection, see Figure 5.1 and section 6.2. Selecting the best parameters for the best model allows to predict samples using the validation subset, yielding a validation error on data never seen by the model in any previous training step. For each regression model we select the one which maximises the adjusted r^2 (accounting for $P \gg N$), and develop models that a) target protein abundance, b) target mRNA abundance and c) target protein abundance, including mRNA level in $\mathbf{X}$ input (see Figure 5.2). Gradient-boosting with histogram-binning performed best on average ($r^2 = 0.5$) against protein and protein with mRNA models. SVM, XGBoost (Gradient-boosting) and Elastic Net [114] all following closely within the margin of error. Most algorithms fall within the $r^2 =$[0.45, 5] range.

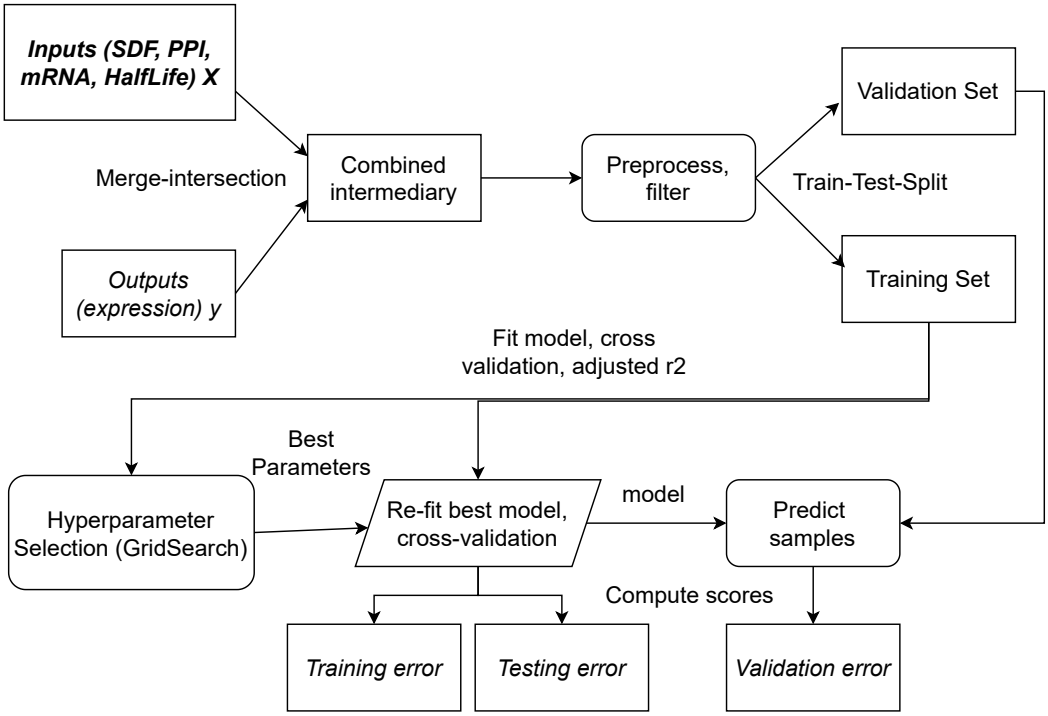


Figure 5.1: Flow diagram of model selection. *Flowchart describing the process of converting SDF inputs and expression data into outputs in the form of predictions and model scores. For clarification, the 'validation set' and 'test set' are interchanged terms within the literature. Here we refer to the 'validation set' is the out-of-sample set.*

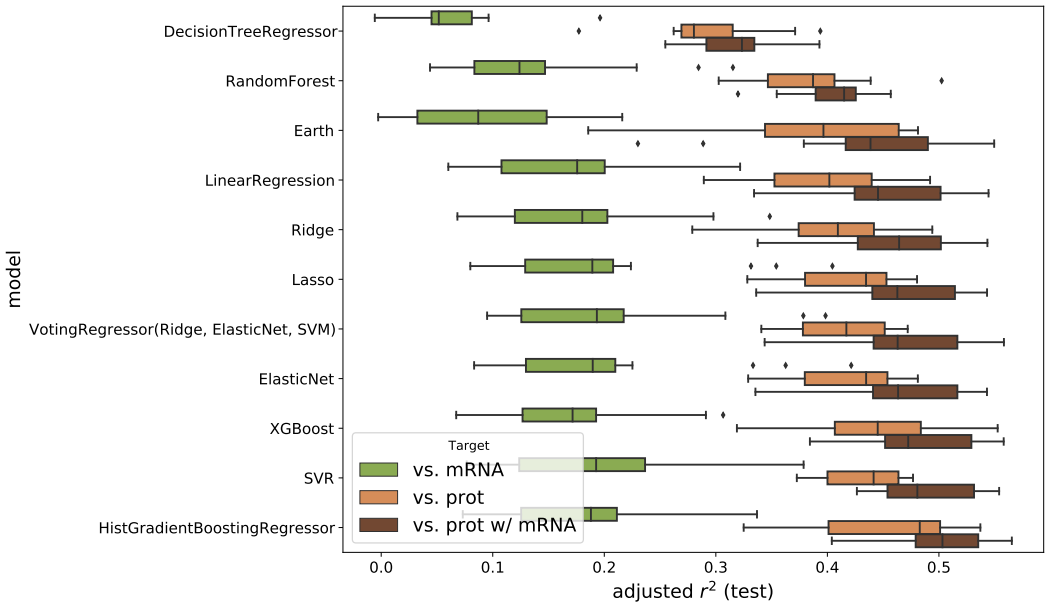


Figure 5.2: SDF Model Selection in aggregate using maximum score hyperparameters. *Calculated adjusted r^2 (eqn 2.50) against different regression models, with SDF using mRNA as target* $\mathbf{y}$ *(green), protein (orange) or protein with mRNA included in* $\mathbf{X}$ *(brown). Errors are across different* $\mathbf{\Phi}^{(m)}$*, 5-fold cross validation, and cell lines* $\mathcal{C}$*. For the avoidance of doubt, these are not out-of-sample scores.*

Fitting by unsupervised learning technique and cell line What is most striking is the broad similarity in the performance of most models with the exception of Decision trees, which are too simplistic to cope with this many features. However, the errors with regards to OLS and MARS indicate that different transformations of $\mathbf{\Phi}^{(m)}$ play a more important role with training these particular models; Elastic Net and SVM on the other hand have significantly lower variance across these domains. Whilst these scores may seem rather unimpressive, we are not selecting the best matrix input $\mathbf{\Phi}$, cell line and so on; merely plotting the average distribution over these variables. To filter out such noise, we also compare against differing reductions of $\mathbf{X}$ as explored in the previous chapter and against the differing cell lines $\mathcal{C}$ (see Figure 5.3). For choices of input model m, a stratified approach (sPCA) performs best across the various target variables, with FA in this case being too selective and reducing K too much. The non-reduced data also performs very well on protein prediction, but not so for mRNA prediction. It is clear that dimensionality reduction does not improve model performance, as is common in other use cases such as image processing; with most methods slightly reducing r^2_{adj}. With regards to cell line $\mathcal{C}$, we consistently see SDF-based models outperform mRNA-protein models (red diamonds) across all cell lines that we explored during the model selection process. SDF-alone models that predict global protein abundance typically score around $r^2 \sim 0.44$; roughly twice the score of global mRNA-protein relationships. Note that this score is substantially lower than many previous studies which have either focused on specific subsets of genes that have high mRNA-protein correlation (ribosomes) or in more primitive species. The addition of mRNA into the input model increases r^2 by around 5%. A slightly confusing aspect is the fact that HeLa-based cell line models outperform non-HeLa despite the non-steady-state nature of the study and its subsequent proteins.

Feature Importance via refitting Given the success of many potential models in terms of r^2 score, we opted to choose a model that also trained in a relatively quick time; we turned to the XGBoost package [112] which has great performance and GPU utilisation. For training, we chose the reduced subset $\mathbf{\Phi}$ using stratified PCA for continuous and MCA for discrete amino acid meta features (sPCA—MCA), using protein abundance as the target variable. To explore which of the features are most important, we took the best models by cell line and looked at the relative importance as weighted

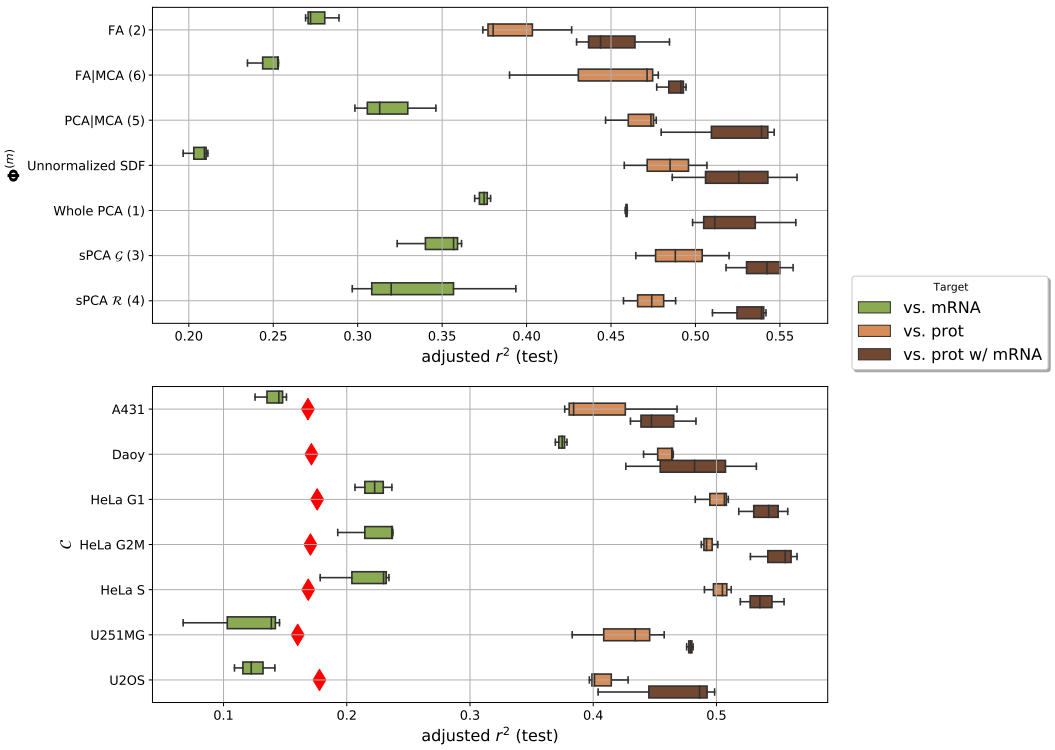


Figure 5.3: SDF model selection against X input and cell line. *Boxplots of 5-fold cross validated adjusted r^2 scores for various regression models against top) reduced $\mathbf{\Phi}$ matrix and bottom) cell lines $\mathcal{C}$. Red diamonds indicate OLS models of $\mathbf{r}_c$ mRNA against $\mathbf{p}_c$ protein abundance for each cell line dataset.*

by the tree-models (see Figure 5.4). The top 2 axes describe the top 10 performing feature importances as determined by data groups. Aggregated PTM features contribute over 30% of feature importance, for models without corresponding mRNA abundance (left) and with mRNA (right). The mRNA addition to score is just under 10% when included as an input vector to $\mathbf{\Phi}$.

It is important to recognise that many data groups have at least one principle component within the top 10 of important features, recognising the diverse contribution of many data sources. To see how each data group $\mathcal{G}$ performs collectively, we sum importances by data group (see bottom two axes), which reveals that apart from dinucleotide frequency (an artefact of having 64 features), the most important features in both w/out and and w/ mRNA abundance contain PTM, amino acid frequency and AA-text features that cumulatively represent nearly 50% of feature importance. When mRNA abundance is included as input, it becomes the 5th most important feature at 5-12%, cell line $\mathcal{C}$ dependent. We provide further breakdowns within Supplementary Figures (see Appendix S28B), whereby a dominance of post-translational modification (PTM) features and length associated with the amino acid sequence appear dominant for all but the Daoy cell line. Both Acetylation and Ubiquitination are associated with protein stability, where Acetylation also deals with protein localization and synthesis, whereas Ubiquitination is also associated with cell cycle division and immune response reaction. Sequence length is consistent with previous studies [165] but relationships between expression and PTMs have seen less interest in the literature. The importance of PTM features is also reflected within partial correlative analysis $r(\mathbf{X}_p, \mathbf{p}_c|\mathbf{m}_c)$ of each SDF feature to protein abundance, fixing for mRNA abundance or length (see section 6.2).

Synthetically-generated SDF benchmark Here we explore synthetic generation of a data matrix to compare to mRNA and protein expression data. we generate these synthetic datasets as follows:

- Sample mRNA or protein abundance from a fitted normal distribution $\mathbf{y}_c \sim \mathcal{N}(\mu_y, \sigma_y^2)$, where μ_y and σ_y^2 are determined by the sufficient statistics of $r_n^{(c)}$ or $p_n^{(c)}$.

- Sample SDF data matrix using singular value decomposition (SVD) in the form $\mathbf{M} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^{\mathrm{T}}$ where $\mathbf{M}$ is a $P \times P$ matrix, then computing

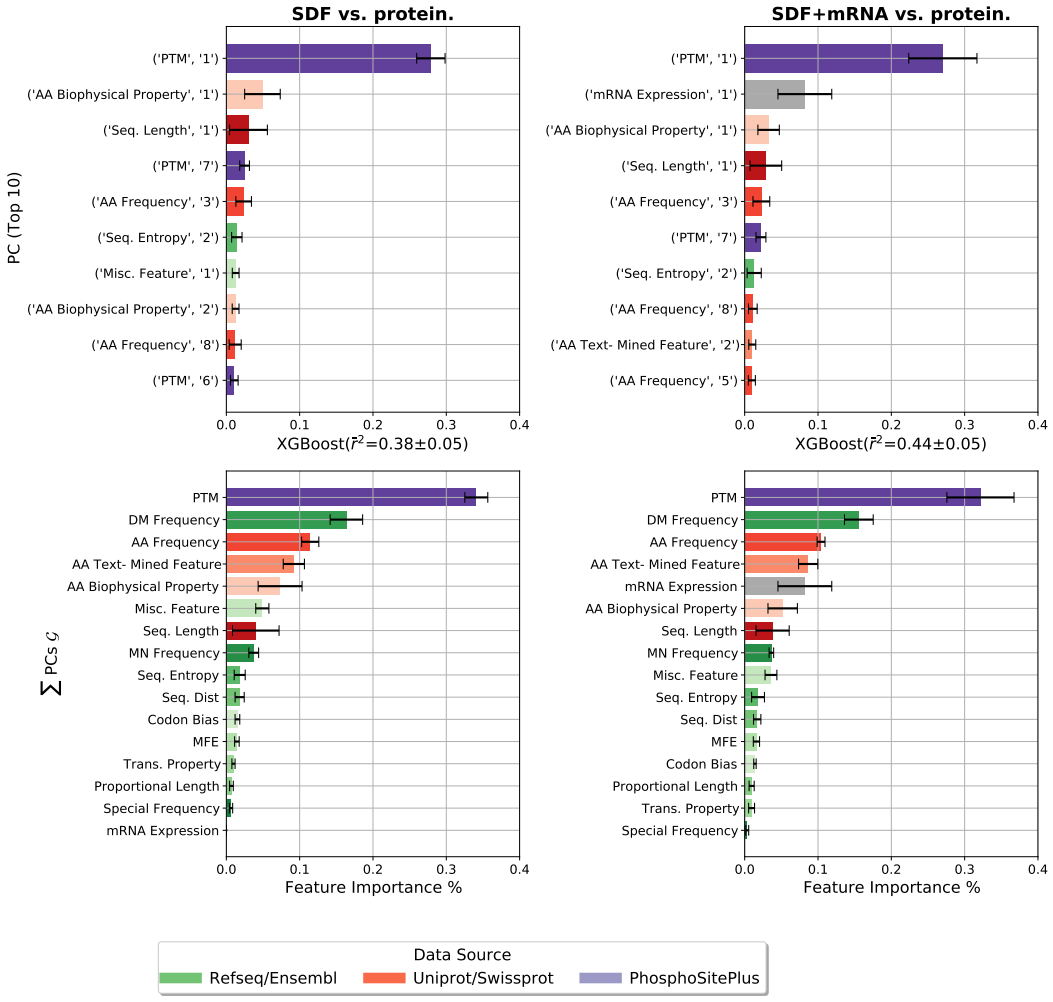


Figure 5.4: Feature importances from XGBoost model for SDF-fitted regression models. *Top) Top 10 normalized feature importance scores for principle components from* $\mathbf{\Phi}$ *using sPCA—MCA. Bottom) Summed feature importances* $\sum_{\mathcal{G}}$ *over each data group. Left) SDF features as input with target protein, right) same as left but with mRNA expression included. Error bars* $\pm$*SD contribute to variation in cell lines* $\mathcal{C}$ *for differing models. Features are coloured according to data group (with data source in legend, mRNA features green, amino acid orange/red, PTM purple).*

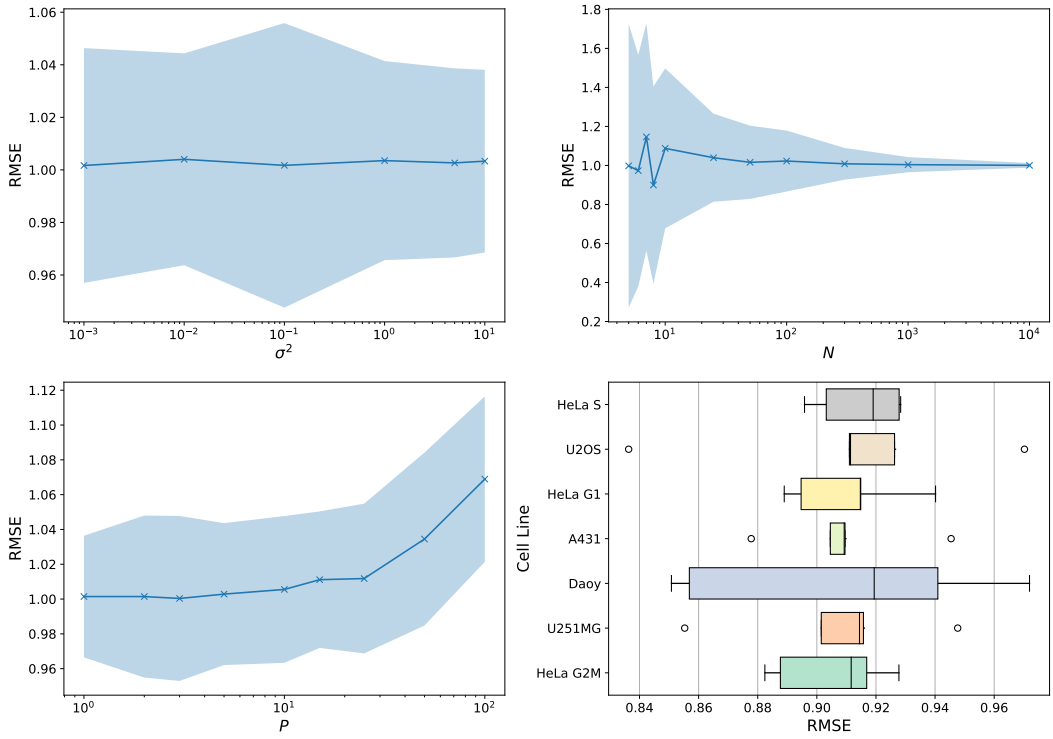


Figure 5.5: Synthetic dataset analysis with respect to SDFs. *Top left) RMSE for differing values of σ^2 when creating* $\mathbf{X}$. *Top right) RMSE against N samples of generated dataset. Bottom left) RMSE against dimensionality P of generated dataset. Bottom right) OLS models of mRNA against protein across different cell lines as baseline comparison.*

$\mathbf{X} = \mathbf{A}\mathbf{U}^{\mathrm{T}} + \sigma^2\mathbf{A}$ where $\mathbf{A}$ is a $N \times P$ matrix, and finally incorporating a homoscedastic noise parameter in the form $\sigma^2\mathbf{A}$. Here $\mathbf{A}$ and $\mathbf{M}$ are both sampled from standard normal distributions $\mathcal{N}(0, 1)$ with the same seed. By default $\sigma^2 = 1$.

- Use the same grid search and compute RMSE using 5-fold cross validation for each c cell line.

Using this formulation, we analyse the impact of changes in N, P and σ^2 on modelling distributions of $\mathbf{y}$ using Gaussian noise (see Figure 5.5). This is useful because it provides a baseline error by which RMSE values less than this baseline indicate improvement over random chance. As we can see, synthetic models when trained to distributions of c converge to an RMSE around 3.2 without z-score transformation and 1 (convergence with σ^2) with z-score. We note that mRNA-protein models only have RMSE around 0.91 0.93, emphasising the weak contribution that mRNA level provides in protein prediction. Similar results were observed by perturbation of training examples $\mathbf{\Phi}$, holding $\mathbf{y}$ fixed and following the model selection pipeline as described previously.

5.2.2 Sequence-Derived Features Aid Prediction By Capturing Information Regarding The Translation Process

Whilst we have begun to show that SDFs as a whole work across the human proteome, it remained unclear which functions and biological processes were benefiting the most from these features as input, and which processes were not being covered by this approach. Thus, using the same models (with hyper-parameters fitted), instead of using the entire expression dataset, we re-trained models whereby the out-of-sample test set consisted of all of the genes which shared a particular Gene Ontology (GO) term, and the training set consisted of every other gene not in this group. We filtered for GO terms which had at least $N \geq 50$ proteins associated with the term, to allow for statistical robustness. This process was repeated for models with different design matrices; containing just mRNA level, just SDF features, or both as input (see Figure 5.6A). Next, we selected the 10 GO term models which have the lowest (5.6B), highest (5.6C) and the most-improved (5.6D) average RMSE score across all cell lines. The lowest RMSE models, similarly

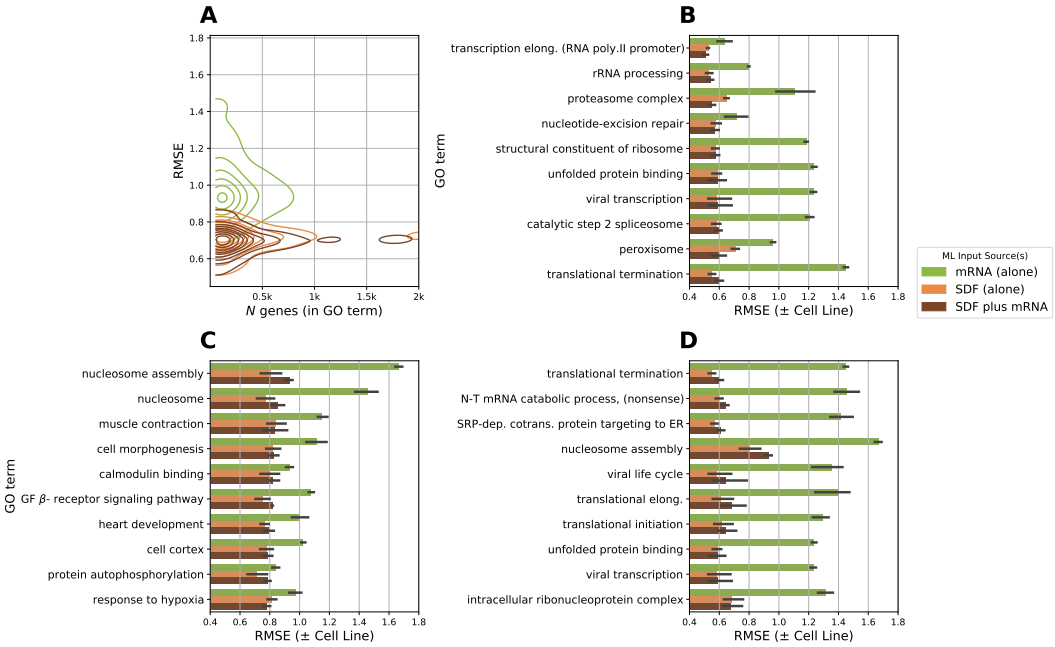


Figure 5.6: Representation of translation-oriented GO terms predominant in best SDF-based models *Out-of-sample root mean squared error (RMSE) scores of Gradient-boosted regression trees (GBRT) by Gene Ontology (GO) terms. (A) Bivariate kernel density estimation of the number of genes by RMSE for each GO term model by data input source. Barplots of the 10 (B) lowest, (C) highest and (D) most improved (difference between SDF plus mRNA and mRNA alone) RMSE-scoring GO term models by data input sources. Within GO; Biological Process, Cellular Component and Molecular Function terms are included. See Methods and Supplementary Material for details on model and feature preprocessing/selection.*

to the global proteome model, reflect a 25-30% improvement in abundance prediction compared to the worst term models. In the vast majority of cases, significant improvement in prediction is achieved with the introduction of static SDFs into the input design matrix. Noticeably, the lowest and most-improved term sets have a good coverage of translation-situated terms, such as translation elongation and initiation, as would be expected by heavy usage of codon bias and frequency-based features, and such relationships have been observed in previous studies [6, 3, 93].

Similarly, the selection of ribosome-oriented terms is expected, given the high correlation between mRNA and protein levels between ribosome-associated genes, which impacts on model performance. More surprising is the selection of protein localization (such as SRP-dependent co-translational protein targeting to ER), and mRNA/protein decay by translational termination and mRNA catabolic process terms. We did not include mRNA or protein half-life features by direct measurement as a part of the input features, so it is interesting that there are aspects of the SDFs that can predict these functions. Genes that under-perform are associated with labile proteins that perform functions such as development (heart development, cell morphogenesis) and/or complex signalling pathways such as response to hypoxia or cell-line specific functionalities not covered in the cell lines we modelled on, such as heart or muscle tissue (e.g muscle contraction).

5.2.3 Protein Interaction Networks Complement SDF Coverage In Predicting Abundance

At this stage the feasible limits of SDF benefit had been reached, and hence the idea of incorporating protein-protein interaction (PPI) network information seemed intuitive and appealing. Given each protein p_n, $n = 1, \ldots, N$, we assume protein p_n has a set of J_n neighbours $(p_1, \ldots, p_{J_n})$. Let p_i and p_j be two potentially interacting proteins. Determining whether proteins p_i and p_j interact is given by an interaction confidence score $s_{ij} \in [0, 1]$ as defined by STRING. Given that s_{ij} is a combination of sub-scores (evidence collected by data-mining, experimentation, homology etc), we can define PPI feature matrix on a per-protein basis as $\mathbf{M} \in \mathbb{R}^{N \times K}$, where K is the number of sub-scores. Each feature $\mathbf{m}_j$ is the mean sub-score across all interactions for each n. Using these features, we compute the intra-correlation of $\mathbf{M}$ with the

HeLa expression dataset [5] and plot this using a Hinton diagram (see Appendix S27). Protein node degree, eigenvector centrality and co-expression sub-scores provide the largest positive correlation r_p to mean mRNA and protein levels; this is intuitive as we may reasonably expect abundance levels to correlate with more associations to proteins. Another interesting aspect of this feature set are the *transferred* features, which represent the evolutionary impact of related species with a similar protein interaction, not accounted for in our original SDFs. We do not however see significant correlations to expression level within these features.

Next, we incorporate the PPI feature matrix into our modelling strategy and compare its performance to our SDFs, in conjunction with mRNA abundance and mRNA and protein half-life data taken from Tani et al.[169] and Cambridge et al.[170] respectively. Here we compare the following cross-section of inputs:

- mRNA expression, mRNA with half-life (HL), mRNA with SDF.
- Protein-protein interaction (PPI): PPI with mRNA, PPI with HL
- SDFs: SDF with HL, SDF with PPI, SDF with mRNA and PPI, SDF with HL and PPI

The results of this process are shown in Figure 5.7, whereby we will break down the subsequent analysis by input type, excluding mRNA abundance as we have already exhaustively covered this in the first chapter (see Supplementary 6.2 for further details).

PPI Protein-protein interaction feature matrix $\mathbf{M}$ was derived ($P = 15$) as previous described, where features provide an average adjusted $r^2 = 0.24 \pm 0.01$ for Lundberg et al. [164] features, whereas HeLa data from Aviner performs better at $r^2 = 0.33 \pm 0.01$. This was somewhat surprising as we expected an average interaction metric to conform with steady-state cell line datasets better than the dynamic cell line. PPI alone performs best with Daoy data from Vogel's group [6], but due to the comparatively small sample size $N = 10^3$, this may be due to selection bias. This is particularly prevalent within SDF modelling of this line as further database integrations with SDF-PPI and SDF-mRNA significantly reduce sample size, generating large standard deviations across 5-fold cross-validation scores. To combat

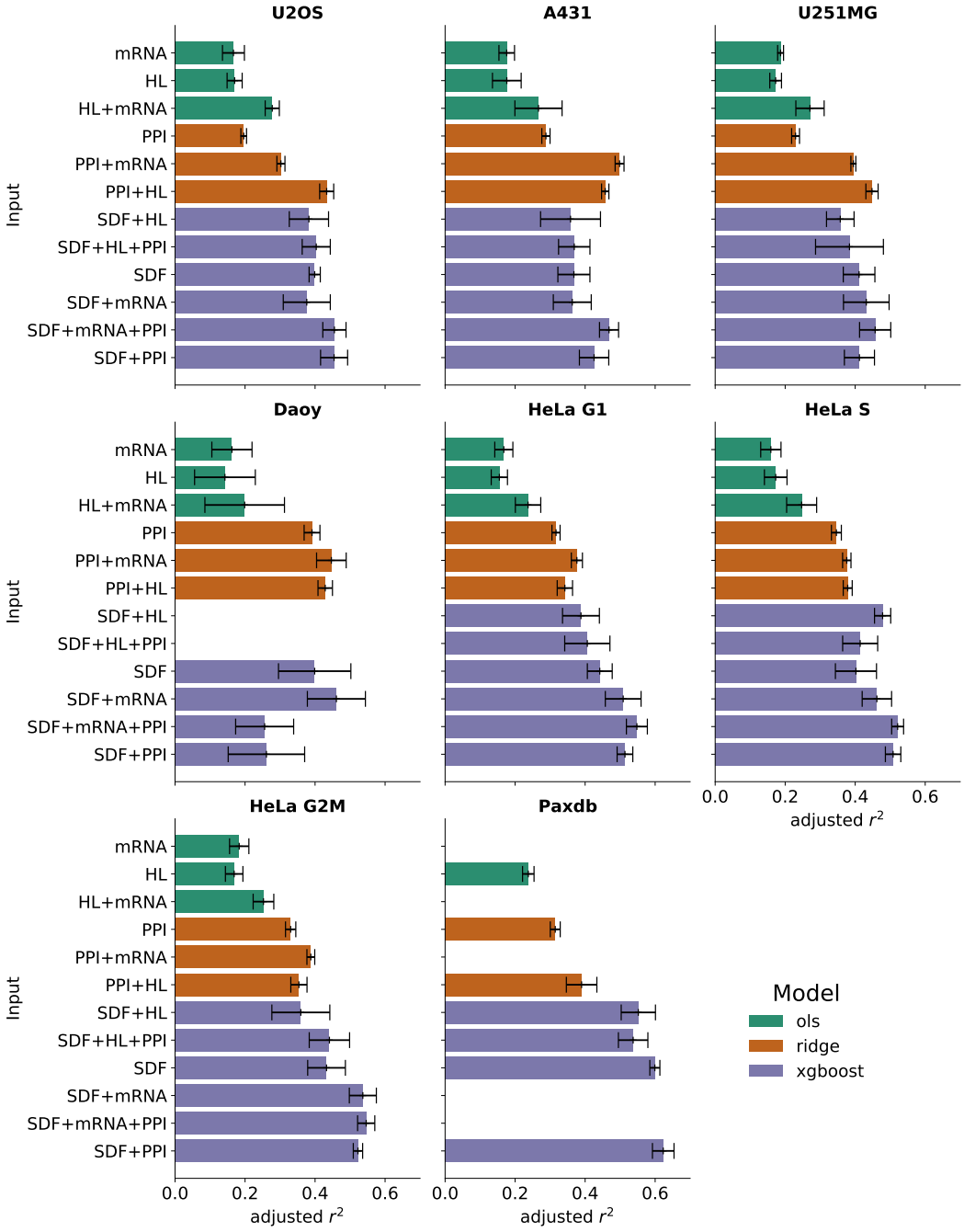


Figure 5.7: Comparison of half-life (HL), PPI, mRNA and SDF models against protein abundance. *Barplots of adjusted r^2 test score $\pm$SD 5-fold cv with fixed repeats. Missing values indicate too few samples n for X_{train}. Axes are split by cell line for expression data.*

this, we performed repeated k-fold cross validation (that is cross validation with r repeats) to mitigate splitting biases.

Half-life Steady state half-life measurements (both mRNA and protein) perform similarly to mRNA expression ($r^2 \sim 0.2$), once again performing best with Paxdb-processed datasets. When combined together, performance noticeably increases by 10% across most cell lines. The most interesting result was half-life in combination with protein interaction features - this lead to a marked improvement in r^2, and in many cases achieved a mean score better than SDF-input models (albeit usually within the margin of error). The only exceptions to this are in the HeLa cell line and PAXDB conglomerate; where SDF-based models of all stripes score better across the board than PPI-based models. HL plus SDF tended to actually lower the score on average compared to just SDF (due to adjusted r^2 punishing a larger dimensionality); indicating redundancy in HL inclusion. It would be reasonable to assume therefore that steady-state HL measurements are being nearly fully factored within our SDF feature set.

SDF Here we use the sPCA—MCA preprocessed feature set ($P \geq 135$), whereby SDFs alone perform fairly well (4th place on average), and inclusion of HL negatively impacts adjusted r^2 performance (see half-life paragraph). The best models tend to include PPI and/or measured mRNA abundance, as these features are likely to include information not obtainable within the fixed sequence data. Indeed the Paxdb dataset performs best with most SDF-based models scoring near to $r^2 = 0.6$ or explaining 60% of model variance. This is likely due to the averaging effect of taking abundances from multiple human cell lines into the combined metric provided by the database; as we have previously suggested, SDFs tend to perform better on steady-state protein abundance prediction. From our first paper Parkes et al. [1], we deployed translation rate abundance from HeLa cells as an input feature in conjunction with mRNA and a basic SDF feature set; this yielded $r^2 = 0.66$, indicating that at least another 6% variance can be provided for by concentrations surrounding the translation process. Hence from all models deployed, there remains approximately one-thirds of the error unexplained by the features; a few percentage points of which can be reasonably attributed to experimental noise (in triplicative measurements, for instance) and averaging

strategies.

5.3 Discussion

SDF Model Selection Gains New Insights on Protein Abundance Impact

We demonstrate that a rich SDF set easily outperforms mRNA abundance alone in predictive power, with mRNA input improving accuracy by around 10% on average. To compare whether these fitted models were an improvement on previous research, we re-used the SDFs extracted by Vogel et al. [6]; fitting them against the same expression levels (see Appendix S28A, including 5-fold cross validation). In all cases these updated models outperform previous work and/or substantially reduce the error variance. Notably, there is a high degree of similarity between the cell line data sources (Aviner et al. [5], for HeLa, etc). This could be for a number of reasons including 1) technique of mRNA/protein measurement, 2) normalization methods, 3) steady-state vs. dynamic nature of expression or any combination of these. The lack of root-mean squared error (RMSE) reduction in the HeLa lines may be due to the dynamic nature of cell cycle activities, whereby SDF features struggle to predict non-steady state protein expression levels. Further to this, our SDF features have been calculated across the entire transcriptome, and are able to capture a much higher percentage of the proteome than previous feature sets (see Appendix S28B); with the exception of the relatively small Daoy expression set, and assuming the hypothesis of 'one gene to one protein', we cover roughly 20-25% of the human proteome (around 4-5k proteins) in these example datasets. Increased coverage could be achieved at the transcript variant level by avoiding the averaging effects and significantly increasing the data size at the cost of computational resources.

Analysis of Input Sources for Global Protein Abundance

In this chapter we have expanded to use a vast array of biological database sources, including PAXDB [168], protein-protein interactions (STRING) [171] and protein half-life [170]. Protein interaction networks intuitively provided data not available at the sequence level, however exploiting this within a

machine learning context proved to be somewhat difficult, due to the lack of connectivity (around 0.3% of total possible connections). This therefore meant that the vast majority of proteins have only one connection to another protein, and particularly once integration between label sets was achieved. This lead to significant falloff in sample size N as the number of neighbours $K \to \infty$ increased. For each trained model that includes mRNA abundance, we always use the values derived from that cell line, but there remains interest as to the cross-talk by using expression values from different cell lines, as one would expect congruent expression across the majority of mRNAs. This may have aided in improving predicted scores, particularly in cell lines such as Daoy where limited data was present. We also did not utilize non-coding RNAs within this Thesis, as one of the early steps involved selecting for coding RNAs with at least one variant with a coding sequence. This could have played a role in reducing the utility of protein interaction network data within our downstream analysis.

Chapter 6

Conclusions and Future Work

Here we will discuss the primary conclusions drawn from the work performed during this dissertation, and discuss future directions that may be taken in response to conclusions.

6.1 Conclusions

This thesis is based fundamentally upon data-driven modelling applied to the analysis of high-throughput measurements of protein abundance. We focus on the relationship between transcriptome and proteome, whereby mRNA abundance has historically been taken as proxy for protein abundance. Many previous authors have looked for correlations (section 2.4.1) between transcriptome-proteome [126, 127, 128, 129, 3] both locally and globally. It is noted that this correlation is unexpectedly weak; particularly for higher-order organisms such as *H. sapiens*. Tuller [137, 53] and later Gunawardana [3, 93] moved beyond correlations to modelling abundance using a constructed predictor in *S. cerevisiae*, via regression. We also focus on the development of a rich sequence-derived feature (SDF) dataset, as inspired by Vogel [6] who found that such features could explain two-thirds of protein variance within the Daoy cell line.

In Chapter 3 we demonstrate the ability to predict protein abundance levels within the human HeLa cell line using a variety of linear and non-linear modelling techniques. This goes further than Gunawardana's [3] work who works with yeast cells, and Aviner's [5] work who does not develop

a protein predictor and instead looks at fold-change differences across the cell cycle. The novel introduction of translation rate measurements via the PUNCH-P technique [4], allowed for improved linear predictors of protein abundance ($r_s = 0.67$), and we found that translation largely superseded mRNA abundance as an important predictor. Model comparison using frequentist (AIC) and Bayesian approaches (section 3.2.1) demonstrated that having both mRNA and translation lead to negligible improvements in model performance, against just using translation. We then used some 30 sequence-derived features such as codon bias, tRNA adaptation index and other metrics as inspired by Gunawardana [93] and Vogel [6] to improve the accuracy of our linear predictors. We show that such features can improve performance, but only in an ensemble approach. Our attempts to reduce dimensionality to the most important features were met with no tangible benefits in model performance, leading to the conclusion that a large number of features contributed a small but significant step in modelling the protein abundance domain (section 3.2.3). This somewhat diverges from Gunawardana, whose yeast-based models yielded very high $r^2 = 0.86$ with less than 10 features. Following from Gunawardana's thesis, we hypothesized overestimated outliers to our fitted models (which included mRNA and translation) to be closely associated with post-translational regulation pathways (sections 3.2.3,3.2.4), specifically favouring the utilisation of ubiquitin-like degradation, which we confirmed by coarse-grain bootstrapping PTM samples and with gene ontology analysis.

In Chapter 4 we expanded on the concept of using sequence-based information as predictors, specifically as a possible replacement proxy for mRNA abundance in studies where translation is not measured. Firstly, we performed feature engineering across the entire human transcriptome for a variety of feature types, such as genome base profiling, biophysical properties, free folding energy and more (section 4.2.1). We then undergo extensive analysis to ascertain the usefulness of each engineered feature group via inter-correlation analysis; as a critical assumption of statistical modelling assumes independence between predictors. We compared both linear and non-linear methodologies of dependency to take into account idiosyncrasies with regards to data transformation and/or scaling. We utilised Vogel's work [6] as a benchmark comparison to our own expanded SDF set, by comparing the raw sequence data via sequence alignment and derived-features via correlation. To overcome inevitable problems with multicollinearity, we ex-

plored an array of feature selection options (section 4.2.3) including PCA and manifold techniques. Recognising the need to balance performance and interpretability, we adopted a compromise solution of stratified PCA (sPCA) which sacrifices some interpretability by using PCA, but retaining decent compression ($K = 135$) and the groups from which SDFs were engineered.

In Chapter 5 we utilised the developed SDF feature matrix as developed in Chapter 4 back to the problem of global protein abundance prediction, across a variety of steady-state and dynamic human cell lines. Firstly, we performed extensive model selection over several PCA-reduced variants and machine learning models, which illustrated that SDF-input trained models consistently provide double r^2 compared to mRNA-input trained models even adjusting for higher dimensionality (section 5.2.1). We found that including mRNA as an input contributed around 5% of the explainable variance, with PTMs providing the bulk of feature importance (over 30%) for fitted models. Refitted models based on gene subsets as determined by Gene Ontology indicated that SDF models tended to represent information regarding the translation process (section 5.2.2), as was evident from terms which provided the highest model accuracies. Likewise, GO terms that yielded the lowest scores tended to be associated with tissue-specific functionality (i.e heart development) and various cell signalling pathways which are inherently uncertain. Finally, we extracted protein interaction data and half-life measurements to perform a system-wide analysis of varying genetic data input sources (section 5.2.3). We found that PPI features contribute an additional 10% to explaining model variance, whereas half-life measurements lead to no increase in model performance; often negatively impactful due to the integration cost of reduced samples. With best models performing around $r^2 = 0.6$, and $r^2 = 0.67$ in HeLa with translation included, we have a remaining deficit of one-thirds model variance unexplained by the features we have sampled; we estimate that a significant portion of this is the true noise, in conjunction with a myriad of small factors, such as experimental noise, accumulated averaging errors, computational errors and lack of spatio-temporal data.

6.2 Future Work

In this thesis we delve deep into transcriptome-proteome analysis, and protein prediction via static sequence information. Here we explore a number of

potential avenues for future work to progress towards. At the outset, we will mention the utilisation of single-cell transcriptomics and proteomics that have only recently become popular and will factor to some extent within all of the below arguments for future work.

Completing the cell cycle A major limiting factor to proteomics is the lack of resolution that is currently provided by expression data, and in particular bulk sequencing. A more complete understanding of protein concentration can also only be undertaken within a spatio-temporal context. Much of the work in this Thesis focuses on the cell cycle dataset provided by Aviner et al. [5], which only provides 3 time-steps (at 2, 8 and 12/14 hours, with δt =6h), representing a small snapshot of the transcriptome, translatome and proteome for each cell cycle phase. A large-scale study should be commissioned along these lines, but increase the time granularity significantly such that δt is closer to 10 or 15 minutes, across at least one complete cycle. Bulk or single-cell sequencing could be deployed in parallel, with the cell population synchronized using a double thymidine block to early S phase. Replicative measurements could then be taken for mRNA abundance by single-cell RNA-Seq, and translation/protein abundance through single-cell LC/GS-MS proteomics. Measuring replicates will be especially important to minimize single-cell and instrumental noise, and to aid with minor cellular de-synchronizations that will occur throughout experimentation. Conducting such a study would be incredibly time-consuming and expensive, but would yield a global map of expression within a cell line across time. As the cell cycle is the core process that initiates most regulatory pathways (excluding external signalling), this could help to a) reveal more labile regulatory proteins/RNA, b) provide proof-of-concept to mathematical models of cell cycle expression and c) enable accurate predictions of most proteins involved in internal-signalling pathways.

Identifying biomarkers for disease The SDF set can be applied to a number of specific biological problems, including the identification of new candidates for various cancers and diseases. A natural follow up would be the association of this global sequence-derived data with custom single-cell transcriptomics/proteomics data and other database sources applied to the given cell line(s) of interest. A number of modelling techniques could then be deployed to either directly classify potential biomarkers, or indirectly via out-

lier detection. These models could then filter through thousands of biomarker candidates to highlight the most likely to be sent for experimental validation. Sensible choices for feature selection and/or dimensionality reduction will remove irrelevant features and naturally fit to the associated expression data, providing a custom SDF-reduced set for the particular problem.

Sequence Information Utility Our SDF set is designed such that it could be incorporated into a number of future analyses at multiple levels of the gene hierarchy, such as transcriptomics, proteomics and/or metabolomics. There are a number of ways we could have expanded the remit of sequence information which we engineered, such as:

- *Breadth-wise*: Whilst our input feature space ($P > 200$) is relatively large, we could expand it by considering miRNA and other non-coding RNA information at the transcriptomic level, particularly if studies did not require any proteomic analysis. Indeed Vogel [6] and others included various miRNA features in their SDFs, but found most of them correlated very weakly $r_p < 0.05$ with protein abundance.

- *Depth-wise*: Instead of engineering features on a per-protein basis, we could have instead worked on a per-transcript variant basis, in order to encapsulate sequence variations that many genes have. This would have provided benefits in an increased sample size N, which could have possibly lead to requiring deep neural network-like applications, leading to higher accuracy. This was however practically infeasible due to the computational cost involved in performance and memory, and provides little in consistent interpretability needed to discover the underpinning mechanisms surrounding the proteome and its regulatory pathways.

Unrelated to the sequence, but still potential applicants as suitable predictors are various measured quantities such as estimates of ribosome binding (Ribo-seq), average global DNA methylation profiles and chromatin accessibility.

Recent work on protein abundance prediction by Deepmind [178] yielded model accuracies of $r^2 = 0.86$ using a complex deep learning (DL) architecture over a 100 kilobase range for each gene, with $r^2 = 0.93$ being the maximum r^2 obtainable given current experimental standards of measuring

relative abundance. In light of this, whilst there remains a space for feature engineering in research and conceptualising the problem domain, gains in model accuracy provided by DL will likely overshadow our approach, as it has done in other fields with historically dominant use of engineered features such as computer vision and natural language processing.

Alternative Modelling Strategies We exclusively limit our analysis to global protein abundance and modelling through a continuous target variable (regression), but discrete target variables could also be utilised for prediction; such as protein function or interaction. We also primarily used the identification of post-translational regulation/degradation as a research field of interest, via outlier analysis; but the general principles of finding interesting outliers through input selection are applicable to many research questions. In particular, we did not fully utilize the information provided by a protein interaction network (we primarily adopted a tabular-based approach to analysis); future approaches could move towards graph-based analysis, using SDF information to characterise weight edges between interacting proteins, for instance. Other approaches with merit are probabilistic and/or generative modelling; able to more easily handle missing values and accurately estimate noise. A full Bayesian treatment with parameter uncertainty quantified could be considered for a variety of future approaches, such as predicting whether two proteins interact, whether protein A has function $f(A)$ and so on. Finally, a significantly larger input domain could warrant a deep learning/RL approach, as those adopted by others [178]. Our systemic global proteome approach could also serve as a benchmark for a myriad of specific future projects that look at interesting subsets of proteins, whether to discover new theory, interactions, biomarkers or therapeutic interventions.

Supplementary Material

Derivation of Ordinary Least Squares

Given the description of a *Linear Regression Model* (2.7), we begin with the problem using the sum-of-squares error function defined as:

$$\mathcal{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left(\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) - y_n\right)^2 \tag{6.1}$$

Using this error function, we incorporate this into the log-likelihood function which is an adaptation from (2.9):

$$\ln p(\mathbf{y}|\mathbf{w},\lambda) = \sum_{n=1}^{N}\mathcal{N}\left(y_n|\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n),\lambda^{-1}\right) \tag{6.2}$$

$$= \frac{N}{2}\ln\lambda - \frac{N}{2}\ln(2\pi) - \lambda\mathcal{E}(\mathbf{w}) \tag{6.3}$$

where λ is the precision, or inverse variance. Note that we drop $\mathbf{x}$ from the conditional distribution parameters as this is assumed to reduce notational clutter. To solve we maximise the log-likelihood (which is equivalent to minimizing the sum-of-squares error) by computing the gradient of the log-likelihood:

$$\nabla \ln p(\mathbf{y}|\mathbf{w},\lambda) = \lambda\sum_{n=1}^{N}\left(y_n - \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n)\right)\boldsymbol{\phi}(\mathbf{x}_n)^{\mathrm{T}} \tag{6.4}$$

And thus setting the gradient to zero gives:

$$\sum_{n=1}^{N} y_n\phi(\mathbf{x}_n)^{\mathrm{T}} - \mathbf{w}^{\mathrm{T}}\left(\sum_{n=1}^{N}\phi(\mathbf{x}_n)\phi(\mathbf{x}_n)^{\mathrm{T}}\right) = 0 \tag{6.5}$$

Solving for $\mathbf{w}$ we have [173]:

$$\mathbf{w} = (\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi})^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{y} \tag{6.6}$$

where $\mathbf{\Phi}$ is the *design matrix* where elements are given as $\Phi_{np} = \phi_p(\mathbf{x}_n)$. The quantity:

$$\mathbf{\Phi}^{\dagger} \equiv (\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi})^{-1}\mathbf{\Phi}^{\mathrm{T}} \tag{6.7}$$

is known as the Moore-Penrose pseudo-inverse of the matrix [174]. The projection matrix $\mathbf{P}$ which corresponds to an orthogonal projection of $\mathbf{y}$ onto the column-space of $\mathbf{\Phi}$ is given as:

$$\mathbf{P} = \mathbf{\Phi}\mathbf{\Phi}^{\dagger} \tag{6.8}$$

and is also known as the hat matrix (since it puts a hat on y). Predictions are then given as $\hat{\mathbf{y}} = \mathbf{P}\mathbf{y}$. Note that computing the maximum likelihood over the entire training set in one go can be highly costly where N is large. Instead a *Sequential* or online algorithm can be deployed whereby data points are considered one at a time. Sequential algorithms also find home in real-time applications whereby data observations arrive in a continuous stream, or predictions must be made before all of the data points can be observed. One of the main algorithms that achieves this is *Stochastic Gradient Descent* (SGD), where parameters are updated as:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta\nabla_{\mathbf{w}}\mathcal{E}(\mathbf{w}) \tag{6.9}$$

where t denotes the iteration count, and η determines the step size of the gradient, or learning rate. $\mathbf{w}$ is initialised to some starting vector $\mathbf{w}^{(0)}$ when the algorithm begins. Another iterative method is the *Newton-Raphson* update, which takes the form:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \mathbf{H}^{-1}\,\nabla_{\mathbf{w}}\mathcal{E}(\mathbf{w}) \tag{6.10}$$

where $\mathbf{H}$ is the *Hessian* matrix whose elements comprise the second derivatives of $\mathcal{E}(\mathbf{w})$ with respect to $\mathbf{w}$. Here the first and second derivatives of the sum-of-squares error function are:

$$\nabla\mathcal{E}(\mathbf{w}) = 2\mathbf{\Phi}^{\mathrm{T}}\left(\mathbf{\Phi}\mathbf{w} - \mathbf{y}\right) \tag{6.11}$$

$$\mathbf{H} \equiv \nabla^2\mathcal{E}(\mathbf{w}) = \mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi} \tag{6.12}$$

note that these values mean that for the least-squares solution, this formula gives the exact solution in one step.

Bayesian Analysis of Cell Cycle Protein levels

Pertaining to section 3.2.1, here we go into further details regarding preliminary Bayesian Linear Regression of cell cycle protein. Here we only cover G1 (t) phase as we assume the results correspond in similar fashion across all phases. The prior distributions for all subsequent models were drawn as:

$$\sigma \sim \text{HalfCauchy}(\beta_0) \tag{6.13}$$
$$w \sim \mathcal{N}(\mu_0|0, \sigma_w^2) \tag{6.14}$$

using fixed hyperpriors $\beta_0 = 10, \sigma_w^2 = 10$. There is an analytical solution to this problem, however we used MAP estimates for initialization, with NUTS sampler from PyMC3 Python package to perform Markov-Chain-Monte-Carlo sampling ($k = 10^3$ samples) of the posterior and posterior predictive.

mRNA to protein Firstly, we model the parameter uncertainty with regards to mRNA-to-protein expression level using the likelihood:

$$\mathbf{p}^{(t)} \sim \mathcal{N}(w_0 + w_1\mathbf{m}^{(t)}, \sigma^2) \tag{6.15}$$

We do not see any significant divergence in uncertainty from the σ^2_{ML} estimate, with $w_1 = 0.99\pm0.024$. To ensure that outliers do not play a disproportionate effect on these estimates, we also sample the posterior using a Student T distribution ($\nu \sim \mathcal{U}(0, 20)$), where we see no significant difference in parameterization.

Translation to protein Now our likelihood function becomes:

$$\mathbf{p}^{(t)} \sim \mathcal{N}(w_0 + w_1\mathbf{r}^{(t)}, \sigma^2) \tag{6.16}$$

Once again we see a slightly lower slope at $w_1 = 0.931\pm0.016$, slightly better reflecting the impact of degradation on protein level. The same results hold for T-distributed predictors as before. For visualizations see Figure S1.

mRNA with translation to protein Here we include both terms into the likelihood as:

$$\mathbf{p}^{(t)} \sim \mathcal{N}(w_0 + w_1\mathbf{m}^{(t)} + w_2\mathbf{r}^{(t)}, \sigma^2) \tag{6.17}$$

Our weights $w_1 = 0.317 \pm 0.03$ and $w_2 = 0.823 \pm 0.018$

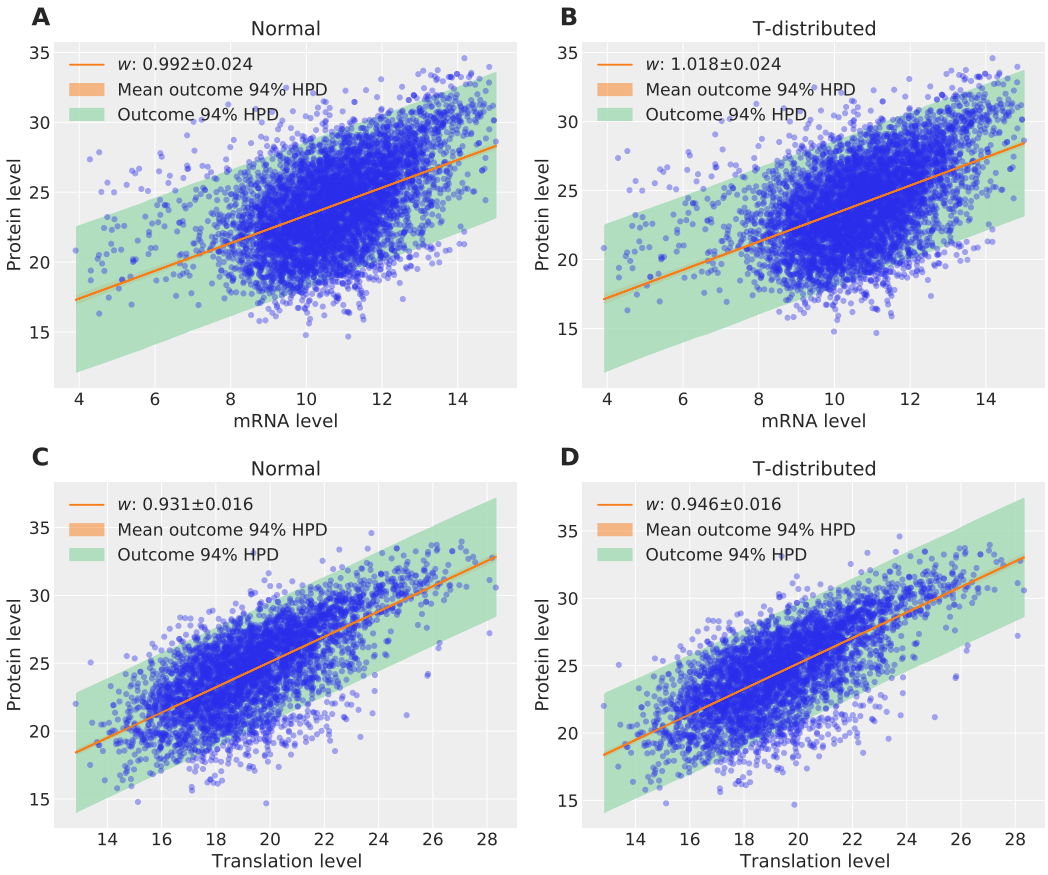


Figure S1: *Scatterplots of (A-B) mRNA vs protein and (C-D) translation vs protein with Bayesian HDI intervals.*

Feature Selection Approaches

Pertaining to section 3.2.2, we cover here in further detail the process of feature selection across 30 or SDFs by building models that map to protein abundance within the HeLa cell cycle dataset. Our methods of feature selection we will cover are:

1. Recursive Feature Elimination (RFE)
2. L_1 sparsity-inducing regularization (LASSO)
3. Selecting k-Best (ANOVA)

For all of our models using gradient-boosted regression trees (GBRT), these are our tuned parameters from model selection:

1. G1; learning rate = 0.02, max depth = 3, min samples leaf = 10, 'n_base_estimators' = 1000
2. S; learning rate = 0.01, max depth = 3, min samples leaf = 10, n_base_estimators = 1000
3. G2/M; learning rate = 0.02, max depth = 3, min samples leaf = 5, n_base_estimators = 1000

Feature matrix $\mathbf{X}$ is always standardised using Z-score, $\mathbf{y}$ is not standardized. G2 and M are inter-changeable as labels for the final cell cycle phase.

RFE Firstly, we combine recursive feature elimination method from Scikit-Learn (python) [177] with 10-fold CV to automatically filter features of interest, employing a greedy-backward algorithm for iteratively removing uninteresting features. We use a GBRT as the estimator, with 1000 base decision tree estimators, using MSE as a normalized RSS function (see 2.14). The results of this only drop around 2 features per cell cycle phase, indicating that MSE does not seem to improve when features are dropped (Fig S2).

We see a clear a clear threshold area around 7-8 features where -MSE appears to flatten out and not increase, therefore we formulate a method where we choose the number of features to select based on the change in MSE with respect to each subset, which is approximated using Euler method:

$$\frac{dMSE}{dF} \approx (MSE_{n+1} - MSE_n) < \eta \tag{6.18}$$

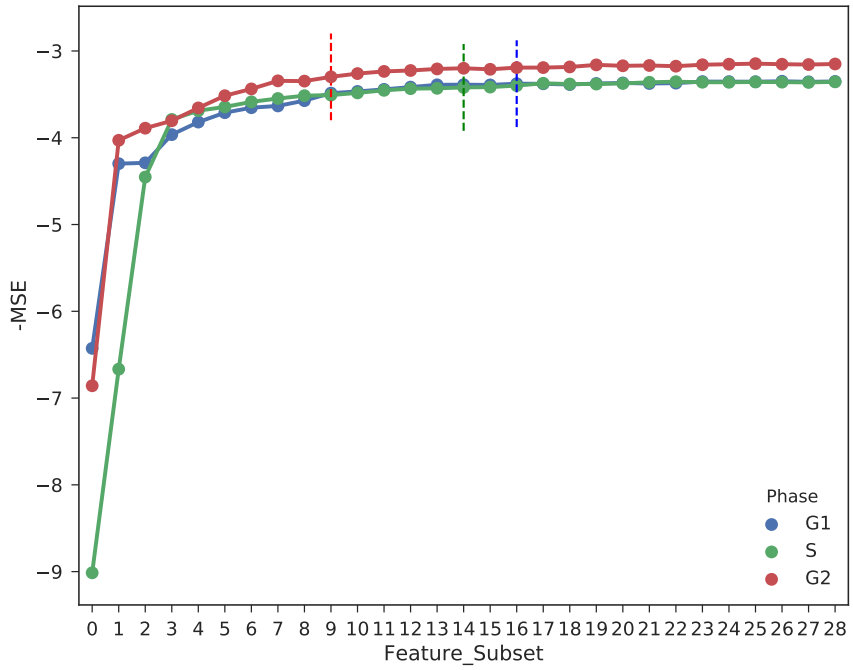


Figure S2: RFECV fails to select a suitable feature subset.

where we choose η as a small constant tolerance, where the number of features is where the change begins to slow down in terms of -MSE increase. Here we choose $\eta = 0.005$. Lines are drawn on the graph as to n_features selected ($\hat{p}$) per cell cycle phase, colour-respective.

Using $\hat{p}$, we generate new models of GBRT without CV, indicating the number of features we want the recursion to stop at. This yields a ranking for each of the features, per cell cycle phase, in addition to feature importances for the features that remain (Fig S3).

These selected features are stored and used later on in LOOCV analysis to reduce the feature matrix.

ℓ_1 **Regularization** Firstly, to find an optimal α for the regularization term, we use LASSO without GBRT, using a brute-force grid search method, across 10-fold cross validation, with $\alpha \in [10^{-3}, 10^{1}]$, using the negative mean

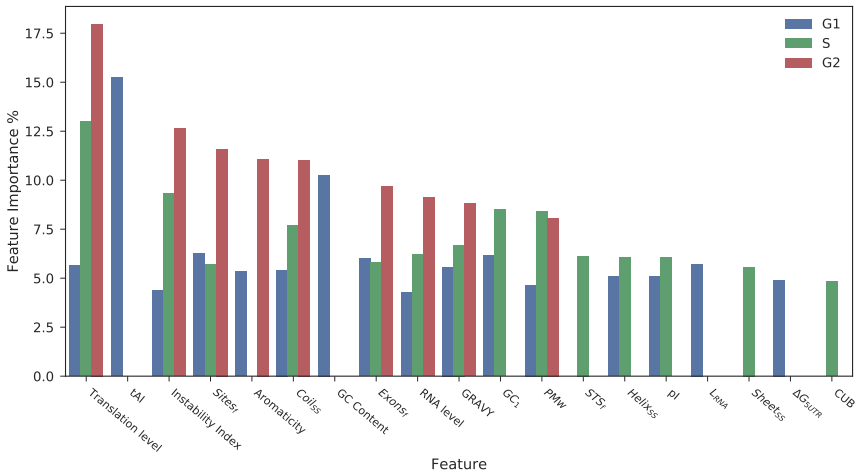


Figure S3: Feature importances generated from RFE-GBRT.

squared error as before as the scoring function (6.1).

We begin to see a drop in accuracy around $\alpha \approx 0.1$, again with no real increase in -MSE at any particular stage within the range.

We then repeat the run by not predicting using the LASSO regressor, but using the sparsity-inducing coefficients to induce sparsity in a reduced X matrix, which is then used as input to a GBRT model. The results of these are shown in Fig 3 in the main paper. For Fig S4, like the RFE example we choose a threshold η which signifies where the change in MSE should not exceed by, the inverse of described in equation 6.18. We chose $\eta = 0.02$ in LASSO case. We then generated 10-fold CV test scores (r^2) across a grid of 30 α points, as $\alpha \in [10^{-3}, 1]$, for each cell cycle phase. This generated optimal $\alpha = [0.05, 0.07]$ range. Using these regularisers, we generated final GBRT models using single alpha per cell cycle phase, using 10-fold cross validation and extracting the feature importances from each estimator, and plotting the mean ($\pm$SD as error bars) in Fig S4.

Select K Best In this feature selector, we optimise to find the most suitable k using Analysis of Variance (ANOVA). This entails, for a given k, calculating the F-value for each feature calculated as the covariance between

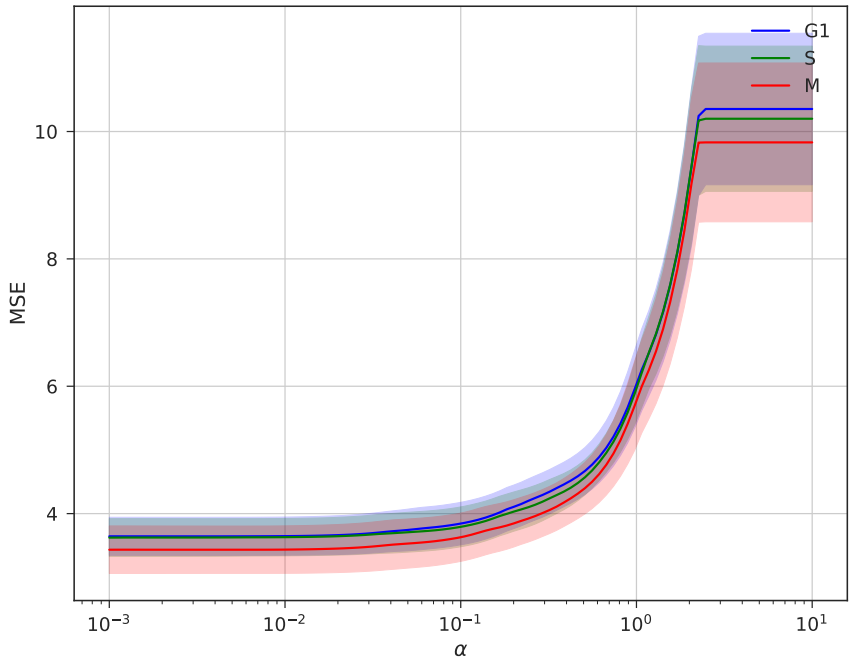


Figure S4: LASSO only parameter tuning of α versus negative MSE. *Error bars ±SD 10-fold CV.*

features, and selecting features with the largest F-value. We create a grid of $k \in [1, P]$, where P is the total number of features initially, then we use a Pipeline object in Scikit-Learn to reduce $\mathbf{X} \in \mathbb{R}^{N \times P}$ with selecting the $k \in P$ best features as described above, then using a GBRT model for prediction. Again we use the negative mean-squared error as a scoring function (6.1), with 10-fold CV.

We see that unlike the first 2 feature selectors, there is not an exponential curve but rather linear decreases in MSE beyond $k = 3, 4$. Therefore there was no clear threshold to choose to find an optimal $\hat{k}$, so we followed the default settings in scikit-learn [177], which by default selects $\hat{k} = P/2$ to be half of the original number of features. In this case, $\hat{k} = 14$ for all cell cycle phases.

Using optimal $\hat{k}$ we created a GBRT model with 10-fold cross validation, as with the previous 2 feature selection procedures (Figure S6). Interestingly

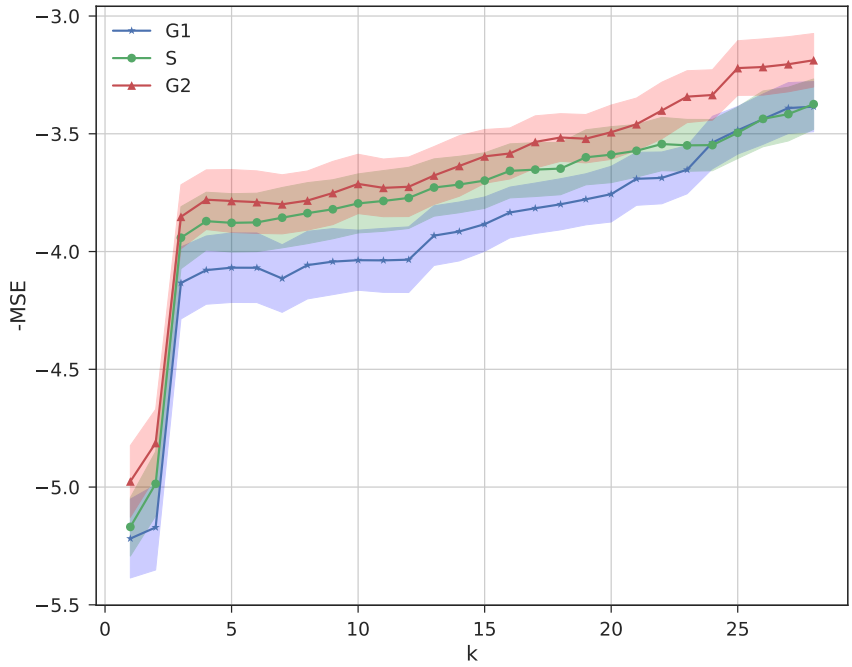


Figure S5: Parameter tuning of k versus -MSE of resulting model on test data. *Error bars ±SD 10-fold CV.*

ANOVA chooses very different features in order compared to the 2 previous selectors, ranking mRNA level quite high with some mRNA-derived features. This is likely due to the Gaussian nature of mRNA abundance, compared to a number of the SDFs which often do not follow a Gaussian distribution.

Text Feature Table Descriptions

Pertaining to section 4.2.1, we cover here the full description pertaining to each mRNA and amino-acid derived text feature.

Amino acid (from SwissProt) Here we describe the text-mined features from Uniprot/Swissprot and how these features are classified according to their database:

Molecular processing features

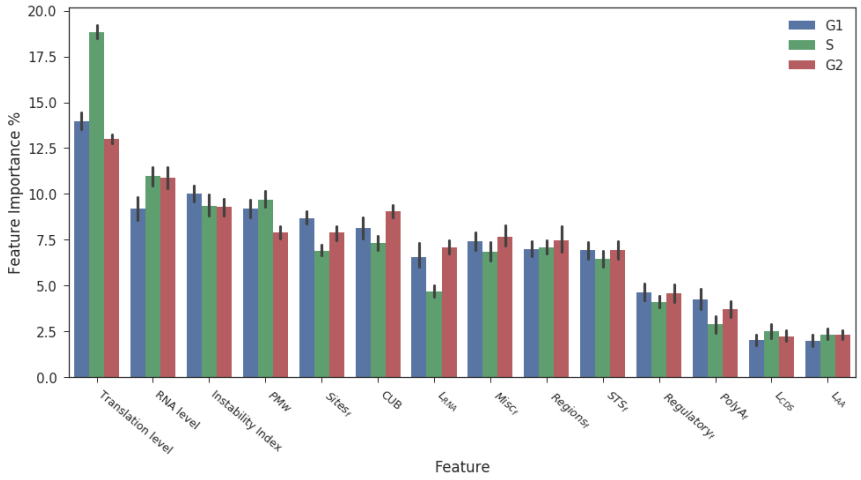


Figure S6: Feature importances generated from KBest-GBRT. *Error bars ±SD 10-fold CV.*

- Initiator methionine: Cleavage of the initiator methionine
- Signal peptide: Sequence targeting proteins to the secretory pathway or periplasmic space
- Transit peptide: Extent of a transit peptide for organelle targeting
- Propeptide: Part of a protein that is cleaved during maturation or activation
- Chain: Extent of a polypeptide chain in the mature protein
- Peptide: Extent of an active peptide in the mature protein

Amino Acid Regions

- Topological domain: Location of non-membrane regions of membrane-spanning proteins
- Transmembrane: Extent of a membrane-spanning region
- Intramembrane: Extent of a region located in a membrane without crossing it
- Domain: Position and type of each modular protein domain
- Repeat: Positions of repeated sequence motifs or repeated domains

- Calcium binding: Position(s) of calcium binding region(s) within the protein
- Zinc finger: Position(s) and type(s) of zinc fingers within the protein
- DNA binding: Position and type of a DNA-binding domain
- Nucleotide binding: Nucleotide phosphate binding region
- Region: Region of interest in the sequence
- Coiled coil: Positions of regions of coiled coil within the protein
- Motif: Short (up to 20 amino acids) sequence motif of biological interest
- Compositional bias: Region of compositional bias in the protein

Amino acid sites

- Active site: Amino acid(s) directly involved in the activity of an enzyme
- Metal binding: Binding site for a metal ion
- Binding site: Binding site for any chemical group (co-enzyme, prosthetic group, etc.)
- Site: Any interesting single amino acid site on the sequence

Amino acid modifications

- non-standard residue: Occurence of non-standard amino acids (selenocysteine and pyrrolysine) in the protein sequence
- modified residue: Modified residues excluding lipids, glycans and protein cross-links
- Lipidation: Covalently attached lipid group(s)
- Glycosylation: Covalently attached glycan group(s)
- Disulfide bond: Cysteine residues participating in disulfide bonds
- Cross-link: Residues participating in covalent linkage(s) between proteins

Natural variations

- Alternative sequence: Amino acid change(s) producing alternate protein isoforms

- Natural variant: Description of a natural variant of the protein

Secondary structure

- Helix: Helical regions within the experimentally determined protein structure
- Turn: Turns within the experimentally determined protein structure
- Beta strand: Beta strand regions within the experimentally determined protein structure

Methodology of Vogel/Parkes sequence and SDF analysis

Pertaining to section 4.2.2, here we cover the experimentation and methodology details when performing co-correlate analysis to the Vogel [6] dataset.

Data Preparation Firstly, we loaded in Vogel's sequence data ($n = 1051$) and SDF ($n = 476$, $p = 135$) data separately, in conjunction with our sequences derived from NCBI ($n = 51837$) and SDFs ($n = 17440$, $p = 211$). Since our SDFs are labelled with a Refseq primary key, and Vogel used EnsemblIDs, we firstly downloaded the Biomart Ensembl-Refseq ID dataset ($n = 48628$) and performed an intersection operation between Biomart-Parkes ($n = 47957$), merging on Refseq ID, and then between Parkes-Vogel ($n = 933$), merging on EnsemblGeneId key.

Pairwise Sequence Alignment We used the Biopython Align [142] package to calculate both pairwise global and local alignments, using arguments match=1, mismatch=-1, open=-1 and extend=-1. We then take the score of each resulting alignment and normalize by gene length, then take the μ average across alignment results. Note that due to the underlying matrix being $\mathcal{O}(L^2)$ on memory, where $L = \max(L_1, L_2)$ is the largest gene length between the pair, a dynamic programming approach is relatively fast but we do not perform alignment on sequences over 14k in length due to memory consumption.

Feature Correlations In this case we load our SDF which has undergone three transformations, see the SDF Preprocessing subsection of Data Preparation within Chapter 4 for details as to how we transform the features. The reason just one transformation is not chosen is that we are not always certain how Vogel has preprocessed some their features, with documentation within the Supplementary material paper not always clear. We use the unscaled features, normalization 1 and 2 inputs. We then select the features that pair across Parkes-Vogel datasets, leading to 84 features which are labelled similarly. We then compute Spearman-rank r_s correlations between each Parkes-Vogel pair for our 3-SDF transformations, and select the *maximum* r_s across transformations. This way we are more likely to capture the true transformation.

Partial correlations between sequence-derived features and protein abundance, factoring mRNA and length

In this section we explore the partial correlations between sequence-derived features (SDFs) and protein abundance, fixing for various factors such as mRNA expression and length. Recall that from section 2.3.4 we define partial correlation as $\rho(\mathbf{x}, \mathbf{y}|\mathbf{Z})$ [119]. This section pertains to section 3 of the second paper produced by Parkes et al. but was viewed as significantly divergent from the main thrust of this thesis. We briefly mention this work within the second chapter, second section of this thesis.

The relationship between log-normalized protein and mRNA concentrations is non-linear, but can be estimated using piece-wise linear functions [6] and is observed in many studies [147, 3, 137, 5, 1]. However, a significant portion of SDFs are not normally distributed or cannot be easily transformed to be distributed as such; this presents a challenge in emulating the kind of information contained in the mRNA expression to be represented by SDFs. We calculate the monotonic relationship between each SDF and the corresponding mRNA, protein (left) and partial protein (right) expression for 5 different *H. sapiens* cell lines (*Daoy* [6], *A431*, *U2OS*, *U251MG* [164], *HeLa* [5]), corresponding to three different studies, to explore the capacity of SDFs as proxies (see Figure S7). We define 'partial protein' as the correlation between SDF-

to-protein, controlling for mRNA level. Whereby 'y=x' represents features that correlate just as much with mRNA as protein abundance (meaning the feature is agnostic), in all cell lines we notice that post-translational modifications (PTM) correlate strongest with both mRNA and protein abundance; particularly Ubiquitination ($r = 0.4, 0.6$) which is associated with protein degradation. mRNA/protein length features are strongly negative with respect to protein level, particularly in HeLa/Daoy cell lines ($r = -0.4, -0.6$). The majority of changes in correlation when mRNA is controlled for is not substantial; most features are corrected for by $\Delta r < 0.2$ (see S5-6 Figure.), and nearly always converging the correlation towards zero, rather than increasing it. Similar observations are made when the sequence length (mRNA/amino acid) is controlled for instead of mRNA level (see Supplementary Material 5.), particularly codon bias features are appropriately reduced in correlative power.

One clear trend is the 'flattening' of the relationship for partial-correlations as the fixing of mRNA level has the tendency to drive partial-protein correlations to 0, and this is particularly noticeable in the Doay and HeLa cell lines. The variance of correlated features is substantially higher in Aviner's dataset than Lundberg; we suspect this is because HeLa is drawn from a cell cycle study where dynamic time-effects are not removed in the preprocessing pipeline, whereas Lundberg [164] measurements are aggregated and more likely to conform to steady-state protein levels. This loss of information could therefore lead to small correlation coefficients in these sets. Alternatively, differences in RNA sequencing technology (Aviner et al. [4, 5]; microarray, Lundberg et al. [164]; RNA-Seq) are known to vary in variance and this may be reflected in the variance of correlation coefficients. These results are checked against the estimated mutual dependency (MI) conditioned against mRNA level or gene length, where we find similar loss in the relationship between SDFs and protein; in particular we see that the mutual information reflects the same relationship either against mRNA or protein level, or both.

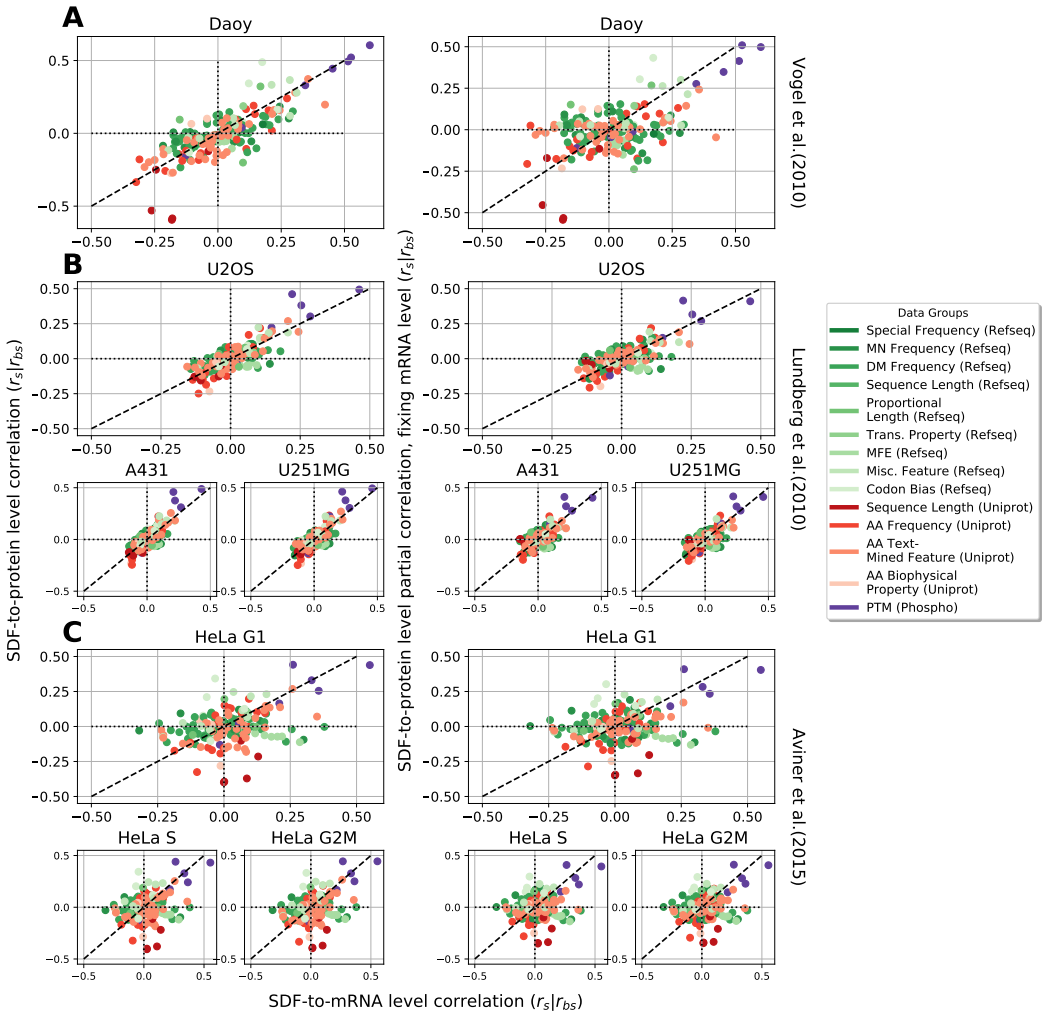


Figure S7: Variations in a priori experimental dataset assumptions considerably alter SDF impactfulness. *Scatterplots of Spearman-rank (r_s)/Biserial correlation (r_{bs})-mixed correlation between sequence-derived features (SDFs) and mRNA (x-axis), protein (y-axis, left) and partial protein fixing mRNA (y-axis, right). From top; (A) Daoy (Vogel et al.), (B) U2OS;A431;U251MG (Lundberg et al.), and (C) HeLa G1;S;G2/M (Aviner et al.) from different mRNA-protein expression datasets. Each point represents a correlation between an SDF and an expression dataset. See Methods for details on correlation method. mRNA/RNA features are denoted in green shades, amino-acid features are denoted in red shades, PTM features are denoted in purple.*

Detailed Model Selection of Sequence-Derived Features

Pertaining to section 5.2.1, here we explore the exact processes that went into generating the main results of model selection with respect to SDF vs. mRNA and protein abundance across multiple cell lines. For regression models with hyperparameters we selected:

- **OLS**: No hyperparameters
- **Ridge**: $\alpha \in [10^{-4}, 10^{1.5}]$, 50 samples.
- **Elastic Net**: $\alpha \in [10^{-4}, 10^{1.5}]$, 50 samples. $\gamma \in [0, 1]$, 3 samples. See equation 2.23 for mathematic formulation.
- **SVM**: $C \in [10^{-3}, 10^{1.5}]$, 50 samples. C acts as a regularizing coefficient. See scikit-learn documentation.
- **Decision tree**: Max depth: [2, 3, 4], Max features: ['auto', 'sqrt'].
- **Random Forest**: Number of sub-trees/estimators $\mathcal{T}$: [10, 200], 6 samples.
- **Histogram GBRT**: ℓ_2 regularization: 10^{-2}. learning rate $\eta = [10^{-2}, 0.25]$, 4 samples. Max iterations: [100, 250].
- **XGBoost**: Learning rate $\eta = [10^{-4}, 10^{1/2}]$, $\alpha = [0.001, 0.1]$, using HalvingGridSearch.
- **MARS (Earth)**: Default parameters.
- **Ensemble (Voting)**: Combination of Ridge, ElasticNet and Linear SVM models in equal weighting with aforementioned hyperparameter ranges for each model as above.

We also consider a ensemble regression model which is a voted aggregate of several regression models with optimal hyperparameters. We then define 5 cell line within 7 datasets as U2OS, A431, U251MG, Daoy, HeLa (G1, S, G2/M). We compute estimates of RMSE and adjusted r^2 using 5-fold cross validation across the product of all dimensionality-reduced subsets $\mathbf{\Phi}^{(m)}$, along with cell lines $\mathcal{C}$. This is achieved for each subset, cell line pair in the following order:

1. Merge together dataset $\mathbf{\Phi}^{(m)} \cap \mathbf{y}_{\mathcal{C}}$ with expression set using HGNC/Biomart labels.

2. Preprocess numeric columns with low-variance filtering ($\sigma^2 < 1e-6$) and missing value (proportion of $N < 0.3$) filters.

3. Select appropriate subset of columns, dropping label columns and non-numeric. Recognised categorical or binary data is transformed using one-hot encoding.

4. Split merged dataset into random training (80%) and validation (20%) sets $\tilde{\mathbf{X}}_{\text{train}}, \tilde{\mathbf{X}}_{\text{test}}$ and $\tilde{\mathbf{y}}_{\text{train}}, \tilde{\mathbf{y}}_{\text{test}}$, dropping to conform to the number of y data points.

5. Generate grid search and compute 5-fold cross validation training and testing errors fitting each model to $\tilde{\mathbf{X}}_{\text{train}}$ and $\tilde{\mathbf{y}}_{\text{train}}$. XGBoost, RandomForest and Ensemble/Voting models are fitted using HalvingGridSearch, rather than exhaustive grid search, which manipulates subsample sizes prior to fitting to achieve a performance boost at the risk of more inaccurate hyperparameter estimates. For XGBoost/RandomForest we use the *n_estimators* parameter for halving instead of N, i.e the number of weak learner trees $\mathcal{T}$.

6. (Post-model selection) Re-fit on best models using the parameter runs with the largest mean adjusted r^2 test scores.

7. (Optional) Compute validation r^2 scores using $\tilde{\mathbf{y}}_{\text{test}}$ and predicted values $\hat{\mathbf{y}}_{\text{test}}$.

Model Selection of Protein Abundance using SDF, PPI, HL and mRNA

Pertaining to section 5.2.3, we define the key steps prior to and during model fitting.. All models are fitted with either OLS, Ridge regression with ℓ_2 regularization, or XGBoost (Gradient-boosted regression trees)[112]. The choice of model is indicated in Figure 5.7 by colour. All models including SDFs as input use the sPCA—MCA unsupervised learning preprocessor $\mathbf{\Phi}^{(m)} \in \mathbb{R}^{N \times P}$, with $N = 15269$, $P = 137$. Protein-protein interaction feature matrix $\mathbf{M} \in \mathbb{R}^{N \times K}$ is computed from network analysis derived from the

STRING database [171]. mRNA and protein half-life information (for HeLa) is taken from Tani et al. [169] and Cambridge et al. [170] respectively. Paxdb database information [168] is extracted and abundances are log-2 plus 1 normalized. Given that corresponding mRNA abundance is not provided with this data, any analyses involving measured mRNA abundance are excluded from the figures. Firstly, we convert half-life $t^{\frac{1}{2}}$ in hours into decay constants k^{deg} using the equation:

$$k_n^{\text{deg}} = \frac{\ln 2}{t_n^{\frac{1}{2}}} \tag{6.19}$$

where decay constants assume constant logarithmic decay. The decay constants are normally distributed across the protein population, which makes them fit nicely within statistical modelling. Now we will overview the modelling pipelines for each regression model:

- **OLS and Ridge**: Split data into training and testing sets, perform model fitting with OLS with 5-fold cross validation. Refit the model on the training data and score on the testing data. For Ridge, regularization parameter $\alpha = 1$ is left to default.

- **XGBoost**: Split data into training and testing sets, compute an adjusted r^2 metric based on N and P, perform hyperparameter search to find best η, α, λ. Refit XGBoost with best parameters on whole N, and score on the testing data. Learning rate $\eta \in [10^{-4}, 0.2]$, $\alpha \in [10^{-4}, 10^{-1}]$ and $\lambda \in [10^{-4}, 10^{-1}]$. We draw $g = 100$ guesses for each parameter on 40% of the training data, from log-uniform distributions within the aforementioned ranges.

For non-trivial regressions, i.e including more than one source of database input, an intersection merge operation occurs between $\mathbf{\Phi}$ and $\mathbf{p}$ just prior to train-test-splitting. This process usually leads in a modest reduction in N, particularly if multiple datasets are integrated. Several of the analyses involving the Daoy cell line were dropped due to insufficient sample size N.

Appendices

Feature name	Mol type	Abbr.	Description
mRNA length	RNA	L_{RNA}	The length in base pairs of the mRNA sequence
Instability Index	protein	**	A measure of protein stability in a test tube
Protein molecular weight	protein	PMw	The estimated weight of all the amino acids
Isoelectric-point	protein	pI	The pH at which a molecule carries no net electric charge (neutral)
Grand Average of Hydropathy	protein	GRAVY	The mean of hydropathy values for each amino acid
Aromaticity	protein	**	The mean of aromatic amino acids
GC content	RNA	GC	The proportion of G and C bases in the sequence
Effective number of codons	RNA	Nc	Quantifies how far codon usage of a gene departs from equal usage
Codon length	RNA	L_{CDS}	The number of codons in the sequence
Secondary structure coil	RNA	$Coil_{SS}$	The estimated proportion of CDS in a coiled-secondary structure
Secondary structure helix	RNA	$Helix_{SS}$	The estimated proportion of CDS in a helix-secondary structure
Secondary structure free energy	RNA	ΔG_{5UTR}	The estimated free energy release from secondary-structure bonds forming
Codon usage bias	RNA	CUB	The bias usage among synonymous codons for CDS
Codon Adaptation Index	RNA	CAI	A similar metric to CUB, uses estimates for the population to normalize against
tRNA Adaptation Index	RNA	tAI	The bias usage among tRNA molecules during translation
Number of exons	RNA	Exons_f	The number of exons scaled by gene length
Number of PolyA tails	RNA	PolyA_f	The number of polyadenylated regions identified in mRNA sequence
Number of STSs	RNA	STS_f	The number of sequence-tagged sites in the sequence

Table S1: *Engineered SDF features used in Parkes & Niranjan (2019) [1].*

Method	Technique	Description	Pros	Cons	Target?
Missing Value Filter	Filter	Drops columns that have missing values above threshold percentage	Simple	Arbitrary selection of drop threshold	No
Low Variance Filter	Filter	Drops columns that have variance below some threshold	Intuitive with ML	May have no impact on predictability	No
High Correlation Filter	Filter	Drops columns with low correlations to each other/target	Reduces multicollinearity, improves model stability	Only works with linear or monotonic relationships	Yes/ No
Random Forest	Feature Selection	Uses tree-based models which contain in-built feature selection	Handles nonlinear relationships, fast	Models can be too complex, do not generalize	Yes
Recursive Feature Elimination (RFE)	Feature Selection	Iteratively trains a model and drops a feature at each step until model change is less than tol	Informative on each features' impact	Assumes underlying model fits well, Computationally expensive	Yes
Forward Feature Selection (ANOVA)	Feature Selection	Trains an additive model using F-score 1 feature at a time	Simple, established technique within biology	Only works with linear relationships	Yes
Factor Analysis	Dimensionality Reduction	Variables are grouped by correlations, known as 'factors'	Factors theoretically relate to real-world phenomena	Number of factors must be known beforehand, factors difficult to observe	No
PCA	Dimensionality Reduction	Extracts a subset of uncorrelated principle components using SVD	Computationally cheap, provides explained variance, nice mathematical properties	Only produces linear combinations, selection of K not always intuitive	No
ICA	Dimensionality Reduction	Reduces parameter space using information theory by maximising kurtosis of projected values	Intuitively similar to PCA, highly theoretical	Assumes variables are linear mixtures, latent variables are mutually independent	No
Manifold Isomap	Dimensionality Reduction	Finds manifolds by projecting points onto lower dimensional space	Handles nonlinear relationships	Assumes continuous manifold, very computationally expensive	No
t-SNE	Dimensionality Reduction	Uses nearest-neighbor techniques to provide low-dimensional representation	Retains local and global data stucture, good visualizations, handles nonlinear relationships	Computationally expensive, large loss of information	No
NMF	Dimensionality Reduction	Factorizes the data matrix into factor and loading matrices	Improved interpretability over PCA, automatic regularization	Requires all non-negative elements	No

Table S2: *Overview of feature selection and dimensionality reduction techniques.*

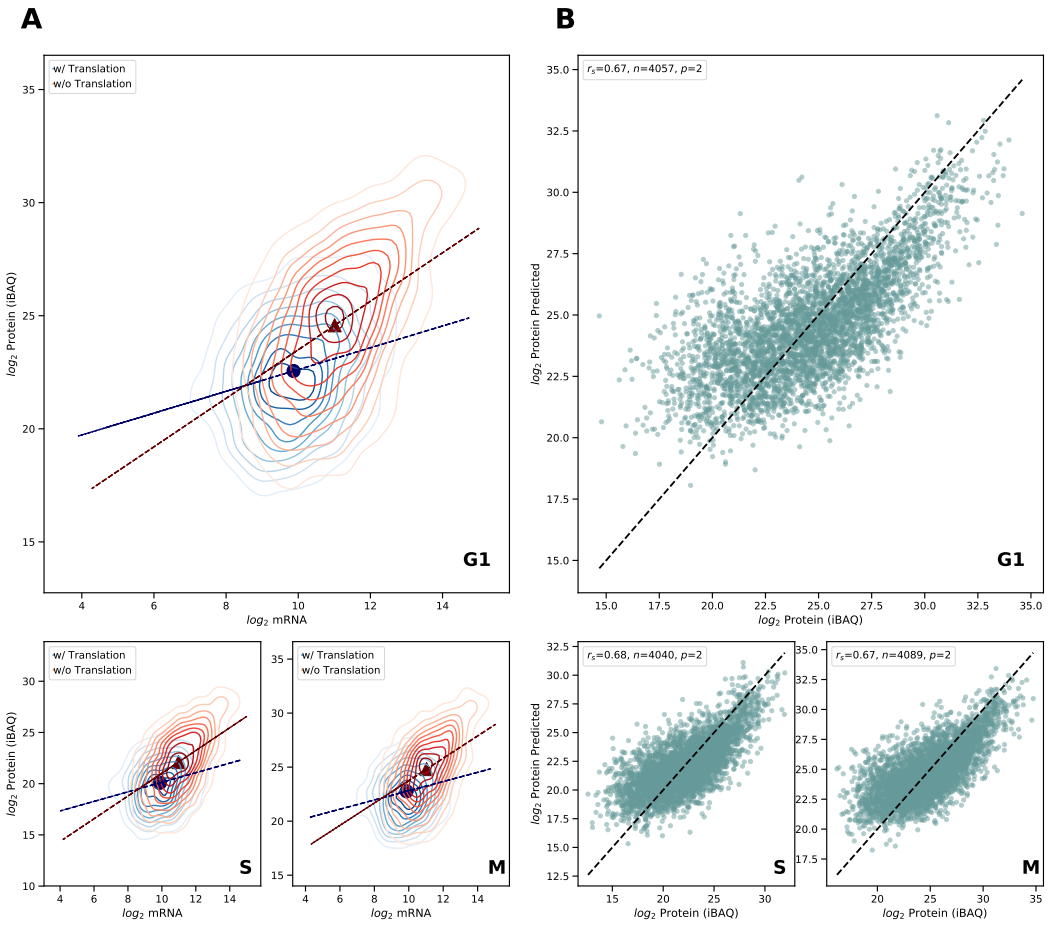


Figure S8: Effects of translation data on model parameters. *(**A**): Contour densities between* $\log_2$ *mRNA and protein with (blue,* $r_s = 0.23$-0.24*) and without (red,* $r_s = 0.46$-0.48*) associated translation measurements. Linear model (black) with mean centre of cluster (shape refers to group). (**B**): Scatterplots of measured (*y*) versus predicted (*$\hat{y}$*) protein across G1, S and G2/M cell cycle phases, using mRNA-translation predictor (see Figure S10).*

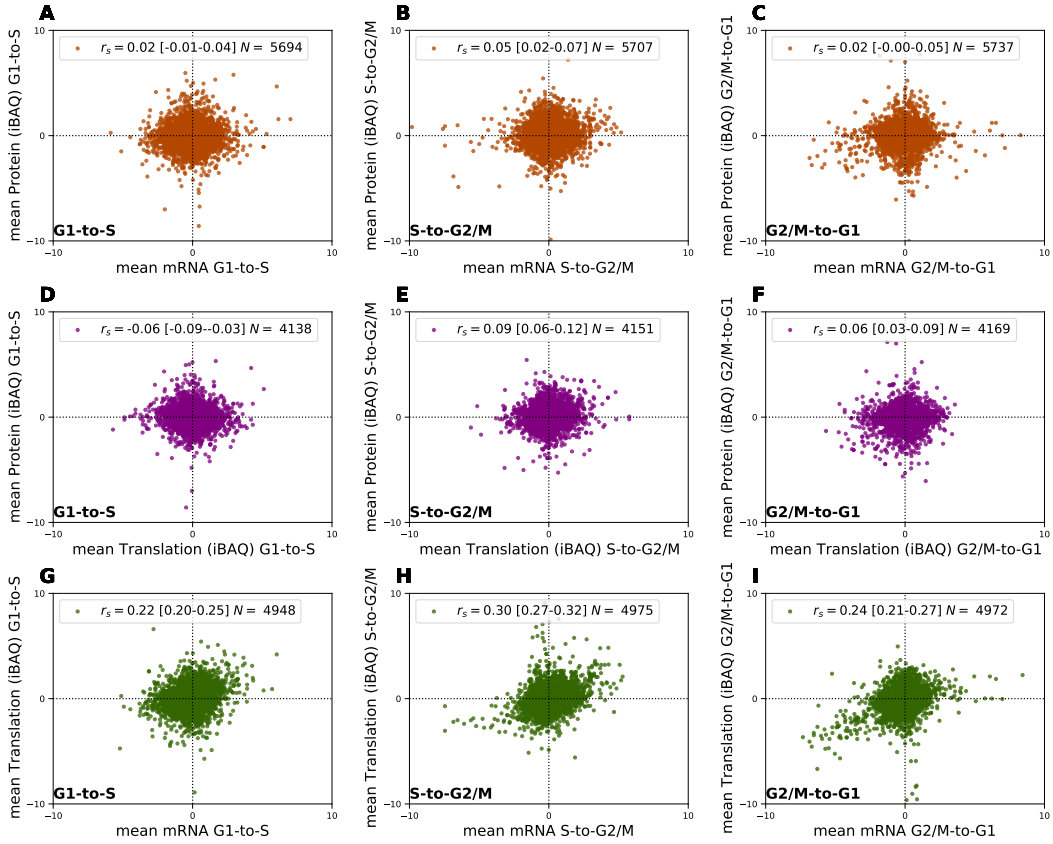


Figure S9: Changes in mRNA, translation and protein across the HeLa cell line. *Scatterplots across G1-S, S-G2/M and G2/M-G1 of z-score transformed (A-C, orange) mean* $\log_2$ *mRNA level against mean* $\log_2$ *protein level, (D-F, purple) mean* $\log_2$ *translation level against mean* $\log_2$ *protein level, (G-I, green) mean* $\log_2$ *mRNA level against mean* $\log_2$ *translation level. Spearman-rank correlations* (r_s) *with 95% confidence intervals and sample size* N *in legends.*

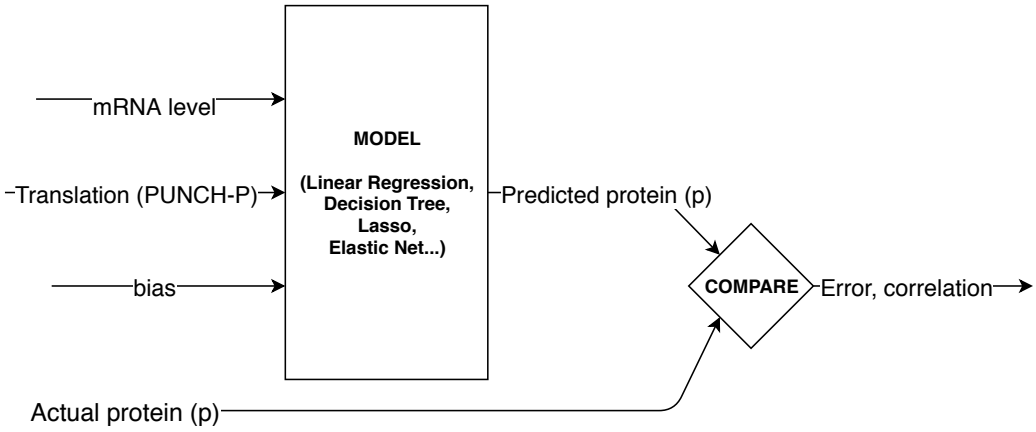


Figure S10: Model for mRNA and translation against protein abundance. *Flowchart diagram describing the model construction for a mRNA-translation predictor. Figure S8B shows the lack of improvement incorporating both mRNA and translation.*

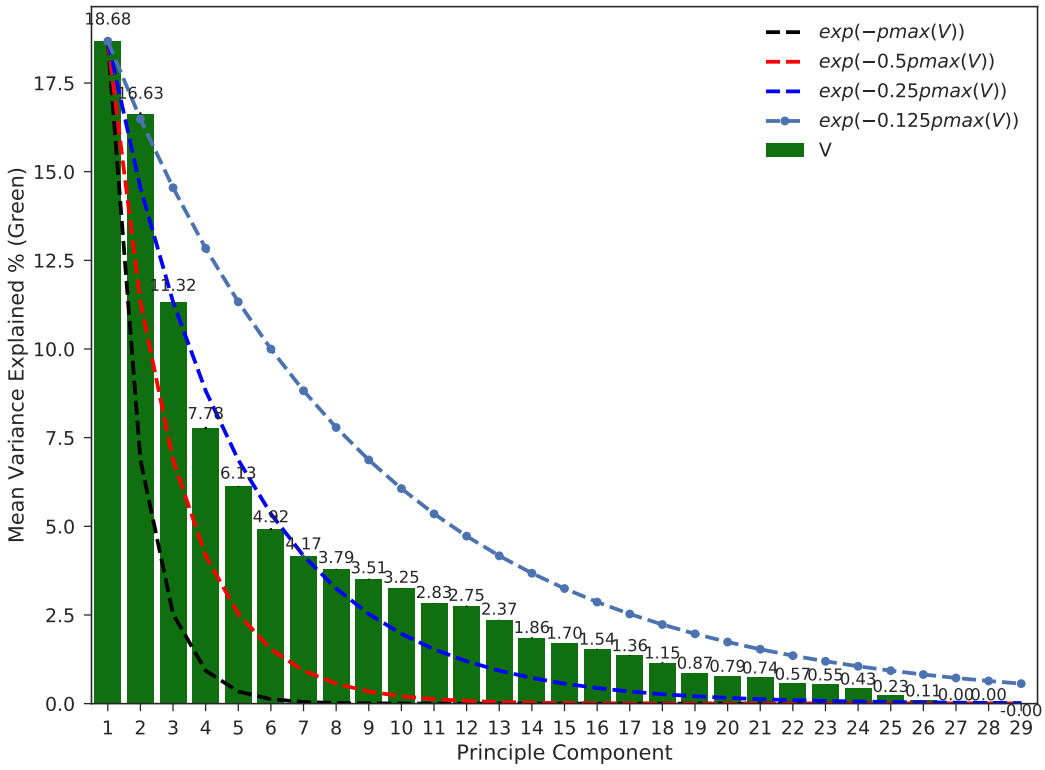


Figure S11: PCA eigenvalue analysis of feature matrix X. *Barplot of percentage of variance explained (scaled singular values) using* $\mathbf{X} \in \mathbb{R}^{N \times P}$ *matrix of 29 SDFs, excluding target protein level. Lineplots of various negative exponential decay curves to illustrate falloff.*

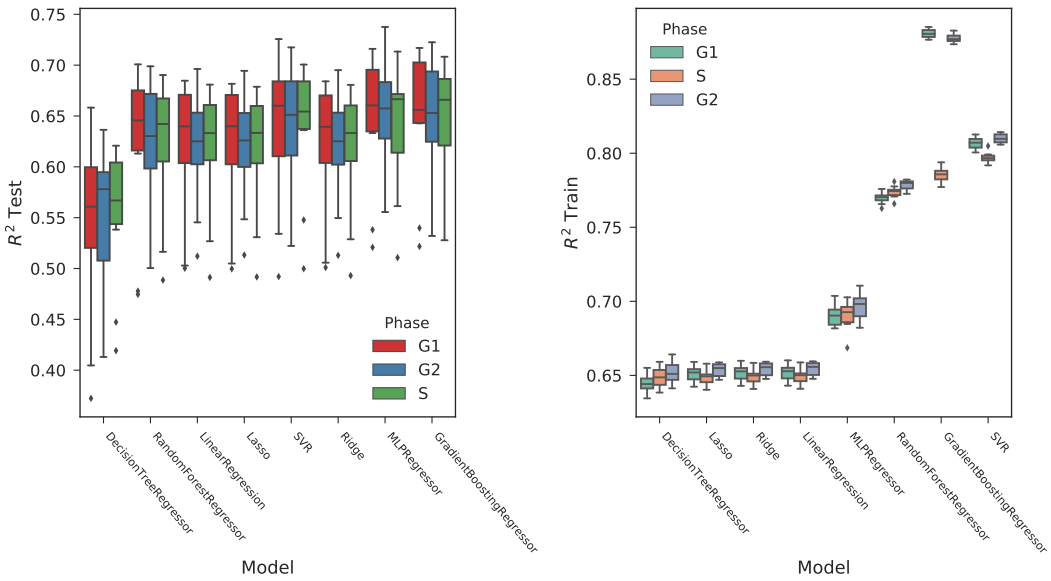


Figure S12: Selecting algorithm with highest correlation using GridSearch 10-fold cross validation. *Barplot representation of different algorithms for training score (right) and testing score (left). Gradient-boosted regression trees (GBRT) performed best across all phases. ±SD indicate cross-validation scores.*

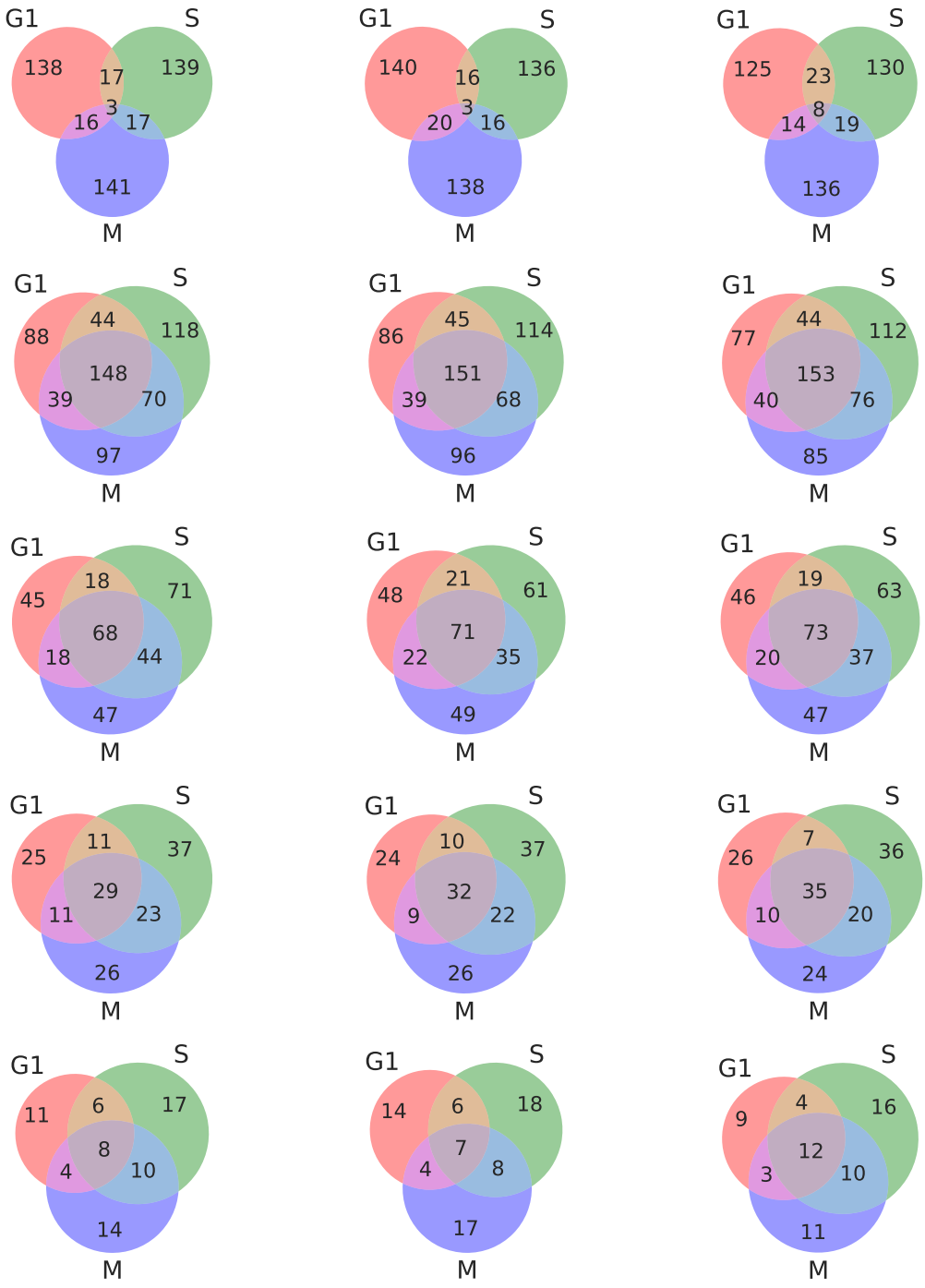


Figure S13: Outlier overlap for all feature selectors across q5, q90, q95, q97.5 and q99. *Venn diagrams across RFE (left), ℓ_1 (middle) and KBest (right) feature selectors, with vertical representing n-th percentiles q5, q90, q95, q97.5, q99 respectively.*

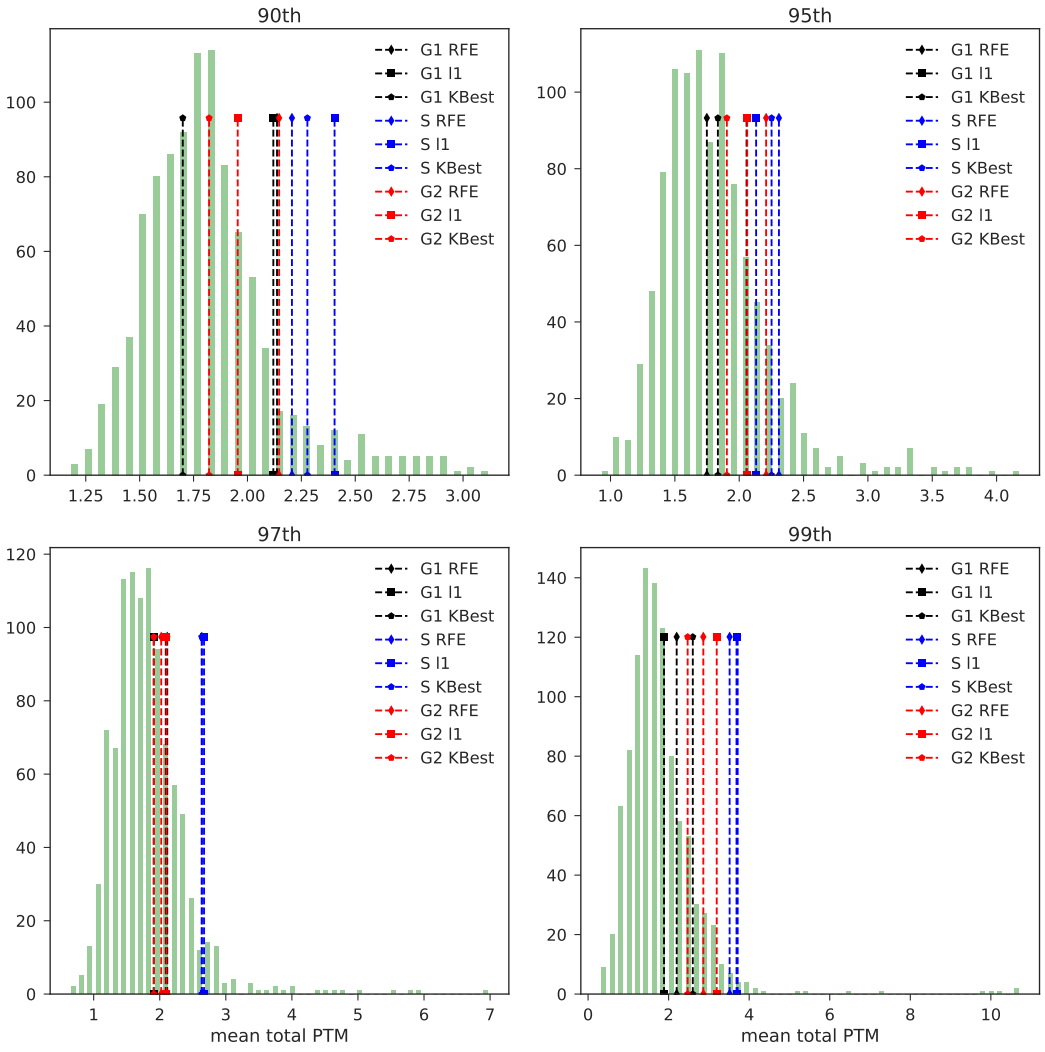


Figure S14: Distributions of random-subsampled PTM sites versus. outlier PTM sites. *Histogram of* 10^4 *bootstrap sub-samples of mean total post-translational modification (PTM) prediction sites versus sample outlier sets (vertical lines), using 90th, 95th, 97.5th and 99th percentiles.*

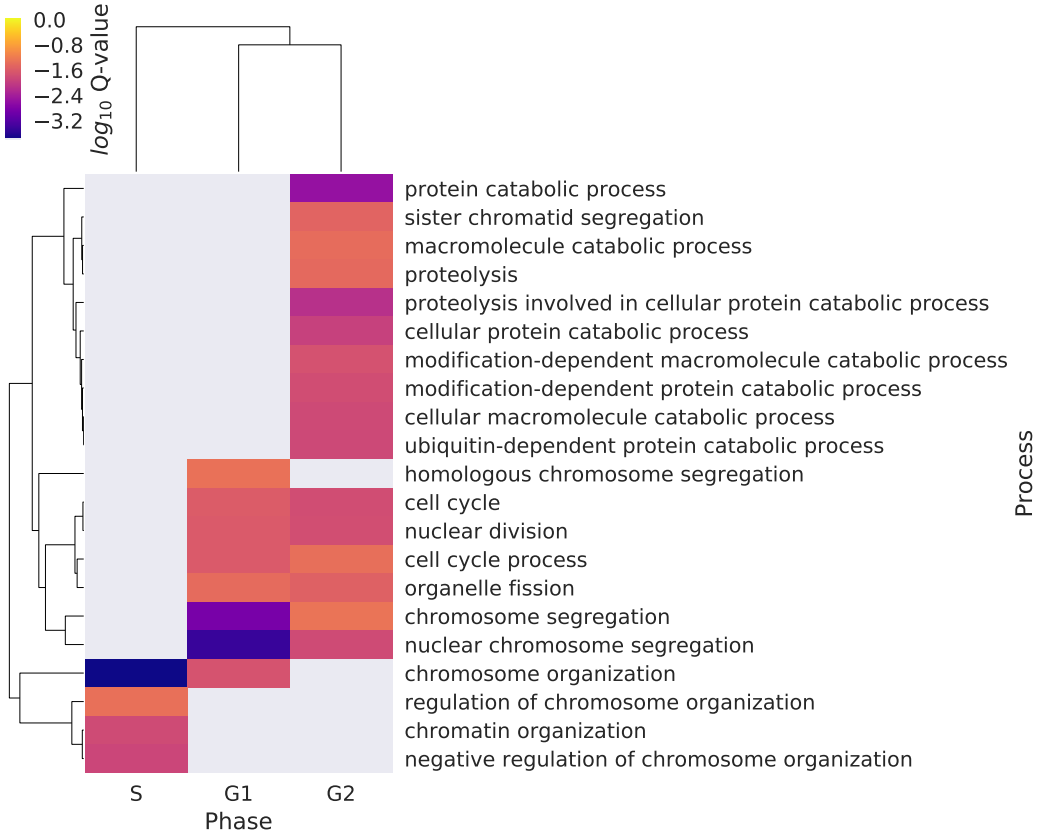


Figure S15: GOBP Analysis of 99th percentile terms. *Hierarchical clustering of* $\log_{10}$ *Q-value GOBP terms by cell cycle phase, using 99th percentile-selected outliers to model Appendix 3.6.*

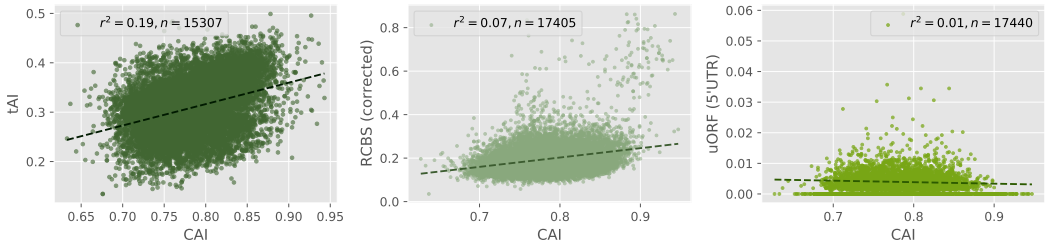


Figure S16: Scatterplots of codon bias features. *Scatterplots of Codon Adaptation Index (CAI) against 1) tRNA adaptation index, 2) Relative Codon Bias and 3) uORF count in the 5'-UTR region.*

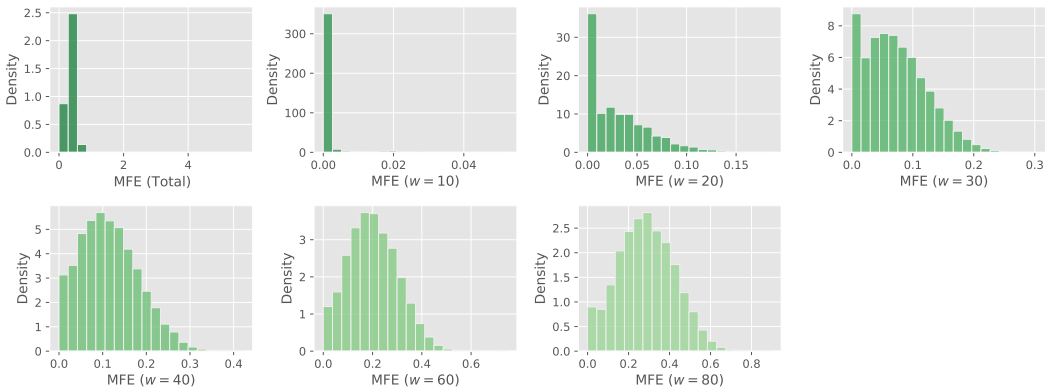


Figure S17: Histograms of ΔG across 5'-UTR for varying window sizes.

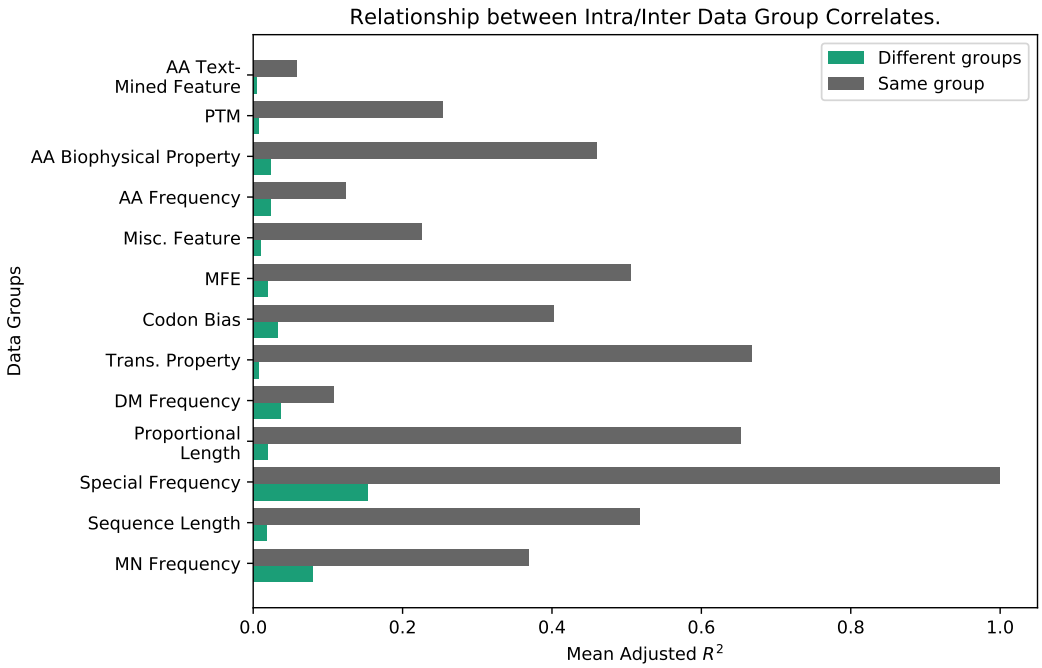


Figure S18: *Calculations of the mean Adjusted r^2 for each SDF feature by data group for the same group $r(x, y)$ where $\forall x, y$ is in the same group and $r(x, z)$ where $\forall z$ is in a different group to $\forall x$.*

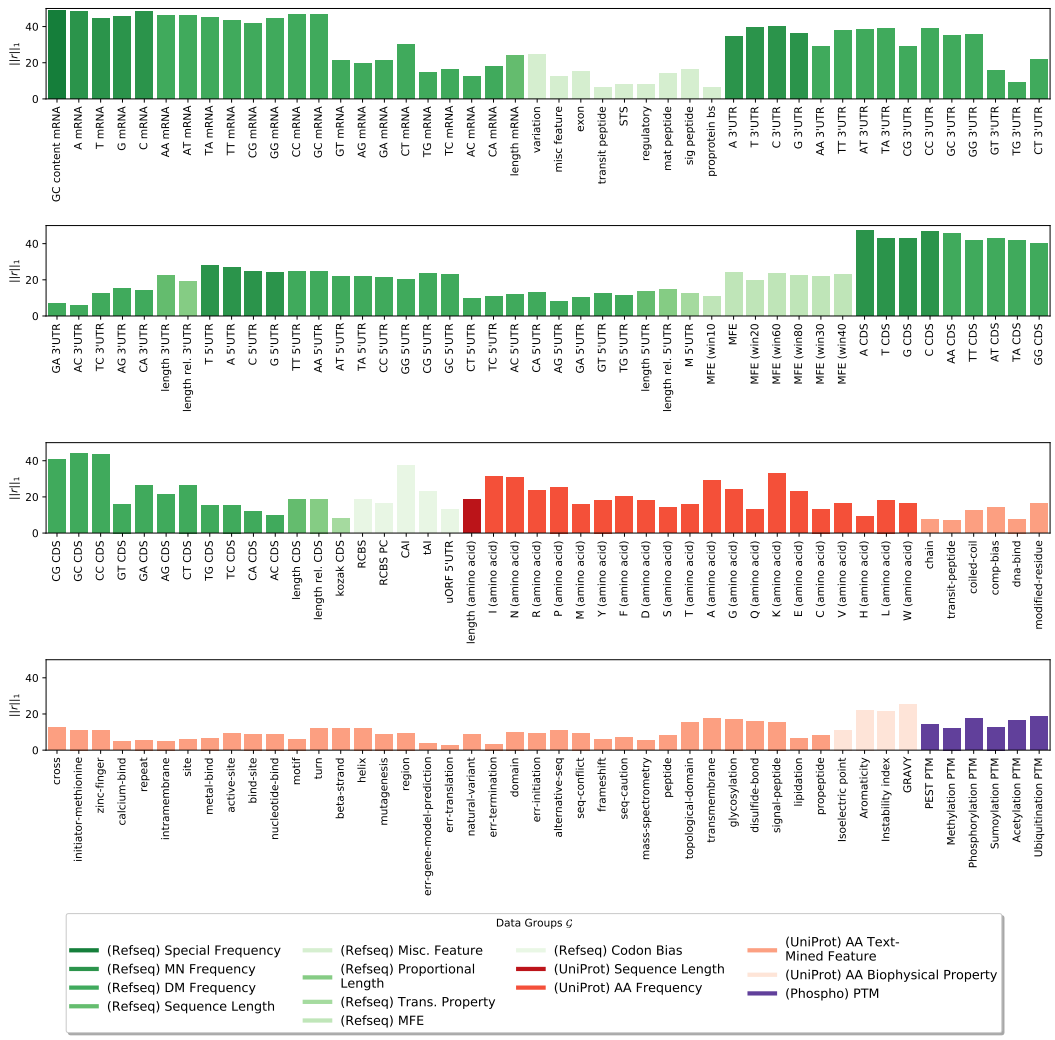


Figure S19: Correlation magnitudes across the pairwise SDF correlation matrix. *Barplots ℓ_1-norm magnitudes for each SDF feature coloured by data grouping $\mathcal{G}$. Greens represent mRNA features, orange/red proteins and purples are PTMs. All plots scaled to same axis.*

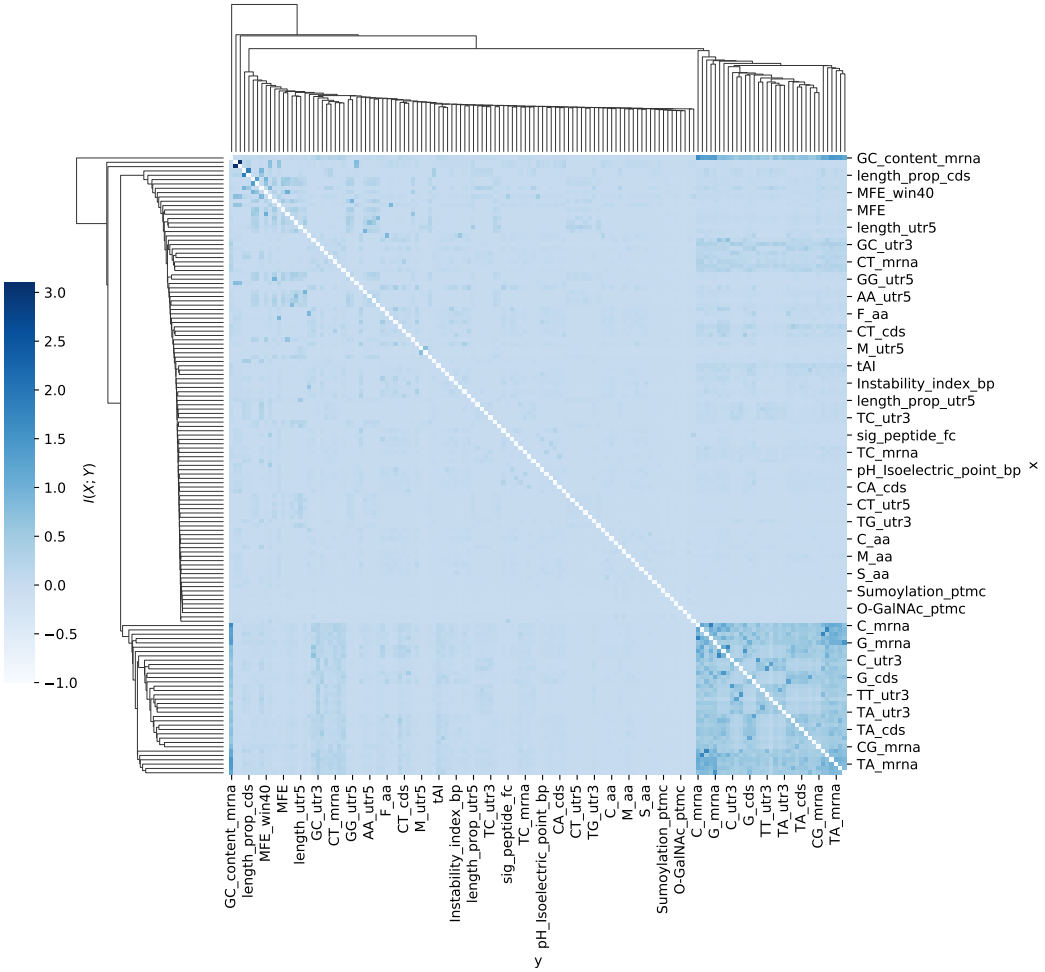


Figure S20: Hierarchical clustering of Mutual Information (MI) scores for pairwise SDF features. *Pairwise MI scores with forced $I(X;X) = -1$ on the diagonal, across all continuous $X, Y \in \mathbb{R}^n$ SDF features. Agglomerative clustering on correlations using average linkage.*

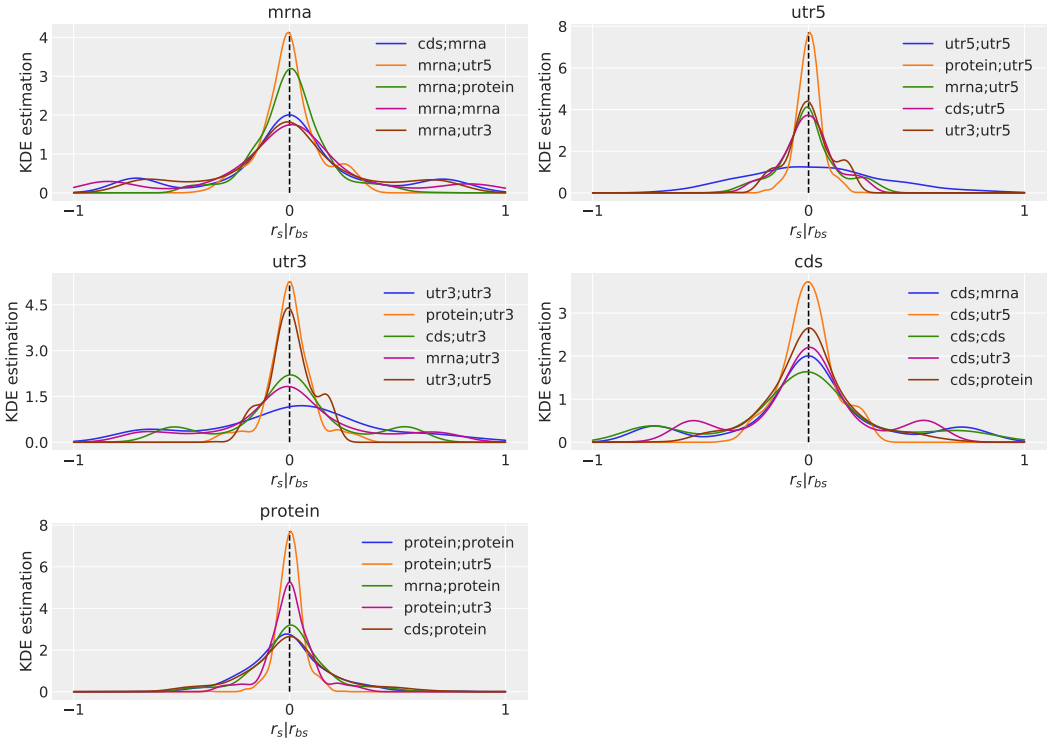


Figure S21: Kernel density estimations (KDE) of pairwise SDF correlations by gene region. *Gaussian non-parametric KDE estimates of pairwise Spearman-rank (r_s)/Biserial (r_{bs}) correlations filtered by gene regions mRNA/CDS/5'-UTR/3'-UTR or Protein/Amino acid. Duplicate KDEs do exist across the plots, with each figure sorted if the SDF data region is in at least one of the pairs. For example a 'mrna;mrna' pair would contain two features derived from the mRNA sequence, and so on.*

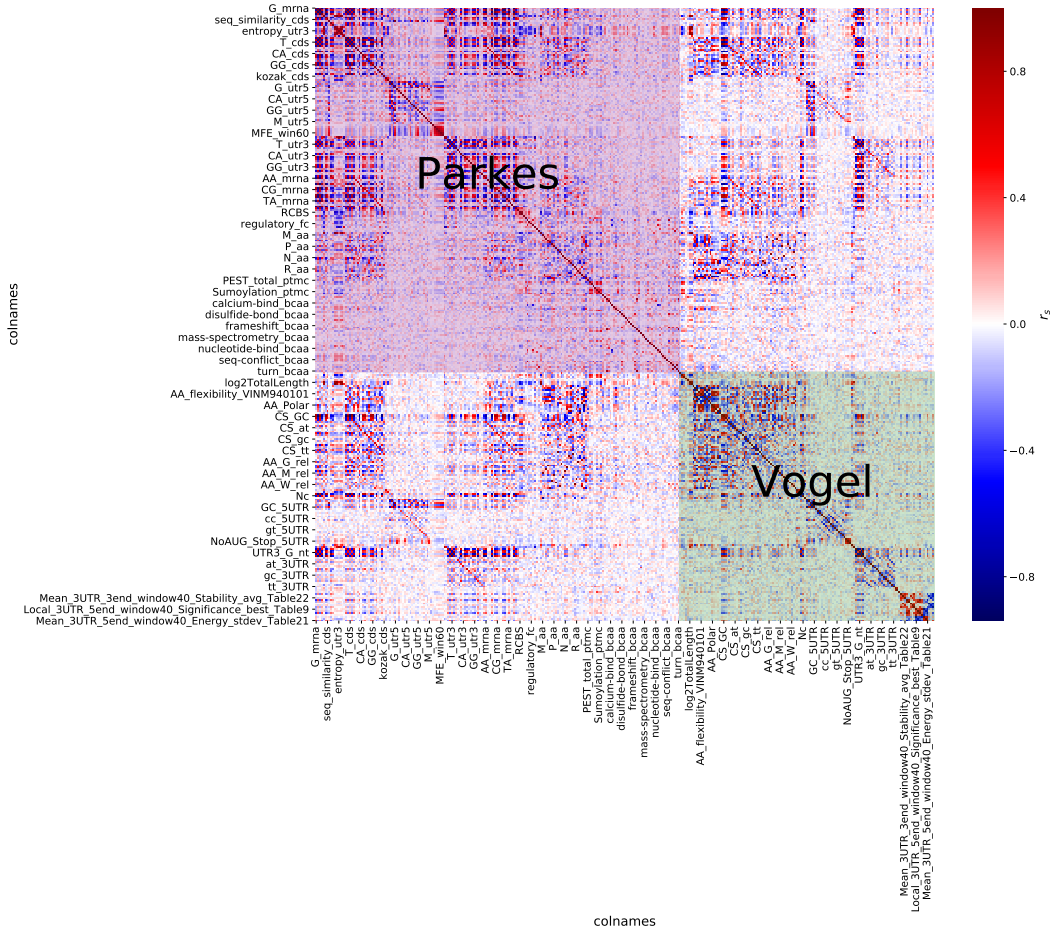


Figure S22: CCA analysis on Parkes and Vogel data matrices.

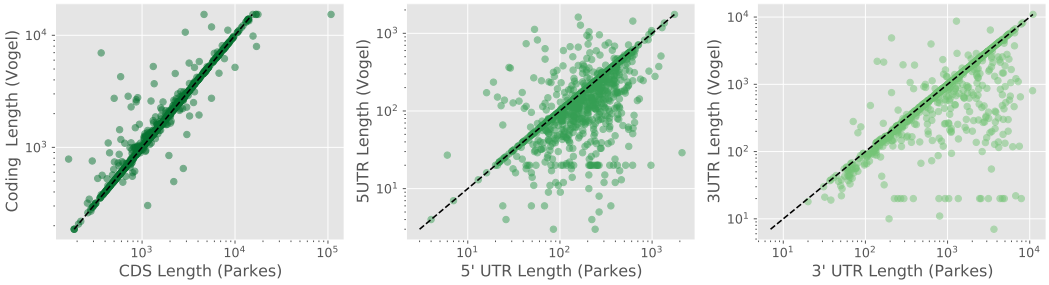


Figure S23: Scatterplots of Sequence Lengths between Parkes and Vogel sub-sets. *Log-log Scatterplots of CDS (n =), 5'-UTR and 3'-UTR sequence lengths, with black line as $y = x$.*

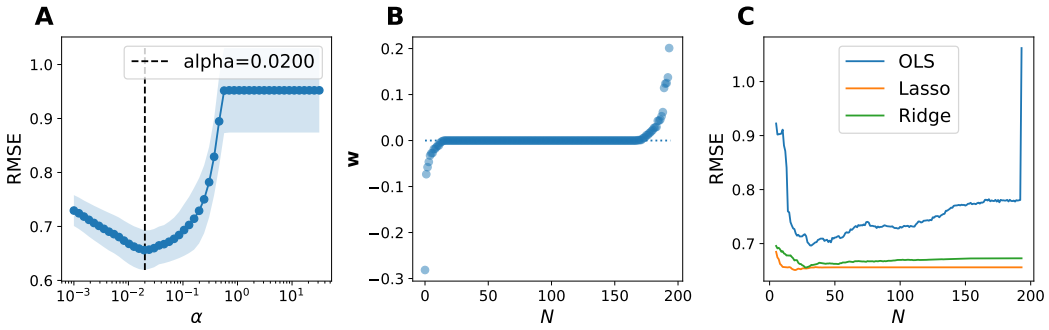


Figure S24: Feature Selection with SDF on Vogel's expression set. *A) Hyperparameter selection of α for LASSO model using SDF as input* **X** *with protein expression from Vogel et al [6] as target. B) Coefficients from subsequent fit model using best α. C) Recursive feature elimination (RFE) scores using 3 different models as N increases.*

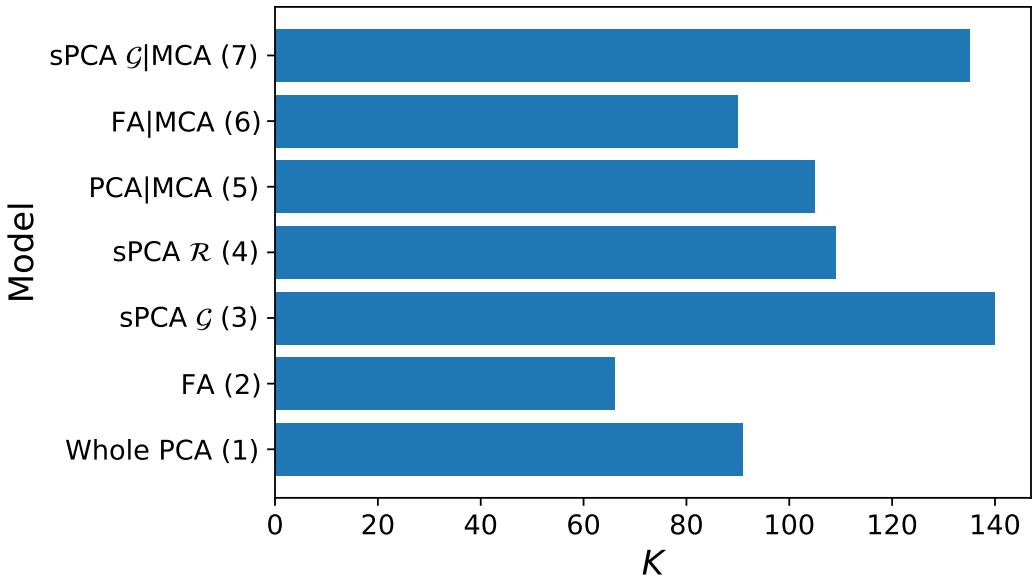


Figure S25: Comparison between dimensionality reduction methods on K. *Default $K > 200$ prior to reduction.*

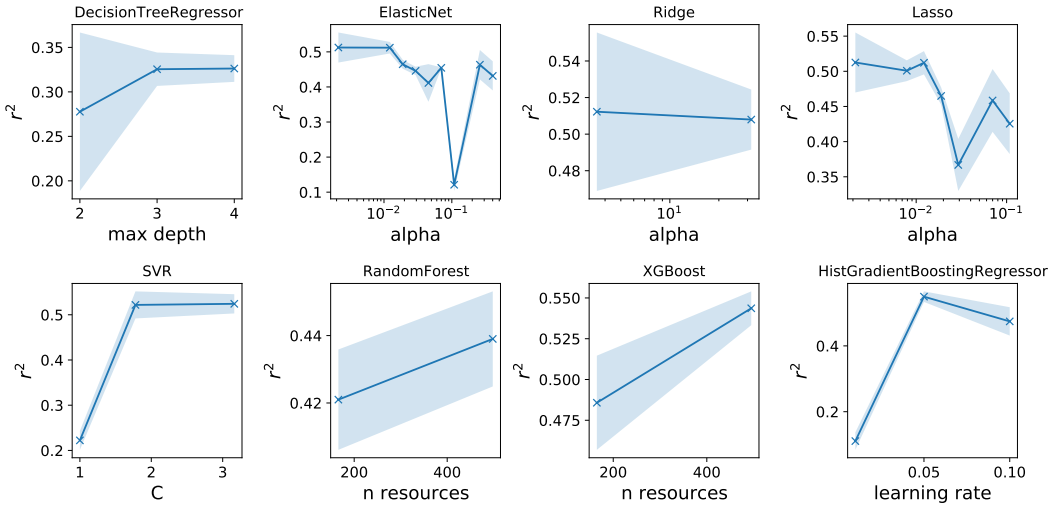


Figure S26: Parameter tuning of SDF-protein model selection. *Different models with key parameters are shown as they tune with respect to adjusted r^2. Here n_resources refers to the number of trees/estimators. Additional points were sampled but may not be displayed due to producing spurious results.*

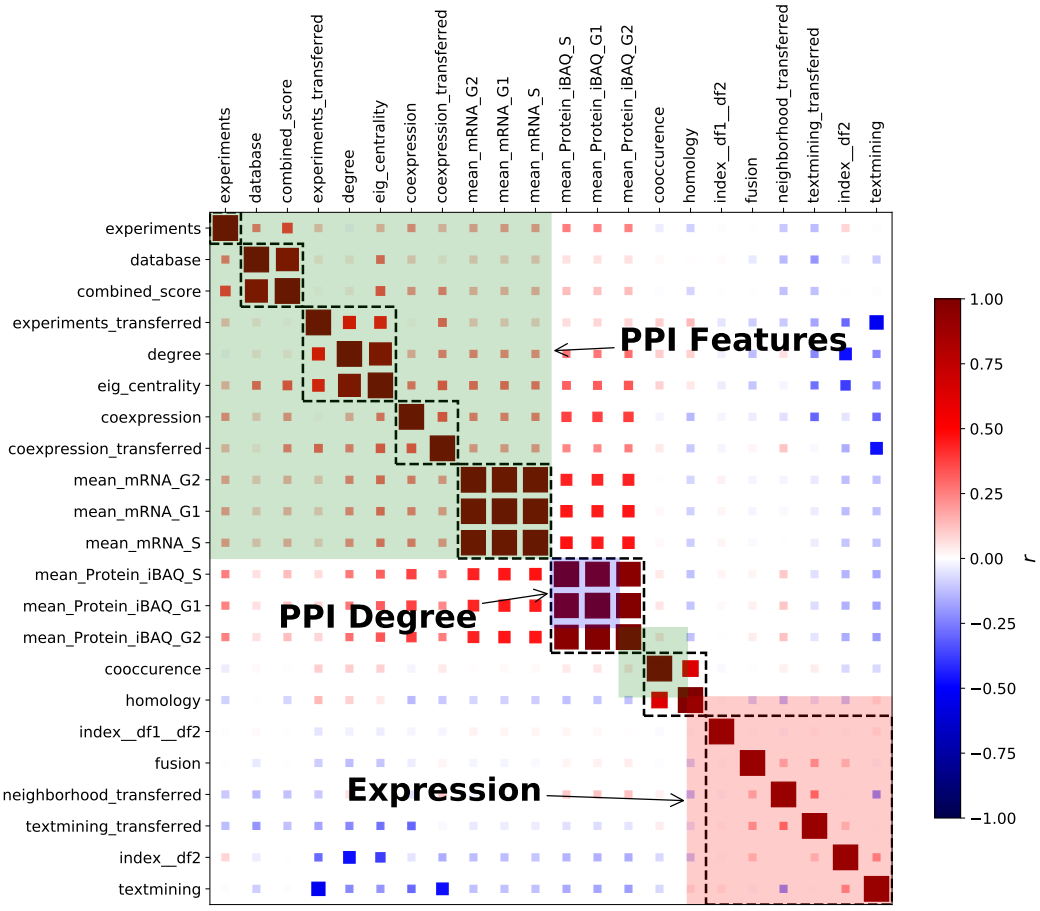


Figure S27: Correlations between PPI features and expression levels. *Hierarchical clustering of Pearson correlations between expression features (red region) and SDF features by network degree (blue) and other scores (PPI features). Black boxes indicate preferred feature groupings as determined by single linkage.*

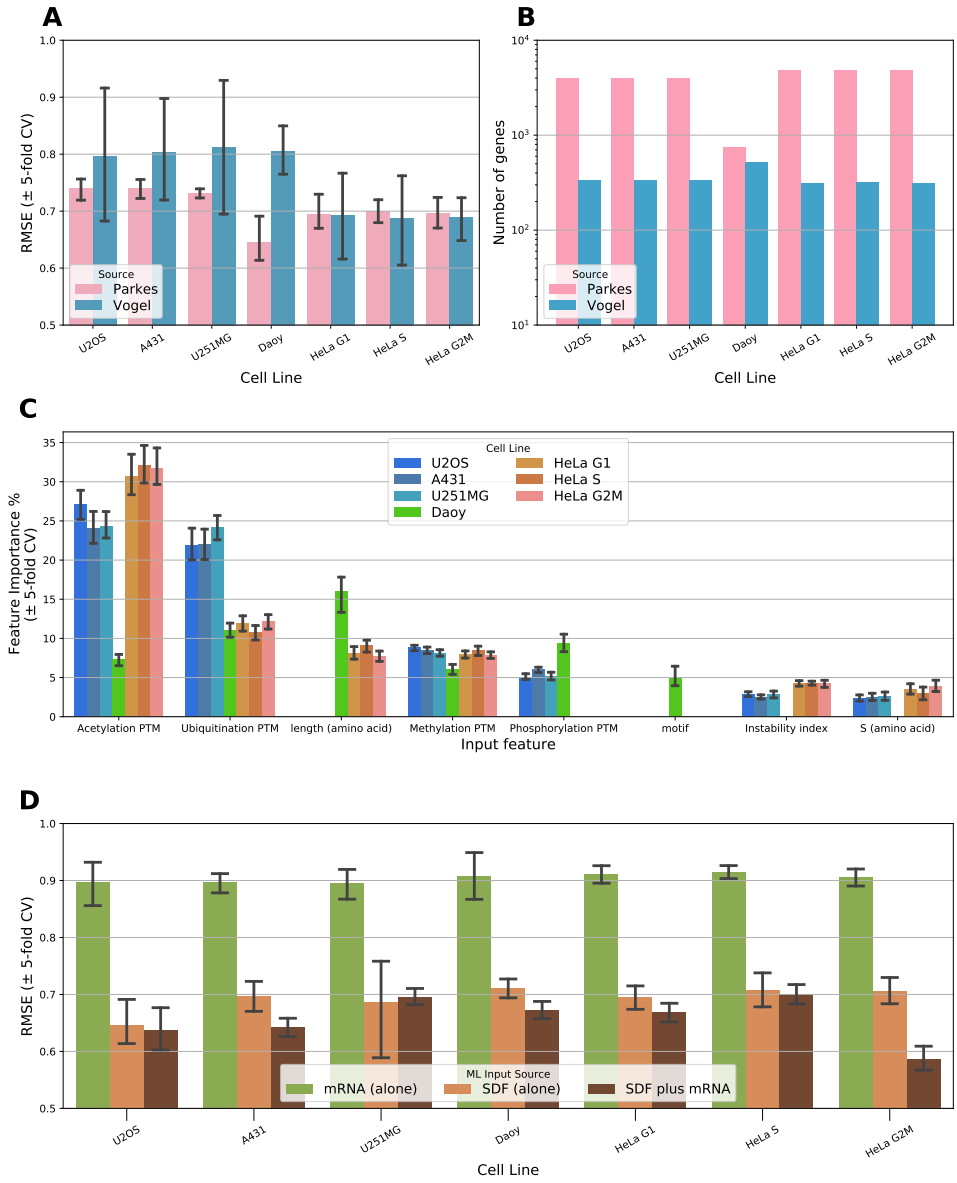


Figure S28: Substantial improvement in SDF inference capacity with respect to protein concentration to mRNA *(A) Whole genome RMSE test-error (± 5-fold CV) from XGBoost models across different cell lines for whole input sources, (B) Sample sizes for each resultant cell line ML model by input source, (C) Feature weights as a relative importance by the 6 most important features by cell line, (D) RMSE test-error (± 5-fold CV) by cell line depending on input features (using mRNA expression, SDFs, or both). See Supplementary 6.2 for model and feature preprocessing/selection.*

Abbreviations

Broken down in this paper into biological and computational abbreviations as follows:

Biological

SDF Sequence-derived feature

DDR DNA Damage Response

dNTP Dye-labelled normal deoxynucleotides

cDNA Complementary DNA

RISC RNA-induced silencing complex

SNP Single-nucleotide polymorphism

PCR Polymerase Chain Reaction

qPCR Quantitative PCR

emPCR Emulsion PCR

GWAS Genome-wide Association Study

CNV Copy number variants

DNA Deoxyribonucleic acid

HGP Human Genome Project

RNA Ribonucleic acid

rRNA Ribosomal RNA

NGS Next-generation sequencing

ATAC-seq Assay for transposase-accessible chromatin using sequencing

PUNCH-P PUromycin-associated Nascent CHain Proteomics

tAI tRNA Adaptation Index

CAI Codon Adaptation Index

ER Evolutionary Rate

RCB Relative Codon Bias

GRAVY Grand Average of Hydropathy

PTM Post-translational modification

PTR Post-translationally regulated

2D-E 2D Gel Electrophoresis

DIGE Flourescence 2D Differential Gel Electrophoresis

ICTA Isotope-Coded Affinity Tag

iTRAQ Isobaric Tag for Relative and Absolute Quantification

Computational

ML Machine Learning/Maximum Likelihood

EM Expectation-Maximization

LM Linear Model

DT Decision Tree

LRM Linear Regression Model

GLM Generalized Linear Model

LMM Linear Mixed Model

OLS Ordinary Least Squares

GBRT Gradient-boosted regression tree

GFFS Greedy forward feature selection

CART Classification and regression tree

RFE Recursive Feature Elimination

DOF Degrees of Freedom

MSE Mean-squared error

SILAC Stable Isotope labelling with Amino Acids in Cell Culture

MS Mass Spectrometry

SLiM Short Linear Motif

ELM Eukaryotic Linear Motif

AA Amino Acid

RMSE Root mean squared error

RSS Residual Sum of squares

EVR Explained variance ratio

SD Standard deviation

Var Variance

MLE Maximum Likelihood Estimate/Estimation

MVN Multivariate Normal

MCA Multiple correspondence analysis

i.i.d Independent and identically distributed

SGD Stochastic Gradient Descent

MCMC Markov Chain Monte Carlo

NLL (Negative) Log-Likelihood

NN Neural Network

MLP Multilayer perceptron

PCA Principle component analysis

PPCA Probabilistic principle component analysis

sPCA Stratified PCA

SVM Support Vector Machine

RF Random Forest

DL Deep Learning

Bibliography

[1] Parkes, G. M., & Niranjan, M. (2019). Uncovering extensive post-translation regulation during human cell cycle progression by integrative multi-'omics analysis. BMC Bioinformatics. https://doi.org/10.1186/s12859-019-3150-5

[2] Soboleva, A., Schmidt, R., Vikhnina, M., Grishina, T. and Frolov, A. (2017). Maillard Proteomics: Opening New Pages. International Journal of Molecular Sciences, 18(12), p.2677.

[3] Gunawardana, Y. and Niranjan, M. (2013). Bridging the gap between transcriptome and proteome measurements identifies post-translationally regulated genes. Bioinformatics, 29(23), pp.3060-3066.

[4] Aviner, R., Geiger, T. and Elroy-Stein, O. (2013). Novel proteomic approach (PUNCH-P) reveals cell cycle-specific fluctuations in mRNA translation. Genes & Development, 27(16), pp.1834-1844.

[5] Aviner, R., Shenoy, A., Elroy-Stein, O. and Geiger, T. (2015). Uncovering Hidden Layers of Cell Cycle Regulation through Integrative Multi-omic Analysis. PLOS Genetics, 11(10), p.e1005554.

[6] Vogel, C., de Sousa Abreu, R., Ko, D., Le, S., Shapiro, B., Burns, S., Sandhu, D., Boutz, D., Marcotte, E. and Penalva, L. (2010). Sequence signatures and mRNA concentration can explain two-thirds of protein abundance variation in a human cell line. Molecular Systems Biology, 6.

[7] Sharma, S., Singh, G., & Singh, R. (2019). Human protein function prediction enhancement using decision tree based machine learning approach. Communications in Computer and Information Science, 1025 CCIS. https://doi.org/10.1007/978-981-15-1384-8_23

[8] Singh, M., Singh, G., & Sharma, S. (2012). Human Protein Function Prediction from Sequence Derived Features using See5. International Journal of Scientific Research, 3(7).

[9] Watson, J. and Crick, F. (1969). Molecular Structure of Nucleic Acids: A Structure for Deoxyribose Nucleic Acid (Reprinted from Nature, April 25, 1953). Nature, 224(5218), pp.470-471.

[10] Crick, F. (1970). Central Dogma of Molecular Biology. Nature, 227(5258), pp.561-563.

[11] Sanger, F., & Coulson, A. R. (1975). A rapid method for determining sequences in DNA by primed synthesis with DNA polymerase. Journal of Molecular Biology. https://doi.org/10.1016/0022-2836(75)90213-2

[12] Sanger, F., Nicklen, S. and Coulson, A. (1977). DNA sequencing with chain-terminating inhibitors. Proceedings of the National Academy of Sciences, 74(12), pp.5463-5467.

[13] Church, G. M., Gao, Y., & Kosuri, S. (2012). Next-generation digital information storage in DNA. In Science. https://doi.org/10.1126/science.1226355

[14] Mattick, J. (2001). Non-coding RNAs: the architects of eukaryotic complexity. EMBO reports, 2(11), pp.986-991.

[15] Lackner, D.H. and Bähler, J.(2008). Chapter 5 Translational Control of Gene Expression: From Transcripts to Transcriptomes. International Review of Cell and Molecular Biology, Academic Press, 271, pp.199-251.

[16] Wu, J., Xiao, J., Zhang, Z., Wang, X., Hu, S. and Yu, J. (2014). Ribogenomics: the Science and Knowledge of RNA. Genomics, Proteomics & Bioinformatics, 12(2), pp.57-63

[17] Stults, D. M., Killen, M. W., Williamson, E. P., Hourigan, J. S., Vargas, H. D., Arnold, S. M., Moscow, J. A., & Pierce, A. J. (2009). Human rRNA gene clusters are recombinational hotspots in cancer. Cancer Research. https://doi.org/10.1158/0008-5472.CAN-09-2680

[18] Woodson, S. A. (2011). RNA folding pathways and the self-assembly of ribosomes. Accounts of Chemical Research. https://doi.org/10.1021/ar2000474

[19] Chan, J. C., Hannan, K. M., Riddell, K., Ng, P. Y., Peck, A., Lee, R. S., Hung, S., Astle, M. V., Bywater, M., Wall, M., Poortinga, G., Jastrzebski, K., Sheppard, K. E., Hemmings, B. A., Hall, M. N., Johnstone, R. W., McArthur, G. A., Hannan, R. D., & Pearson, R. B. (2011). AKT promotes rRNA synthesis and cooperates with c-MYC to stimulate ribosome biogenesis in cancer. Science Signaling. https://doi.org/10.1126/scisignal.2001754

[20] Li, S., Ibaragi, S., & Hu, G. F. (2011). Angiogenin as a molecular target for the treatment of prostate cancer. Current cancer therapy reviews, 7(2), 83–90. https://doi.org/10.2174/1573394711107020083

[21] Larson, K., Yan, S. J., Tsurumi, A., Liu, J., Zhou, J., Gaur, K., Guo, D., Eickbush, T. H., & Li, W. X. (2012). Heterochromatin formation promotes longevity and represses ribosomal RNA synthesis. PLoS Genetics. https://doi.org/10.1371/journal.pgen.1002473

[22] Lu, J., & Salzberg, S. L. (2020). Ultrafast and accurate 16S rRNA microbial community analysis using Kraken 2. Microbiome. https://doi.org/10.1186/s40168-020-00900-2

[23] Bartel, D.P. (2004). MicroRNAs: genomics, biogenesis, mechanism, and function. CELL. 116, pp. 281–297.

[24] Lee, R. C., Feinbaum, R. L., & Ambros, V. (1993). The C. elegans heterochronic gene lin-4 encodes small RNAs with antisense complementarity to lin-14. Cell. https://doi.org/10.1016/0092-8674(93)90529-Y

[25] Reinhart, B. J., Slack, F. J., Basson, M., Pasquinelll, A. E., Bettlnger, J. C., Rougvle, A. E., Horvitz, H. R., & Ruvkun, G. (2000). The 21-nucleotide let-7 RNA regulates developmental timing in Caenorhabditis elegans. Nature. https://doi.org/10.1038/35002607

[26] Lee, R. C., & Ambros, V. (2001). An extensive class of small RNAs in Caenorhabditis elegans. Science. https://doi.org/10.1126/science.1065329

[27] Bartel, D.P. (2009). MicroRNAs: Target Recognition and Regulatory Functions. Cell, 136(2), pp.215-233.

[28] Iwakawa, H. & Tomari, Y. (2015). The Functions of MicroRNAs: mRNA Decay and Translational Repression. Trends in Cell Biology. 25, pp. 651–665.

[29] Calin, G. A., Dumitru, C. D., Shimizu, M., Bichi, R., Zupo, S., Noch, E., Aldler, H., Rattan, S., Keating, M., Rai, K., Rassenti, L., Kipps, T., Negrini, M., Bullrich, F., & Croce, C. M. (2002). Frequent deletions and down-regulation of micro-RNA genes miR15 and miR16 at 13q14 in chronic lymphocytic leukemia. Proceedings of the National Academy of Sciences of the United States of America. https://doi.org/10.1073/pnas.242606799

[30] Kim, S., Rhee, J. keun, Yoo, H. J., Lee, H. J., Lee, E. J., Lee, J. W., Yu, J. H., Son, B. H., Gong, G., Kim, S. B., Singh, S. R., Ahn, S. H., & Chang, S. (2015). Bioinformatic and metabolomic analysis reveals miR-155 regulates thiamine level in breast cancer. Cancer Letters. https://doi.org/10.1016/j.canlet.2014.11.058

[31] Esteller, M. (2011). Non-coding RNAs in human disease. Nature Reviews Genetics 12, pp. 861-864.

[32] Hou, Y., Wang, J., Wang, X., Shi, S., Wang, W., & Chen, Z. (2016). Appraising MicroRNA-155 as a noninvasive diagnostic biomarker for cancer detection: A meta-analysis. Medicine 95: e2450.

[33] Kent, S. (2009). Total chemical synthesis of proteins. Chem. Soc. Rev., 38(2), pp.338-351.

[34] Xu, D., & Nussinov, R. (1998). Favorable domain size in proteins. Folding and Design. https://doi.org/10.1016/S1359-0278(98)00004-2

[35] Diella, F., Haslam, N., Chica, C., Budd, A., Michael, S., Brown, N. P., Travé, G., & Gibson, T. J. (2008). Understanding eukaryotic linear motifs and their role in cell signaling and regulation. Frontiers in Bioscience. https://doi.org/10.2741/3175

[36] Ren, S., Uversky, V. N., Chen, Z., Dunker, A. K., & Obradovic, Z. (2008). Short Linear Motifs recognized by SH2, SH3 and Ser/Thr Kinase domains are conserved in disordered protein regions. BMC genomics, 9 Suppl 2(Suppl 2), S26. https://doi.org/10.1186/1471-2164-9-S2-S26

[37] Dinkel, H., Michael, S., Weatheritt, R. J., Davey, N. E., Van Roey, K., Altenberg, B., Toedt, G., Uyar, B., Seiler, M., Budd, A., Jödicke, L., Dammert, M. A., Schroeter, C., Hammer, M., Schmidt, T., Jehl, P., McGuigan, C., Dymecka, M., Chica, C., ... Gibson, T. J. (2012). ELM - The database of eukaryotic linear motifs. Nucleic Acids Research. https://doi.org/10.1093/nar/gkr1064

[38] Edwards, J., Yarychkivska, O., Boulard, M. and Bestor, T. (2017). DNA methylation and DNA methyltransferases. Epigenetics & Chromatin, 10(1).

[39] Edwards, J., O'Donnell, A., Rollins, R., Peckham, H., Lee, C., Milekic, M., Chanrion, B., Fu, Y., Su, T., Hibshoosh, H., Gingrich, J., Haghighi, F., Nutter, R. and Bestor, T. (2010). Chromatin and sequence features that define the fine and gross structure of genomic methylation patterns. Genome Research, 20(7), pp.972-980.

[40] Ehrlich, M. (2009). DNA hypomethylation in cancer cells. In Epigenomics. https://doi.org/10.2217/EPI.09.33

[41] Klutstein, M., Nejman, D., Greenfield, R. and Cedar, H. (2016). DNA Methylation in Cancer and Aging. Cancer Research, 76(12), pp.3446-3450.

[42] Mikeska, T., Bock, C., Do, H., and Dobrovic, A. (2012). DNA methylation biomarkers in cancer: Progress towards clinical implementation. In Expert Review of Molecular Diagnostics. https://doi.org/10.1586/erm.12.45

[43] Gonzalo, S. (2010). Epigenetic alterations in aging. In Journal of Applied Physiology. https://doi.org/10.1152/japplphysiol.00238.2010

[44] Watson, J.D., Baker, T.A., Bell, S.P., Gann, A.A., Levine, M., Losick and R.M. (2013). Molecular Biology of the Gene (7th ed.). Pearson.

[45] Proudfoot, N., Furger, A. and Dye, M. (2002). Integrating mRNA Processing with Transcription. Cell, 108(4), pp.501-512.

[46] Temperley, R., Wydro, M., Lightowlers, R. and Chrzanowska-Lightowlers, Z. (2010). Human mitochondrial mRNAs—like members of

all families, similar but different. Biochimica et Biophysica Acta (BBA) - Bioenergetics, 1797(6-7), pp.1081-1085.

[47] Monde, R., Schuster, G. and Stern, D. (2000). Processing and degradation of chloroplast mRNA. Biochimie, 82(6-7), pp.573-582.

[48] Will, C. and Luhrmann, R. (2010). Spliceosome Structure and Function. Cold Spring Harbor Perspectives in Biology, 3(7), pp.a003707-a003707.

[49] Black, D. (2003). Mechanisms of Alternative Pre-Messenger RNA Splicing. Annual Review of Biochemistry, 72(1), pp.291-336.

[50] Catalanotto, C., Cogoni, C., & Zardo, G. (2016). MicroRNA in control of gene expression: An overview of nuclear functions. In International Journal of Molecular Sciences. https://doi.org/10.3390/ijms17101712

[51] Orlowski, M. (1981). Growth-rate-dependent adjustment of ribosome function in the fungus Mucor racemosus. Biochemical Journal, 196(2), pp.403-410.

[52] Hershey, J. W. B., Sonenberg, N., & Mathews, M. B. (2012). Principles of translational control: An overview. Cold Spring Harbor Perspectives in Biology. https://doi.org/10.1101/cshperspect.a009829

[53] Tuller, T., Waldman, Y., Kupiec, M. and Ruppin, E. (2010). Translation Efficiency Is Determined By Both Codon Bias And Folding Energy. Proceedings of the National Academy of Sciences 107.8, pp.3645-3650.

[54] Mann, M. and Jensen, O. (2003). Proteomic analysis of post-translational modifications. Nature Biotechnology, 21(3), pp.255-261.

[55] Nguyen, L., Kolch, W. and Kholodenko, B. (2013). When ubiquitination meets phosphorylation: a systems biology perspective of EGFR/MAPK signalling. Cell Communication and Signaling, 11(1), p.52.

[56] Swaney, D., Beltrao, P., Starita, L., Guo, A., Rush, J., Fields, S., Krogan, N. and Villén, J. (2013). Global analysis of phosphorylation and ubiquitylation cross-talk in protein degradation. Nature Methods, 10(7), pp.676-682.

[57] Walsh, C. and Walsh, C. (2006). Posttranslational modifications of proteins. Englewood, Colo: Roberts and Company Publishers.

[58] Pickart, C. and Eddins, M. (2004). Ubiquitin: structures, functions, mechanisms. Biochimica et Biophysica Acta (BBA) - Molecular Cell Research, 1695(1-3), pp.55-72.

[59] Hershberg, R., & Petrov, D. A. (2008). Selection on codon bias. In Annual Review of Genetics. https://doi.org/10.1146/annurev.genet.42.110807.091442

[60] Sharp, P. M., & Li, W. H. (1987). The codon adaptation index-a measure of directional synonymous codon usage bias, and its potential applications. Nucleic Acids Research. https://doi.org/10.1093/nar/15.3.1281

[61] Roymondal, U., Shibsankar, D., and Satyabrata, S. (2009). Predicting Gene Expression Level from Relative Codon Usage Bias: An Application to Escherichia Coli Genome. DNA Research: An International Journal for Rapid Publication of Reports on Genes and Genomes 16.1 13–30.

[62] Sabi, R., Volvovitch Daniel, R. and Tuller, T. (2016). stAI calc: tRNA adaptation index calculator based on species-specific weights. Bioinformatics, pp.647.

[63] Chan, P.P. and Lowe, T.M. (2016). GtRNAdb 2.0: an expanded database of transfer RNA genes identified in complete and draft genomes. Nucl. Acids Res. 44. :D184-D189.

[64] Bryant, J. and Francis, D. (2008). The eukaryotic cell cycle. New York: Taylor & Francis.

[65] Kastan, M. and Bartek, J. (2004). Cell-cycle checkpoints and cancer. Nature, 432(7015), pp.316-323.

[66] Bertoli, C., Skotheim, J. and de Bruin, R. (2013). Control of cell cycle transcription during G1 and S phases. Nature Reviews Molecular Cell Biology, 14(8), pp.518-528.

[67] Campisi, J., Kim, S., Lim, C. and Rubio, M. (2001). Cellular senescence, cancer and aging: the telomere connection. Experimental Gerontology, 36(10), pp.1619-1637.

[68] Visconti, R., Della Monica, R. and Grieco, D. (2016). Cell cycle checkpoint in cancer: a therapeutically targetable double-edged sword. Journal of Experimental & Clinical Cancer Research, 35(1).

[69] Parker, L.L. and Piwnica-Worms, H. (1992). Inactivation of the p34cdc2-cyclin B complex by the human WEE1 tyrosine kinase. Science. 257:1955–7.

[70] Tyson, J. J. (1975). On the existence of oscillatory solutions in negative feedback cellular control processes. Journal of Mathematical Biology. https://doi.org/10.1007/BF00279849

[71] Novak, B., & Tyson, J. J. (1993). Modeling the cell division cycle: M-phase trigger, oscillations, and size control. Journal of Theoretical Biology. https://doi.org/10.1006/jtbi.1993.1179

[72] Ferrell, J. E., Tsai, T. Y. C., & Yang, Q. (2011). Modeling the cell cycle: Why do certain circuits oscillate? In Cell. https://doi.org/10.1016/j.cell.2011.03.006

[73] Alfieri, R., Barberis, M., Chiaradonna, F., Gaglio, D., Milanesi, L., Vanoni, M., Klipp, E., & Alberghina, L. (2009). Towards a systems biology approach to mammalian cell cycle: Modeling the entrance into S phase of quiescent fibroblasts after serum stimulation. BMC Bioinformatics. https://doi.org/10.1186/1471-2105-10-S12-S16

[74] Lander, E. S., Linton, L. M., Birren, B., Nusbaum, C., Zody, M. C., Baldwin, J., Devon, K., Dewar, K., Doyle, M., Fitzhugh, W., Funke, R., Gage, D., Harris, K., Heaford, A., Howland, J., Kann, L., Lehoczky, J., Levine, R., McEwan, P., ... Morgan, M. J. (2001). Initial sequencing and analysis of the human genome. Nature. https://doi.org/10.1038/35057062

[75] Craig Venter, J., Adams, M. D., Myers, E. W., Li, P. W., Mural, R. J., Sutton, G. G., Smith, H. O., Yandell, M., Evans, C. A., Holt, R. A., Gocayne, J. D., Amanatides, P., Ballew, R. M., Huson, D. H., Wortman, J. R., Zhang, Q., Kodira, C. D., Zheng, X. H., Chen, L., ... Zhu, X. (2001). The sequence of the human genome. Science. https://doi.org/10.1126/science.1058040

[76] Mardis, E. R. (2013). Next-generation sequencing platforms. Annual Review of Analytical Chemistry. https://doi.org/10.1146/annurev-anchem-062012-092628

[77] Levy, S. E., & Myers, R. M. (2016). Advancements in Next-Generation Sequencing. In Annual Review of Genomics and Human Genetics. https://doi.org/10.1146/annurev-genom-083115-022413

[78] Altshuler, D. M., Durbin, R. M., Abecasis, G. R., Bentley, D. R., Chakravarti, A., Clark, A. G., Donnelly, P., Eichler, E. E., Flicek, P., Gabriel, S. B., Gibbs, R. A., Green, E. D., Hurles, M. E., Knoppers, B. M., Korbel, J. O., Lander, E. S., Lee, C., Lehrach, H., Mardis, E. R., ... Lacroute, P. (2012). An integrated map of genetic variation from 1,092 human genomes. Nature. https://doi.org/10.1038/nature11632

[79] Fu, W., O'Connor, T. D., Jun, G., Kang, H. M., Abecasis, G., Leal, S. M., Gabriel, S., Altshuler, D., Shendure, J., Nickerson, D. A., Bamshad, M. J., & Akey, J. M. (2013). Analysis of 6,515 exomes reveals the recent origin of most human protein-coding variants. Nature. https://doi.org/10.1038/nature11690

[80] Maxam, A. M., & Gilbert, W. (1977). A new method for sequencing DNA. Proceedings of the National Academy of Sciences of the United States of America. https://doi.org/10.1073/pnas.74.2.560

[81] Scharf, S. J., Horn, G. T., & Erlich, H. A. (1986). Direct cloning and sequence analysis of enzymatically amplified genomic sequences. Science. https://doi.org/10.1126/science.3461561

[82] Brock, T. D., & Freeze, H. (1969). Thermus aquaticus gen. n. and sp. n., a nonsporulating extreme thermophile. Journal of Bacteriology. https://doi.org/10.1128/jb.98.1.289-297.1969

[83] Nolan, T., Hands, R. E., & Bustin, S. A. (2006). Quantification of mRNA using real-time RT-PCR. Nature Protocols. https://doi.org/10.1038/nprot.2006.236

[84] Holland, P. M., Abramson, R. D., Watson, R., & Gelfand, D. H. (1991). Detection of specific polymerase chain reaction product by utilizing the $5' \rightarrow 3'$ exonuclease activity of Thermus aquaticus DNA polymerase. Proceedings of the National Academy of Sciences of the United States of America. https://doi.org/10.1073/pnas.88.16.7276

[85] S. Goodwin, J. D. McPherson, and W. R. McCombie, "Coming of age: Ten years of next-generation sequencing technologies," Nature Reviews Genetics. 2016, doi: 10.1038/nrg.2016.49.

[86] Augenlicht, L. H., and Kobrin, D. (1982). Cloning and Screening of Sequences Expressed in a Mouse Colon Tumor. Cancer Research. PMID:7059971

[87] Dandy, D. S., Wu, P., and Grainger, D. W. (2007). Array feature size influences nucleic acid surface capture in DNA microarrays. Proceedings of the National Academy of Sciences of the United States of America. https://doi.org/10.1073/pnas.0606054104

[88] Carter, N. P. (2007). Methods and strategies for analyzing copy number variation using DNA microarrays. Nature Genetics. https://doi.org/10.1038/ng2028

[89] Landegren, U., Kaiser, R., Sanders, J., & Hood, L. (1988). A ligase-mediated gene detection technique. Science. https://doi.org/10.1126/science.3413476

[90] Ju, J., Kim, D. H., Bi, L., Meng, Q., Bai, X., Li, Z., Li, X., Marma, M. S., Shi, S., Wu, J., Edwards, J. R., Romu, A., & Turro, N. J. (2006). Four-color DNA sequencing by synthesis using cleavable fluorescent nucleotide reversible terminators. Proceedings of the National Academy of Sciences of the United States of America. https://doi.org/10.1073/pnas.0609513103

[91] Mirkin, S. M. (2007). Expandable DNA repeats and human disease. In Nature. https://doi.org/10.1038/nature05977

[92] Chandramouli, K., & Qian, P.Y. (2009). Proteomics: Challenges, Techniques and Possibilities to Overcome Biological Sample Complexity. Human Genomics and Proteomics. https://doi.org/10.4061/2009/239204

[93] Gunawardana, Y., Fujiwara, S., Takeda, A., Woo, J., Woelk, C. and Niranjan, M. (2015). Outlier detection at the transcriptome-proteome interface. Bioinformatics, 31(15), pp.2530-2536.

[94] Ghaemmaghami, S., Huh, W. K., Bower, K., Howson, R. W., Belle, A., Dephoure, N., O'Shea, E. K., & Weissman, J. S.

(2003). Global analysis of protein expression in yeast. Nature. https://doi.org/10.1038/nature02046

[95] Han, X., Aslanian, A., & Yates, J. R. (2008). Mass spectrometry for proteomics. In Current Opinion in Chemical Biology. https://doi.org/10.1016/j.cbpa.2008.07.024

[96] Yates, J. R., Ruse, C. I., & Nakorchevsky, A. (2009). Proteomics by mass spectrometry: Approaches, advances, and applications. In Annual Review of Biomedical Engineering. https://doi.org/10.1146/annurev-bioeng-061008-124934

[97] Lewis, J. K., Wei, J., & Siuzdak, G. (2006). Matrix-Assisted Laser Desorption/Ionization Mass Spectrometry in Peptide and Protein Analysis. In Encyclopedia of Analytical Chemistry. https://doi.org/10.1002/9780470027318.a1621

[98] Taylor, G. (1964). Disintegration of water drops in an electric field. Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences. https://doi.org/10.1098/rspa.1964.0151

[99] Fournier, M. L., Gilmore, J. M., Martin-Brown, S. A., & Washburn, M. P. (2007). Multidimensional separations-based shotgun proteomics. In Chemical Reviews. https://doi.org/10.1021/cr068279a

[100] Shen, Y., & Smith, R. D. (2002). Proteomics based on high-efficiency capillary separations. In Electrophoresis. https://doi.org/10.1002/1522-2683(200209)23:18¡3106::AID-ELPS3106¿3.0.CO;2-Y

[101] Kenney, J. (1966). Mathematics of statistics. Princeton, NJ: Van Nostrand.

[102] Friedman, J. (1993). Fast MARS. Department of Statistics, Stanford University, Tech. Report LCS110.

[103] Friedman, J. H., & Roosen, C. B. (1995). An introduction to multivariate adaptive regression splines. Statistical Methods in Medical Research, 4(3). https://doi.org/10.1177/096228029500400303

[104] Stigler, S. (1981). Gauss and the Invention of Least Squares. The Annals of Statistics, 9(3), pp.465-474.

[105] Sprent, P. (1966). A Generalized Least-Squares Approach to Linear Functional Relationships. Journal of the Royal Statistical Society: Series B (Methodological), 28(2), pp.278-288.

[106] Swindel, Benee F. (1981). "Geometry of Ridge Regression Illustrated". The American Statistician. 35 (1): 12–15. doi:10.2307/2683577. JSTOR 2683577.

[107] Tibshirani, Robert (1996). "Regression Shrinkage and Selection via the Lasso". Journal of the Royal Statistical Society, Series B. 58 (1): 267–288. JSTOR 2346178.

[108] Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67(2), pp.301-320.

[109] Quinlan, J. (1986). Induction of decision trees. Machine Learning, 1(1), pp.81-106.

[110] Sagi, O. and Rokach, L. (2018). Ensemble learning: A survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8(4), p.e1249.

[111] Friedman, J. (2001). Greedy Function Approximation: A Gradient Boosting Machine. The Annals of Statistics. 29(5).

[112] Friedman, J. (2002). Stochastic gradient boosting. Computational Statistics & Data Analysis, 38(4), pp.367-378.

[113] Ziegel, E. R. (2003). The Elements of Statistical Learning. Technometrics, 45(3), 267–268. https://doi.org/10.1198/tech.2003.s770

[114] Hastie, T.; Tibshirani, R.; Friedman, J. H. (2009). "10. Boosting and Additive Trees". The Elements of Statistical Learning (2nd ed.). New York: Springer. pp. 337–384. ISBN 978-0-387-84857-0.

[115] Rice, John (2007). Mathematical Statistics and Data Analysis. Belmont, CA: Brooks/Cole Cengage Learning. p. 138. ISBN 978-0534-39942-9.

[116] Park, K. (2018). Fundamentals of Probability and Stochastic Processes with Applications to Communications. Cham: Springer International Publishing.

[117] Ledoit, O., & Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. Journal of Multivariate Analysis. https://doi.org/10.1016/S0047-259X(03)00096-4

[118] Yule, G.U and Kendall, M.G. (1950), "An Introduction to the Theory of Statistics", 14th Edition (5th Impression 1968). Charles Griffin & Co. pp 258–270

[119] Baba, K. , Shibata, R. and Sibuya, M. (2004), Partial Correlation and Conditional Correlation as measures of conditional independence. Australian & New Zealand Journal of Statistics, 46: 657-664. doi:10.1111/j.1467-842X.2004.00360.x

[120] Jolliffe, I. T. (1986). Principal Components in Regression Analysis. https://doi.org/10.1007/978-1-4757-1904-8_8

[121] Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. Journal of Educational Psychology, 24(6). https://doi.org/10.1037/h0071325

[122] Bishop, C. M. (2014). Bishop - Pattern Recognition And Machine Learning - Springer 2006. Antimicrobial Agents and Chemotherapy.

[123] Tipping, M. E., & Bishop, C. M. (1999). Probabilistic principal component analysis. Journal of the Royal Statistical Society. Series B: Statistical Methodology, 61(3). https://doi.org/10.1111/1467-9868.00196

[124] Tipping, M. E., & Bishop, C. M. (1999). Mixtures of probabilistic principal component analyzers. Neural Computation, 11(2). https://doi.org/10.1162/089976699300016728

[125] Minka, T. P. (2001). Automatic choice of dimensionality for PCA. Advances in Neural Information Processing Systems.

[126] Futcher, B., Latter, G. I., Monardo, P., McLaughlin, C. S., & Garrels, J. I. (1999). A Sampling of the Yeast Proteome. Molecular and Cellular Biology, 19(11). https://doi.org/10.1128/mcb.19.11.7357

[127] Greenbaum, D., Jansen, R., & Gerstein, M. (2002). Analysis of mRNA expression and protein abundance data: An approach for the comparison of the enrichment of features in the cellular population of proteins and transcripts. Bioinformatics, 18(4). https://doi.org/10.1093/bioinformatics/18.4.585

[128] Greenbaum, D., Colangelo, C., Williams, K., z & Gerstein, M. (2003). Comparing protein abundance and mRNA expression levels on a genomic scale. In Genome Biology (Vol. 4, Issue 9). https://doi.org/10.1186/gb-2003-4-9-117

[129] Beyer, A., Hollunder, J., Nasheuer, H. P. and Wilhelm, T. Post-transcriptional expression regulation in the yeast Saccharomyces cerevisiae on a genomic scale. Mol. Cell. Proteomics , 2004, vol. 3 (pg. 1083-1092)

[130] Wu, G., Nie, L., & Zhang, W. (2008). Integrative analyses of post-transcriptional regulation in the yeast Saccharomyces cerevisiae using transcriptomic and proteomic data. Current Microbiology, 57(1). https://doi.org/10.1007/s00284-008-9145-5

[131] Maier, T., Güell, M., & Serrano, L. (2009). Correlation of mRNA and protein in complex biological samples. In FEBS Letters (Vol. 583, Issue 24). https://doi.org/10.1016/j.febslet.2009.10.036

[132] Ning, K., Fermin, D., & Nesvizhskii, A. I. (2012). Comparative analysis of different label-free mass spectrometry based protein abundance estimates and their correlation with RNA-Seq gene expression data. Journal of Proteome Research, 11(4). https://doi.org/10.1021/pr201052x

[133] Huang, D. W., Sherman, B. T., & Lempicki, R. A. (2009). Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. Nature Protocols, 4(1). https://doi.org/10.1038/nprot.2008.211

[134] Koch, A. L. (1969). The logarithm in biology. Journal of Theoretical Biology, 23(2). https://doi.org/10.1016/0022-5193(69)90040-x

[135] Pancaldi, V., & Bähler, J. (2011). In silico characterization and prediction of global protein-mRNA interactions in yeast. Nucleic Acids Research, 39(14). https://doi.org/10.1093/nar/gkr160

[136] Kannan, A., Emili, A., Frey, Brendan J. (2007). A Bayesian Model That Links Microarray mRNA Measurements to Mass Spectrometry Protein Measurements. Research in Computational Molecular Biology: 11th Annual International Conference. RECOMB 2007. pp.325-338

[137] Tuller, T., Kupiec, M. and Ruppin, E. (2007). Determinants Of Protein Abundance And Translation Efficiency In S. Cerevisiae. PLoS Computational Biology 3.12: e248.

[138] Ma, H. and Poon, R. (2011). Synchronization of HeLa Cells. Methods in Molecular Biology, pp.151-161.

[139] Schwanhausser, B., Busse, D., Li, N., Dittmar, G., Schuchhardt, J., et al. (2011). Global quantification of mammalian gene expression control. Nature 473: 337–342. PMID:21593866

[140] O'Leary, N., Wright, M., Brister, J., Ciufo, S., Haddad, D., McVeigh, R., Rajput, B., Robbertse, B., Smith-White, B., Ako-Adjei, D., Astashyn, A., Badretdin, A., Bao, Y., Blinkova, O., Brover, V., Chetvernin, V., Choi, J., Cox, E., Ermolaeva, O., Farrell, C., Goldfarb, T., Gupta, T., Haft, D., Hatcher, E., Hlavina, W., Joardar, V., Kodali, V., Li, W., Maglott, D., Masterson, P., McGarvey, K., Murphy, M., O'Neill, K., Pujar, S., Rangwala, S., Rausch, D., Riddick, L., Schoch, C., Shkeda, A., Storz, S., Sun, H., Thibaud-Nissen, F., Tolstoy, I., Tully, R., Vatsan, A., Wallin, C., Webb, D., Wu, W., Landrum, M., Kimchi, A., Tatusova, T., DiCuccio, M., Kitts, P., Murphy, T. and Pruitt, K. (2015). Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. Nucleic Acids Research, 44(D1), pp.D733-D745.

[141] Kans J. (2013). Entrez Direct: E-utilities on the UNIX Command Line. Entrez Programming Utilities Help: National Center for Biotechnology Information (US); 2010. Available from: https://www.ncbi.nlm.nih.gov/books/NBK179288/

[142] Cock, P.A., Antao, T., Chang, J.T., Chapman B.A., Cox, C.J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B. and de Hoon, M.J.L. (2009). Biopython: freely available Python tools for computational molecular biology and bioinformatics. Bioinformatics, 25, pp.1422-1423

[143] Gray, K.A., Yates, B., Seal, R.L., Wright, M.W. and Bruford, E.A. (2015). genenames.org: the HGNC resources in 2015. Nucleic Acids Res. 43. doi: 10.1093/nar/gku1071. PMID:25361968

[144] Puigbò, P., Bravo, I. and Garcia-Vallve, S. (2008). CAIcal: A combined set of tools to assess codon usage adaptation. Biology Direct, 3(1), p.38.

[145] Walker, J. (2005). The Proteomics Protocols Handbook. Dordrecht: Springer.

[146] Mathews, D. (2004). Using an RNA Secondary Structure Partition Function to Determine Confidence in Base Pairs Predicted by Free Energy Minimization. RNA, 10:1178-1190.

[147] Payne, S. (2015). The utility of protein and mRNA correlation. Trends in Biochemical Sciences, 40(1), pp.1-3.

[148] de Sousa Abreu, R., Penalva, L.O., Marcotte, E.M. and Vogel, C. (2009). Global signatures of protein and mRNA expression levels. Mol Biosyst 5: 1512–1526. PMID:20023718

[149] Csardi, G., Franks, A., Choi, D., Airoldi, E. and Drummond, D. (2015). Accounting for Experimental Noise Reveals That mRNA Levels, Amplified by Post-Transcriptional Processes, Largely Determine Steady-State Protein Levels in Yeast. PLOS Genetics, 11(5), p.e1005206.

[150] Feng, L. and Niu, D. (2007). Relationship Between mRNA Stability and Length: An Old Question with a New Twist. Biochemical Genetics, 45(1-2), pp.131-137.

[151] Mjelle, R., Hegre, S., Aas, P., Slupphaug, G., Drabløs, F., Sætrom, P. and Krokan, H. (2015). Cell cycle regulation of human DNA repair and chromatin remodeling genes. DNA Repair, 30, pp.53-67.

[152] Li, F., Long, T., Lu, Y., Ouyang, Q. and Tang, C. (2004). The yeast cell-cycle network is robustly designed. Proc. Natl. Acad. Sci., 101(14), pp.4781-4786.

[153] Stothard, P. (2000). The Sequence Manipulation Suite: JavaScript programs for analyzing and formatting protein and DNA sequences. Biotechniques. 28. pp.1102-1104

[154] Hornbeck, P.V., Zhang, B., Murray, B., Kornhauser, J.M., Latham, V., Skrzypek, E. (2015). PhosphoSitePlus, 2014: mutations, PTMs and recalibrations. Nucleic Acids Res. 43:D512-20.

[155] Mi, H., Huang, X., Muruganujan, A., Tang, H., Mills, C., Kang, D., Thomas, P.D. (2019). PANTHER version 14: more genomes, a new PANTHER GO-slim and improvements in enrichment analysis tools. Nucleic Acids Res. 47(D1) :D419-D426.

[156] Smedley, D., Haider, S., Durinck, S., Pandini, L., Provero, P., Allen, J., Arnaiz, O., Awedh, M. H., Baldock, R., Barbiera, G., Bardou, P., Beck, T., Blake, A., Bonierbale, M., Brookes, A. J., Bucci, G., Buetti, I., Burge, S., Cabau, C., ... Kasprzyk, A. (2015). The BioMart community portal: an innovative alternative to large, centralized data repositories. Nucleic Acids Research, 43(W1), W589–W598.

[157] Hanson, G. & Coller, J. (2018). Codon optimality, bias and usage in translation and mRNA decay. Nat Rev Mol Cell Biol, 19, pp.20–30.

[158] Zhou, Z., Dang, Y., Zhou, M., Li, L., Yu, C.H., Fu, J., Chen, S., Liu, Y. (2016) Codon usage is an important determinant of gene expression levels largely through its effects on transcription Proceedings of the National Academy of Sciences of the United States of America, 113(41), pp.E6117–E6125.

[159] Pihlasalo, S., Auranen, L., Hänninen, P., & Härmä, H. (2012). Method for estimation of protein isoelectric point. Analytical Chemistry, 84, pp.8253-8258.

[160] Kyte, J., & Doolittle, R. F. (1982). A simple method for displaying the hydropathic character of a protein. Journal of Molecular Biology, 157(1), pp.105-132.

[161] Bjellqvist, B., Hughes, G. J., Pasquali, C., Paquet, N., Ravier, F., Sanchez, J.C, Frutiger, S., & Hochstrasser, D. (1993). The focusing positions of polypeptides in immobilized pH gradients can be predicted from their amino acid sequences. ELECTROPHORESIS, 14(1). https://doi.org/10.1002/elps.11501401163

[162] Murphy, K. P. (2012). Machine Learning - A Probabilistic Perspective - Table-of-Contents. The MIT Press.

[163] Greenacre, M. and Blasius, J (2006). Multiple Correspondence Analysis and Related Methods, CRC Press. ISBN 1584886285.

[164] Lundberg, E., Fagerberg, L., Klevebring, D., Matic, I., Geiger, T., Cox, J., Älgenäs, C., Lundeberg, J., Mann, M., & Uhlen, M. (2010). Defining the transcriptome and proteome in three functionally different human cell lines. Molecular Systems Biology, 6(1), pp.450.

[165] Ramsköld, D., Wang, E. T., Burge, C. B., & Sandberg, R. (2009). An abundance of ubiquitously expressed genes revealed by tissue transcriptome sequence data. PLoS Computational Biology.

[166] Ramakrishnan, S. R., Vogel, C., Prince, J. T., Li, Z., Penalva, L. O., Myers, M., Marcotte, E. M., Miranker, D. P., & Wang, R. (2009). Integrating shotgun proteomics and mRNA expression data to improve protein identification. Bioinformatics.

[167] Irizarry, R. (2003). Exploration, normalization, and summaries of high density oligonucleotide array probe level data. Biostatistics, 4(2), pp.249-264.

[168] Wang, M., Weiss, M., Simonovic, M., Haertinger, G., Schrimpf, S. P., Hengartner, M. O., & Von Mering, C. (2012). PaxDb, a database of protein abundance averages across all three domains of life. Molecular and Cellular Proteomics, 11(8). https://doi.org/10.1074/mcp.O111.014704

[169] Tani, H., Mizutani, R., Salam, K. A., Tano, K., Ijiri, K., Wakamatsu, A., Isogai, T., Suzuki, Y., & Akimitsu, N. (2012). Genome-wide determination of RNA stability reveals hundreds of short-lived noncoding transcripts in mammals. Genome Research, 22(5). https://doi.org/10.1101/gr.130559.111

[170] Cambridge, S. B., Gnad, F., Nguyen, C., Bermejo, J. L., Krüger, M., & Mann, M. (2011). Systems-wide proteomic analysis in mammalian cells reveals conserved, functional protein turnover. Journal of Proteome Research, 10(12). https://doi.org/10.1021/pr101183k

[171] D. Szklarczyk et al. (2019). STRING v11: Protein-protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. Nucleic Acids Res., 47. D1, 2019, doi: 10.1093/nar/gky1131.

[172] A. A. Hagberg, D. A. Schult, and P. J. Swart. (2008). Exploring network structure, dynamics, and function using NetworkX. 7th Python in Science Conference (SciPy 2008).

[173] Legendre, A. (1805). Nouvelles méthodes pour la détermination des orbites des comètes. The Royal London Society, [online] (13), p.9. Available at: https://archive.org/details/nouvellesmthode00legegoog/page/n9 [Accessed 7 May 2019].

[174] Mayne, A. J., Rao, C. R., & Mitra, S. K. (1972). Generalized Inverse of Matrices and Its Applications. Operational Research Quarterly (1970-1977). https://doi.org/10.2307/3007981

[175] Sakia, R. M. (1992). The Box-Cox Transformation Technique: A Review. The Statistician. https://doi.org/10.2307/2348250

[176] Yeo, I. N. K., & Johnson, R. A. (2000). A new family of power transformations to improve normality or symmetry. Biometrika. https://doi.org/10.1093/biomet/87.4.954

[177] Pedregosa, F., Varoquaux, G., Gramfort., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R. Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. and Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 12. p2825-2830.

[178] Avsec, Ž., Agarwal, V., Visentin, D., Ledsam, J. R., Grabska-Barwinska, A., Taylor, K. R., Assael, Y., Jumper, J., Kohli, P., & Kelley, D. R. (2021). Effective gene expression prediction from sequence by integrating long-range interactions. Nature Methods, 18(10). https://doi.org/10.1038/s41592-021-01252-x

[179] J.A. Bondy and U.S.R Murty. (2008). Graph Theory (1st. ed.). Springer Publishing Company, Incorporated. ISBN:978-1-84628-969-9