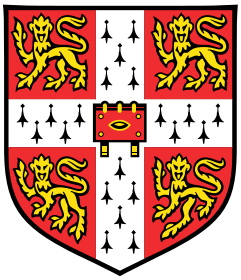
Deep neural networks and statistical models for studying single-cell DNA methylation



Christof Angermüller

European Bioinformatics Institute
University of Cambridge

This dissertation is submitted for the degree of
Doctor of Philosophy

Trinity Hall March 2017

Declaration

This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration except as declared in the Preface and specified in the text.

It is not substantially the same as any that I have submitted, or, is being concurrently submitted for a degree or diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text. I further state that no substantial part of my dissertation has already been submitted, or, is being concurrently submitted for any such degree, diploma or other qualification at the University of Cambridge or any other University of similar institution except as declared in the Preface and specified in the text.

This dissertation does not exceed the specified length limit of 60,000 words as defined by the Biology Degree Committee.

Christof Angermüller
March 2017

Acknowledgements

First and foremost, I would like to thank my PhD supervisor Oliver Stegle for his guidance and feedback throughout my PhD. I am very grateful to my Thesis Advisory Committee members, including Zoubin Ghahramani, Lars Steinmetz, and Sarah Teichman. I would like to thank my collaborators for the interesting data on which I could be working on, including Wolf Reik, Gavin Kelsey, Heather Lee, Sebastien Smallwood, Stephen Clark, and Iain Macaulay. Special thanks to Yarin Gal for his advice on estimating uncertainty in deep neural networks, as well as Brian Trippe and Leland Taylor for their comments on this thesis. Lastly, I would want to express my very profound gratitude to my family for having provided me a cozy home whenever I needed a break from my PhD.

Abstract

Epigenetics is the study heritable changes in gene activity that are not associated with changes of the DNA sequence. DNA methylation of cytosine nucleotides is an epigenetic modification that has critical roles in the regulation and maintenance of cell type-specific transcriptional programs. Recent technological advances have enabled profiling DNA methylation at single-cell resolution and thereby studying DNA methylation variability between cells. However, current technologies are limited by high levels of technical noise and incomplete genomic coverage, which renders the analysis of single-cell methylation profiles challenging.

This thesis contributes computational methods for the analysis of single-cell methylation data. First, we developed statistical methods for the genome-wide analysis of low-coverage single-cell methylation profiles. We applied these methods to mouse embryonic stem cells profiled using single-cell bisulfite sequencing, obtaining new insights into the variability of DNA methylation in different genomic contexts. Additionally, we considered cells profiled using parallel single-cell methylome and transcriptome sequencing, revealing regulatory associations between DNA methylation and gene expression at a genome-wide scale.

Second, we developed DeepCpG, a deep neural network for imputing low-coverage single-cell methylation profiles and thereby facilitating genome-wide analyses. We evaluated DeepCpG on single-cell methylation data from five cell types profiled using alternative sequencing protocols, demonstrating that DeepCpG is widely applicable and yields substantially more accurate predictions than previous methods.

Third, we developed approaches for analysing DNA methylation in single cells using DeepCpG. We show that our approaches enable discovering DNA sequence motifs that are associated with methylation states, identifying variance-associated motifs, and estimating the effect of single nucleotide mutations on DNA methylation.

In summary, this thesis proposes deep neural networks and statistical models for analysing single-cell DNA methylation sequencing data and for studying intercellular differences.

Table of contents

List of figures xiii

List of tables xix

Nomenclature xxi

1 Introduction 1

1.1 DNA methylation 1

1.2 Functions of DNA methylation 3

1.3 Protocols for methylation profiling 5

1.4 Contributions 8

2 Deep learning for computational biology 9

2.1 Machine learning 9

2.2 Artificial neural networks 12

2.2.1 Multilayer perceptron 13

2.2.2 Convolutional neural network 14

2.2.3 Recurrent neural network 16

2.3 Deep learning for regulatory genomics 18
2.3.1 Early applications of neural networks in regulatory genomics 20
2.3.2 Convolutional designs 20
2.3.3 In silico prediction of mutation effects 21
2.3.4 Joint prediction of multiple traits and further extensions 21
2.4 Deep learning for biological image analysis 22
2.4.1 First applications in computational biology 23
2.4.2 Analysis of whole cells, cell populations, and tissues 24
2.4.3 Re-using trained models 25
2.4.4 Interpreting and visualizing convolutional networks 26
2.5 Off-the-shelf tools and practical considerations 27
2.5.1 Deep learning frameworks 27
2.5.2 Data preparation 29
2.5.3 Model building 32
2.6 Discussion 37

3 Protocols and analysis of single-cell DNA methylation data 39

3.1 Estimating DNA methylation variability in embryonic stem cells 40
3.1.1 Single-cell bisulfite sequencing protocol 40
3.1.2 Method for estimating DNA methylation variability 43
3.1.3 Methylation variability in different genomic contexts 47
3.2 Estimating associations between DNA methylation and gene expression . . . 48
3.2.1 Parallel single-cell DNA methylation and gene expression profiling . 50

3.2.2 Methods for estimating associations between DNA methylation and gene expression . . . 53

3.2.3 Associations between DNA methylation and gene expression in different genomic contexts . . . 56

3.3 Discussion . . . 58

4 Deep neural networks for predicting DNA methylation 61

4.1 Motivation . . . 61

4.2 Existing methods and limitations . . . 62

4.3 DeepCpG model architecture . . . 64

4.3.1 DNA module . . . 64

4.3.2 CpG module . . . 66

4.3.3 Joint module . . . 67

4.3.4 Model training . . . 68

4.3.5 Software availability . . . 68

4.4 Prediction performance evaluation . . . 69

4.5 Discussion . . . 73

5 Analysis of deep neural networks for predicting DNA methylation 75

5.1 Discovering DNA sequence motifs . . . 75

5.2 Identifying variance-associated motifs . . . 79

5.3 Estimating the effect of DNA mutations . . . 83

5.4 Discussion . . . 85

6 Summary and future research 87

Appendix A Protocols and analysis of single-cell DNA methylation data **91**

Appendix B Deep neural networks for predicting DNA methylation **103**

Appendix C Analysis of deep neural networks for predicting DNA methylation **111**

References **119**

List of figures

1.1 Chemical reaction of cytosine to 5-methylcytosine. 2

1.2 DNA methyltransferase enzymes. 3

1.3 Workflow of DNA methylation profiling protocols. 5

2.1 Machine learning and representation learning. 10

2.2 Building blocks and learning principles of a neural network. 12

2.3 Convolutional neural network (CNN). 14

2.4 Recurrent neural network (RNN). 17

2.5 Principles of using neural networks for predicting molecular traits from DNA sequence. 19

2.6 Deep neural network for image analysis. 23

2.7 Feature extraction with pre-trained convolutional neural networks. 25

2.8 Data normalization and pre-processing for deep neural networks. 31

3.1 scBS-seq profiling protocol. 41

3.2 scBS-seq mapping efficiency and mean methylation levels. 42

3.3 CpG concordance and reproduction of bulk data. 43

3.4 Representation of single-cell methylation data. 44

3.5 Estimated methylation rates for Nanog locus. 47

3.6 Heatmap of 300 most variable sites. 48

3.7 Genome-wide clustering and estimated variance in genomic contexts. 49

3.8 Schematic overview of the scM&T-seq protocol. 50

3.9 Quality control of the scM&T-seq protocol. 51

3.10 Clustering analysis of transcriptome and methylome data. 52

3.11 Schematic representation of cell-specific and gene-specific correlation analysis between methylome and transcriptome. 53

3.12 Volcano plots of correlation coefficients. 55

3.13 Representative zoom-in view for the gene Esrrb. 57

3.14 Cell-specific correlation analysis 58

4.1 DeepCpG model training and applications. 65

4.2 Prediction performance of DeepCpG. 70

4.3 Prediction performance of the DeepCpG DNA module for DNA sequence windows of increasing length. 71

4.4 Prediction performance of DeepCpG for alternative genomic contexts. 72

4.5 Prediction performance of DeepCpG for alternative datasets. 73

5.1 Principal component analysis of discovered DNA sequence motifs. 76

5.2 Activity of DNA sequence motifs in genomic contexts. 78

5.3 Performance of DeepCpG DNA module to predict methylation variability. . . 79

5.4 Identification of motifs associated with mean methylation levels and cell-to-cell variability. 81

5.5 Functional assessment of predicted cell-to-cell variability. 82

5.6 Average genome-wide effect of single nucleotide mutations on CpG methylation estimated using DeepCpG. 83

5.7 Analysis and example visualization of estimated single nucleotide mutation effects for methylation QTLs (mQTLs). 84

6.1 Imputation of multiple molecular layers in multiple cells. 89

A.1 scBS-seq generates a digital output of DNA methylation. 92

A.2 Comparision of epigenetic heterogeneity in cells profiled using scM&T-seq and scBS-seq. 93

A.3 Quality metrics of scRNA-seq data obtained from mouse ESCs profiled using scM&T-seq. 94

A.4 Hierarchical clustering of DNA-methylation profiles generated by scM&T-seq and scBS-seq. 95

A.5 Correlation between single-cell methylomes and the methylome of a bulk cell population. 96

A.6 Principal-component analysis of gene body methylation and gene expression in serum-grown ESCs. 97

A.7 Scatter-plot matrix of principal components from methylation and gene expression profiles. 98

A.8 Correlation coefficients for associations between DNA-methylation profiles in alternative genomic contexts and gene expression levels. 99

A.9 Volcano plots for associations tests between DNA methylation profiles in alternative genomic context and gene expression levels. 100

A.10 Comparison of results of cell-specific correlation analysis with mean CpG methylation rate. 101

A.11 Comparison of results of cell-specific correlation analysis with CpG coverage. 102

B.1 Receiver operating characteristic and precision recall curves for predicting DNA methylation states using alternative methods. 104

B.2 Prediction performance of alternative methods and metrics across five datasets. 105

B.3 Quality metrics of cells profiled using scBS-seq and scRRBS-seq. 106

B.4 Correlation between filter activities and higher-level sequence features. 107

B.5 Prediction performance of the DeepCpG CpG module depending on the number of cells. 108

B.6 Prediction performances stratified by different metrics. 109

C.1 Importance of the DNA sequence motifs. 112

C.2 Effect of DNA sequence motifs on methylation. 113

C.3 Prediction performance of mean methylation levels. 114

C.4 Sensitivity of discovering highly-variable CpG sites. 115

C.5 Dependency between mean methylation levels and cell-to-cell variance. . . . 116

C.6 Linkage between motif correlation with sequence conservation and cell-to-cell variability. 117

C.7 Correlation between estimated mutation effects and DNA sequence conservation. 118

List of tables

2.1 Overview of existing deep learning frameworks, comparing four widely used software solutions. 28

2.2 Important hyperparameters of deep neural networks and recommended default values. 37

Nomenclature

Acronyms / Abbreviations

5mC	five-methylcytosine
AE	auto encoder
ANN	artificial neural network
AUC	area under the receiver operating characteristic curve
bp	base pair
BS-seq	bisulfite sequencing
CGI	CpG island
CNN	convolutional neural network
CPU	central processing unit
DNA	deoxyribonucleic acid
Dnmt	DNA methyltransferase
DNN	deep neural network
ESC	embryonic stem cell
FDR	false discovery rate
GPU	graphics processing unit
GRU	gated recurrent unit

HCC hepatocellular carcinoma cell

IAP intracisternal A-particle

LMR low methylated region

LSTM long short-term memory

MBD methyl-binding domain

MeDIP methylated DNA immunoprecipitation

mESC mouse embryonic stem cell

MII metaphase II oocytes

miRNA micro RNA

MLP multilayer perceptron

mQTL methylation quantitative trait loci

MRE methylation-sensitive restriction enzymes

NGS next generation sequencing

PBAT post bisulfite adaptor tagging

PC principal component

PCA principal component analysis

QTL quantitative trait loci

RBM restricted Boltzmann machine

ReLU rectified linear unit

RF random forest

RNA ribonucleic acid

RNN recurrent neural network

RRBS-seq reduced representation bisulfite sequencing

SAM S-adenosyl-L-methionine

SC single cell

scBS-seq single-cell bisulfite sequencing

scMT-seq single-cell methylome and transcriptome sequencing

scRRBS-seq single-cell reduced representation bisulfite sequencing

SNV single nucleotide variants

VAE variational auto encoder

WGBS whole genome bisulfite sequencing

Chapter 1

Introduction

1.1 DNA methylation

Epigenetics is the study of heritable changes of gene activity that cannot be explained by changes of the DNA sequence itself [90]. Epigenetic factors include DNA methylation, histone modifications, and small non-coding RNA sequences. The modification of histone proteins by methylation, acetylation, or ubiquitination influences DNA accessibility and thereby gene expression. Small RNA sequences, such as miRNAs, can bind to the RNA of a target gene and thereby silence gene expression. The focus of this thesis is on DNA methylation, a chemical modification of cytosine to 5-methylcytosine (5mC) by the attachment of a methyl group to the fifth carbon atom of cytosine (Figure 1.1). DNA methylation mainly occurs at CpG sites, i.e. cytosine nucleotides that are followed by a guanine nucleotide. CpG sites are underrepresented in the genome since they are 'mutational hotspots': 5mC is prone to be deaminated to thymine, resulting in a C→T mutation. CpG sites therefore cluster into CpG islands (CGIs), which are characterized by a GC density above 50%. CpG methylation plays an important role in biology, including gene regulation, imprinting, X-chromosome inactivation, and the repression of retroviruses [90, 179, 104, 157].

DNA methylation can also occur at non-CpG sites, including CHG and CHH sites, where H can be any nucleotide except for G. Non-CpG methylation has been reported in plants, fungi, and embryonic stem cells (ESCs), but the functional relevance of non-CpG methylation is still unclear [104, 249, 178, 191].

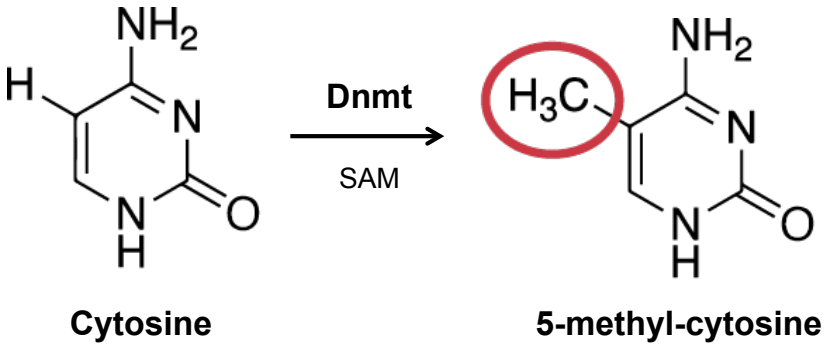


Figure 1.1 Chemical reaction of cytosine to 5-methylcytosine. Cytosine is converted to 5-methylcytosine (5mC) by the transfer of a methyl group from S-adenosyl-L-methionine (SAM) to the fifth carbon atom of the cytosine pyrimidine ring, catalyzed by enzymes of the DNA methyltransferase (Dnmt) family.

DNA methylation is established during embryonic development by a series of de novo methylation and demethylation events. Enzymes of the DNA methyltransferase (Dnmt) family catalyse DNA methylation by transferring the methyl-group of S-adenosyl-L-methionine (SAM) to cytosine (Figure 1.1). Three active Dnmt enzymes have been identified in mammals: Dnmt1, Dnmt3a, and Dnmt3b. Although these enzymes are structurally similar, they serve different functions and are expressed at different time points during embryonic development [157, 105, 24]. Dnmt3a and Dnmt3b are also known as *de novo Dnmts* since they establish new tissue-specific methylation patterns during embryogenesis (Figure 1.2 (a)). Whereas Dnmt3a is ubiquitously expressed in all cell types, Dnmt3b is only expressed in differentiating cells. Dnmt1 is also known as *maintenance Dnmt* since it maintains DNA methylation during cell division and cell differentiation. Dnmt1 preferentially binds to hemimethylated DNA during DNA replication, and methylates cytosine residues on the newly synthesized DNA strand (Figure 1.2 (b)). Dnmt1 regulates genomic imprinting by repressing either the maternal or paternal copy of genes, and is involved in DNA repair.

Although DNA methylation is primarily established during embryonic development and maintained in replicating cells, methylation patterns can be changed by environmental factors and stochastic fluctuations. DNA methylation can be erased passively by the inhibition of Dnmt1, resulting in gradually decreasing methylation levels in differentiating cells. Active demethylation can occur in both differentiating and non-differentiating cells, which involves the removal of the methyl group from 5mC through are series of chemical reactions [149, 242].

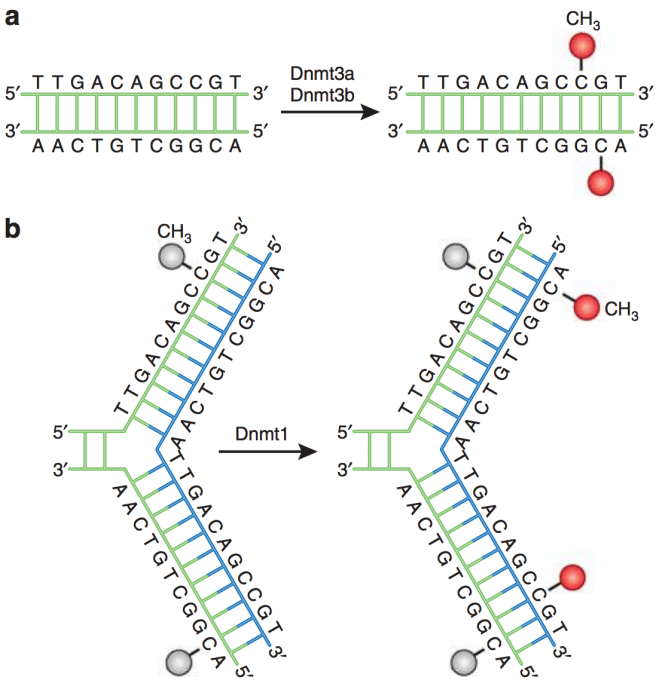


Figure 1.2 DNA methyltransferase enzymes. DNA methyltransferase (Dnmt) is a family of enzymes that catalyzes the conversion from cytosine to 5-methylcytosine. (a) Dnmt3a and Dnmt3b can establish new methylation patterns by methylating DNA de novo. (b) Dnmt1 maintains existing methylation patterns during cell replication by methylating hemimethylated DNA. It binds close to the replication fork and methylates CpG sites on the newly synthesized daughter strand at CpG sites that are methylated on the parent strand. Source: Moore et al. [157].

1.2 Functions of DNA methylation

The function of DNA methylation varies between genomic contexts [24, 157, 104, 27]. Promoter methylation has been associated with regulatory effects on gene expression [157, 104, 27]. About 70% of all promoters in the human genome are CGI promoters, i.e. promoters that are overlapped by at least one CGI. Promoter CGIs are evolutionary conserved, which indicates their functional importance. Most promoter CGIs are unmethylated and nucleosome depleted, which has been associated with increased DNA accessibility and gene expression levels [157]. The high GC density in CGI promoters has further been associated with an increased binding of transcription factors that regulate gene expression [157]. Methylation of promoter CGIs blocks RNA polymerases and thus inhibits gene expression. This is important to variably activate and deactivate genes during cell differentiation and to establish tissue-specific expression patterns in combination with other epigenetic factors. CGI promoter methylation further regulates imprinting by inhibiting the expression of either the maternal or paternal allele of genes. Another import role of CGI promoter methylation is X chromosome inactivation—the deactivation of one copy of the X chromosome in female mammalian cells by bulk methylation of all gene promoters [24].

Unlike CGI promoters, which are predominantly unmethylated, non-CGI promoters are more heterogeneous. Although evidence exists that non-CGI promoter methylation may be involved in tissue-specific gene regulation [157, 104], the exact regulatory mechanisms are still unknown.

Intergenic regions are CpG poor and tend to be methylated. Methylation of intergenic regions is important for repressing transposable and viral elements [157, 104, 179], which account for about 45% of the genomic DNA in mammalian cells. These elements are either repressed by CpG methylation or inactive due to C→T mutations acquired by the spontaneous deamination of 5mC. For example, the intracisternal A-particle (IAP) is a detrimental retrovirus that resides in the mouse genome [225]. IAPs are highly methylated by Dnmt1 and thereby repressed. Knockout of Dnmt1 induces demethylation and hence the expression and IAPs and cell death [225, 97].

Gene bodies are CpG poor and mostly methylated, similar to intergenetic regions. Gene body methylation likewise serves as a mechanism to repress transposable and viral elements [157, 104, 179]. Although gene bodies are generally CpG poor, they can contain CGIs. However, methylation of gene body CGIs does not block transcription, unlike promoter CGIs, and the implications of gene body methylation in gene regulation is still unclear: Whereas some studies found positive associations between gene body methylation and gene expression, others have reported negative associations in slowly dividing and non-dividing cells.

Gene enhancers are relatively short (50-1500 bp) genetic elements that promote gene expression [29, 170]. They tend to be GC poor and to have heterogeneous methylation levels. For example, low methylated regions (LMRs [204]) are indicative of distal enhancer elements. They are characterized by an average methylation rate of 30%, which can fluctuate considerably during cell differentiation. Although some studies found negative associations between enhancer methylation and gene expression, further studies are needed to explore the regulatory function of these associations.

Insulators are genetic elements that block promoter-enhancer interactions [38]. Insulators are bound by the CTCF zinc finger protein and transported close to gene promoters by looping of the DNA strand. Insulator methylation can block CTCF binding, and thereby support promoter-enhancer interactions. However, the roles of insulator methylation are still unclear [157, 104].

1.3 Protocols for methylation profiling

Several methods have been developed for profiling DNA methylation, which can be broadly grouped into protocols based on enzymatic digestion, affinity enrichment, and bisulfite conversion [239, 187, 172, 94, 120].

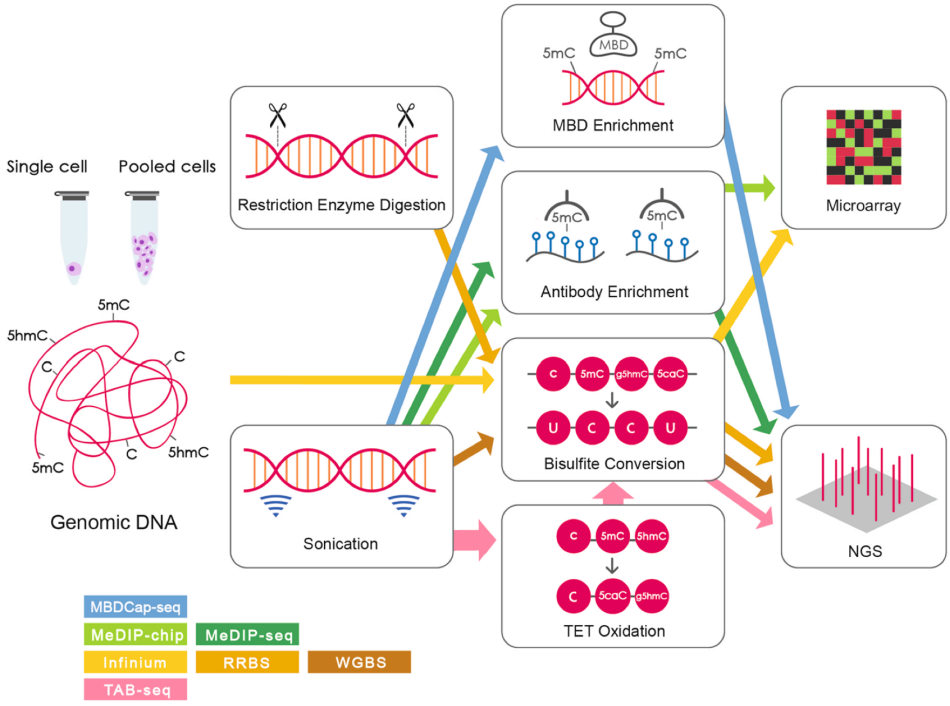


Figure 1.3 Workflow of DNA methylation profiling protocols. DNA starting material is extracted from either a bulk population of cells or a single cell and fragmented by enzymatic digestion or sonication. Enrichment based profiling protocols enrich for methylated DNA fragments by either methyl-binding domain (MBD) proteins or antibodies. Bisulfite conversion-based protocols treat DNA fragments by bisulfite to convert unmethylated cytosine to thymine. DNA methylation levels are quantified by microarray hybridization of next generation sequencing (NGS). Source: Yong et al. [239].

Protocols based on enzymatic digestion leverage methylation-sensitive restriction enzymes (MREs), which have differential digestion properties for methylated and unmethylated CpG sites. The two most common MREs are HpaII, which cleaves unmethylated CpG sites, and Msp1, which cleaves methylated CpG sites. Protocols proceed by enzymatically digesting DNA using a specific MRE, followed by DNA methylation quantification using either array hybridization (MRE-chip [189]) or sequencing (MRE-seq [148]). Enzymatic digestion-based approaches are cost-effective and enable genome-wide methylation profiling. However, their

resolution is limited to regions adjacent to MRE recognition sites, and they cannot quantify the methylation level of single CpG sites.

Affinity enrichment-based protocols enrich for methylated DNA fragments by either methyl-binding domain (MBD) proteins or antibodies that target 5mC. Methylated DNA immunoprecipitation (MeDIP) uses anti-methylcytosine antibodies to bind and quantify 5mC. The protocol consists of DNA fragmentation by sonication, enrichment of methylated fragments by immunoprecipitation, and methylation quantification by either array hybridization (MeDIP-chip [230]) or sequencing (MeDIP-seq [37]). MeDIP is cost effective and can differentiate between CpG, CHG, and CHH methylation contexts. Since MeDIP only requires small amounts of DNA starting material, it is further applicable to small cell populations. On the downside, the resolution of MeDIP is limited to 100-300 bp long fragments and the method is biased towards hypermethylated regions.

Protocols based on bisulfite conversion enable quantifying the methylation levels of single CpG sites. In bisulfite conversion-based protocols, DNA is fragmented using sonication or enzymatic digestion, and DNA fragments treated with sodium bisulfite. Sodium bisulfite converts unmethylated cytosine to uracil, which eventually turns into thymine by PCR amplification. The resulting C→T conversions are detected either using array hybridization or next generation sequencing and thereby CpG sites classified as methylated or unmethylated.

Illumina's 450K bead-chip is the most widely used bisulfite microarray for profiling DNA methylation in human, which integrates bisulfite treatment, PCR amplification, and hybridization [26]. Two distinct primers are used to distinguish between methylated and unmethylated fragments, which are labelled by different fluorescence dyes and hybridized to bead arrays. The chip probes about 450K CpG sites in the human genome, which cover most CGIs. It is cost-effective but biased towards CpG dense contexts.

Whole genome bisulfite sequencing (BS-seq [220]) detects C→T conversions by sequencing bisulfite treated fragments, and aligning the sequenced fragments back to a reference genome. BS-seq is considered as the gold standard protocol since it can profile DNA methylation at single cytosine resolution genome-wide, and can differentiate between CpG, CHG, and CHH contexts. However, BS-seq is more expensive owing to genome-wide deep sequencing of bisulfite-treated fragments.

Reduces representation bisulfite sequencing (RRBS-seq [151, 197]) is a cost-effective alternative to BS-seq, which quantifies DNA methylation only for a subset of genomic fragments.

The method is based on the observation that Msp1-digested DNA fragments of size 20-200 bp cover about 85% of all CpG sites, including most CGIs. By sequencing only bisulfite treated fragments of size 20-200 bp, RRBS-seq is more cost effective than BS-seq, however, biased towards CpG dense regions.

As a consequence of DNA degradation by multiple purification steps and bisulfite treatment, conventional bisulfite protocols require a relatively large amount of DNA starting material, which is usually obtained from a bulk population of thousands or millions of cells. Hence, bulk protocols quantify average methylation levels and cannot assess methylation heterogeneity between cells.

Recent technological advances enabled to reduced DNA degradation and thereby to profile DNA methylation in single cells. Guo et al. [79] adapted the RRBS protocol by integrating all experimental steps before PCR amplification into a single tube. Their reduced representation protocol (scRRBS-seq) probes about 1-10% of all CpG sites, is cost-effective, but limited to CpG dense regions similar to RRBS-seq. We and colleagues proposed scBS-seq [196], the first protocol to profile DNA methylation in single cells genome-wide. scBS-seq reduces DNA degradation by a modification of the post bisulfite adaptor tagging (PBAT [155]) protocol. Here, the DNA is treated with bisulfite prior to adapter ligation instead of afterwards, which simultaneously fragments the DNA and converts unmethylated cytosine to thymine. The protocol covers about 10-30% of all CpG sites per cell and will be described in more detail in section 3.1.

Single-cell protocols have multiple advantages over bulk protocols [187]. First, they enable the study of DNA methylation variability and differences between individual cells, such as ESCs or cancer cells. Second, single-cell protocols for parallel profiling of multiple molecular layers, e.g. DNA methylation and gene expression, provide the potential to analyse the regulatory relationship between these layers. Third, single-cell protocols are applicable to small cell populations that cannot be profiled by bulk protocols due to insufficient DNA starting material. On the downside, single-cell protocols are limited by high levels of technical noise and incomplete CpG coverage, which renders downstream analyses challenging. We therefore developed computational methods for the genome-wide analysis of single-cell DNA methylation profiling data, which will be described in the subsequent chapters.

1.4 Contributions

In this chapter, we have introduced DNA methylation, its roles in biology, and protocols for profiling DNA methylation in bulk populations of cells and single cells. Data from single-cell profiling protocols offer great potential for studying intercellular differences but are difficult to analyse due to high levels of technical noise and incomplete CpG coverage. This thesis contributes deep neural networks and statistical methods for the analysis of single-cell DNA methylation data, which are applicable genome-wide, account for incomplete CpG coverage, and provide mechanistic insights.

In chapter 2, we will introduce deep neural networks for computational biology. We will contrast deep neural networks with conventional machine learning models, present different network architectures, and review applications in computational biology. Chapter two forms the foundation of chapter four and five on deep neural networks for predicting and analysing DNA methylation. The work presented in this chapter is based on Angermueller et al. [8].

In chapter 3, we will present protocols and computational methods for the genome-wide analysis of single-cell DNA methylation. We will first describe scBS-seq, a protocol that enables profiling DNA methylation in single cells genome-wide, as well as a statistical model for assessing DNA methylation heterogeneity between cells. We will subsequently describe scM&T-seq, an extension of scBS-seq that enables parallel profiling of DNA methylation and gene expression in single cells, as well as methods for quantifying associations between DNA methylation and gene expression. The work presented in this chapter is based on Smallwood et al. [196] and Angermueller et al. [7].

In chapter 4, we will present *DeepCpG*, a deep neural network for predicting DNA methylation in single cells. We will discuss the limitations of existing methods, describe the DeepCpG model architecture, and show that DeepCpG yields considerably more accurate predictions than existing methods. The work presented in this chapter is based on Angermueller et al. [9].

In chapter 5, we will describe approaches for analysing DNA methylation using DeepCpG. We will show that DeepCpG can be applied to discover DNA sequence motifs that are associated with methylation states, to identify variance-associated motifs, and to estimate the effect of single nucleotide mutations on DNA methylation. The work presented in this chapter is based on Angermueller et al. [9].

In chapter 6, we will summarize this thesis and provide an outlook on future research.

Chapter 2

Deep learning for computational biology

Machine learning is a pillar of modern computational biology. The latest machine learning methods garnering significant attention are deep neural networks, which have led to breakthroughs in various fields [124, 186, 21], including computational biology [5, 51, 61, 112, 128, 199, 227, 247]. In the following, we will introduce deep neural networks and review applications in regulatory genomics and biological image analysis. This chapter forms the foundation of chapter 4 and chapter 5, which will describe methods based on deep neural networks for predicting and analysing DNA methylation. The presented work is based on Angermueller et al. [8], which was joint work of Christof Angermueller, Tanel Pärnamaa, Leopold Parts, and Oliver Stegle.

> Individual contributions: Christof Angermueller and Oliver Stegle reviewed applications of deep neural networks in regulatory genomics and summarized off-the-shelf tools and practical considerations. Tanel Pärnamaa and Leopold Parts reviewed applications of deep neural network in biological image analysis. All authors wrote the manuscript.

2.1 Machine learning

Machine learning methods are general-purpose approaches to learn functional relationships from data without the need to define them a priori [82, 154, 158]. In computational biology, their appeal is the ability to derive predictive models without a need for strong assumptions

about underlying mechanisms, which are frequently unknown or insufficiently defined. As a case in point, the most accurate prediction of gene expression levels is currently made from a broad set of epigenetic features using sparse linear models [42, 109] or random forests [129]; how the selected features determine the transcript levels remains an active research topic. Predictions in genomics [134, 146], proteomics [215], metabolomics [111], or sensitivity to compounds [59] all rely on machine learning approaches as a key ingredient.

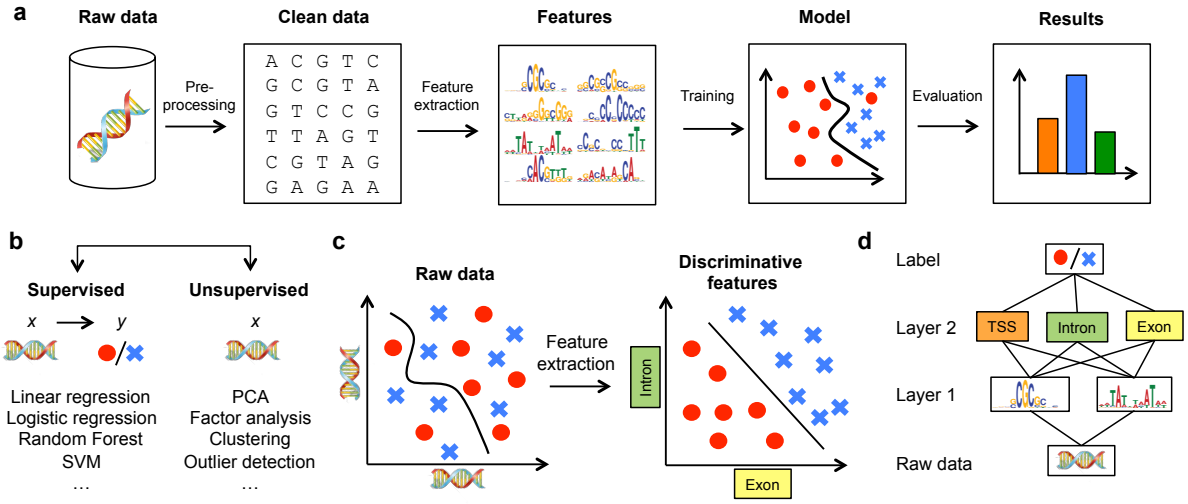


Figure 2.1 Machine learning and representation learning. (a) The classical machine learning workflow can be broken down into four steps: data pre-processing, feature extraction, model learning, and model evaluation. (b) Supervised machine learning methods relate input features x to an output label y, whereas unsupervised method learn factors about x without observed labels. (c) Raw input data are often high dimensional and related to the corresponding label in a complicated way, which is challenging for many classical machine learning algorithms (left plot). Alternatively, higher-level features extracted using a deep model may be able to better discriminate between classes (right plot). (d) Deep networks use a hierarchical structure to learn increasingly abstract feature representations from the raw data.

Most of these applications can be described within the canonical machine learning workflow, which involves four steps: data cleaning and pre-processing, feature extraction, model fitting, and evaluation (Figure 2.1 (a)). It is customary to denote one data sample, including all covariates and features as input x (usually a vector of numbers), and label it with its response variable or output value y (usually a single number) when available.

A supervised machine learning model aims to learn a function $y = f(x)$ from a list of training pairs $(x_1, y_1), (x_2, y_2), \ldots$ for which data are recorded (Figure 2.1 (b)). One typical application in biology is to predict the viability of a cancer cell line when exposed to a chosen drug [59, 152]. The input features x would capture somatic sequence variants of the cell line, chemical makeup

of the drug, and its concentration, which together with the measured viability (output label y) can be used to train a support vector machine, a random forest classifier or a related method (functional relationship f). Given a new cell line (unlabelled data sample x^*) in the future, the learned function predicts its survival $y^* = f(x^*)$, even if f resembles more of a black box, and its inner workings of why particular mutation combinations influence cell growth are not easily interpreted. Both regression (where y is a real number), and classification (where y is a categorical class label) can be viewed in this way. As a counterpart, unsupervised machine learning approaches aim to discover patterns from the data samples x itself, without the need for output labels y. Methods such as clustering, principal components analysis, and outlier detection are typical examples of unsupervised models applied to biological data.

The inputs x, calculated from the raw data, represent what the model 'sees about the world', and their choice is highly problem specific (Figure 2.1 (c)). Deriving most informative features is essential for performance, but the process can be labour-intensive and requires domain knowledge. This bottleneck is especially limiting for high dimensional data; even computational feature selection methods do not scale to assess the utility of vast number of possible input combinations. A major recent advance in machine learning is automating this critical step by learning a suitable representation of the data with deep artificial neural networks [21, 124, 186] (Figure 2.1 (d)). Briefly, a deep neural network takes the raw data at the lowest (input) layer, and transforms them into increasingly abstract feature representations by successively combining outputs from the preceding layer in a data-driven manner, encapsulating highly complicated functions in the process (Figure 2.2). Deep learning is now one of the most active fields in machine learning and has been shown to improve performance in image- and speech recognition [54, 76, 86, 119, 241], natural language understanding [13, 135, 213, 234], and most recently, in computational biology [5, 51, 61, 112, 128, 199, 227, 247].

The potential of deep learning in high throughput biology is clear: in principle, it allows to better exploit the availability of increasingly large and high-dimensional datasets (e.g. from DNA sequencing, RNA measurements, flow cytometry, or automated microscopy) by training complex networks with multiple layers that capture their internal structure (Figure 2.1 (c)). The learned networks discover high-level features, improve performance over traditional models, increase interpretability and provide additional understanding about the structure of the biological data.

In the following, we will describe prominent deep neural network architectures and review applications in regulatory genomics and biological image analysis. We will further provide practical pointers, summarise current software solutions, and give recommendations for ap-

plying them to data. Finally, we will discuss both the potential and possible pitfalls of deep learning and contrast deep learning with traditional machine learning and statistical analysis.

2.2 Artificial neural networks

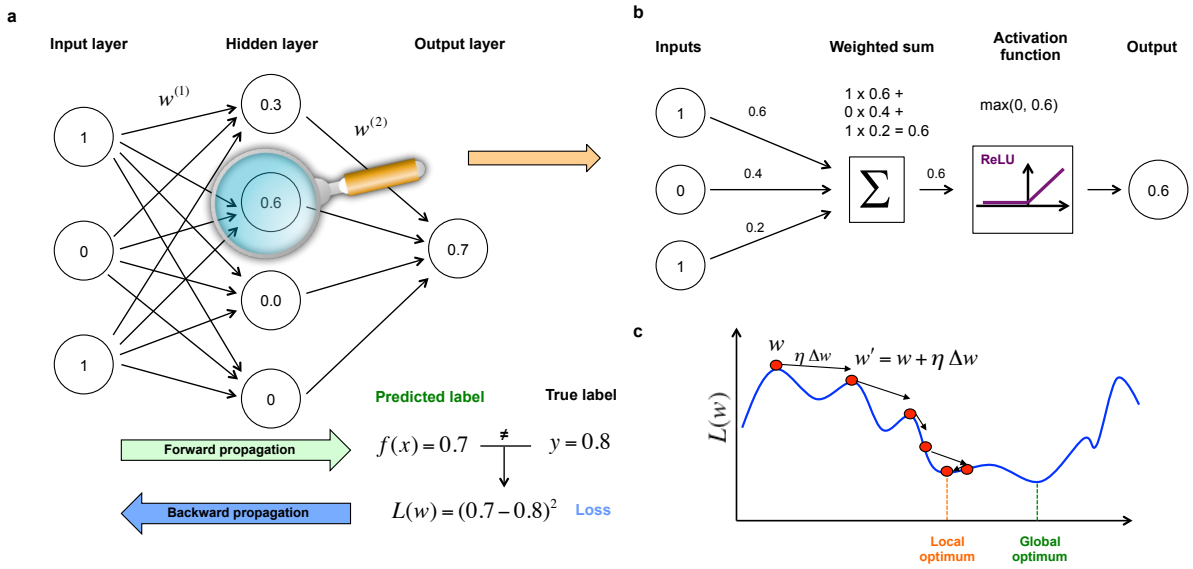


Figure 2.2 Building blocks and learning principles of a neural network. (a) Fully connected feedforward neural network with one input layer, hidden layer, and output layer. Each layer i consists of neurons which are connected to all neurons in the previous layer with weights $w(i)$. Given input x, neuron activations are calculated and forward propagated to the output layer to obtain a prediction $f(x)$. (b) Zoom-in view into one neuron, which computes the weighted sum of its inputs and applies a rectification function that thresholds negative signals to 0, and passes through positive signal. (c) Gradient-based optimization of the loss function $L(w)$. In each step, the current weight vector (red dot) is moved along the direction of steepest descent Δw (direction arrow) by learning rate η (length of vector). Decaying the learning rate over time allows to explore different domains of the loss function by jumping over valleys at the beginning of the training (left side), and fine-tune parameters with smaller learning rates in later stages of the model training.

An artificial neural network, initially inspired by neural networks in the brain [64, 150, 181] consists of layers of interconnected compute units (neurons). In the canonical configuration, the network receives data in an input layer, which are then transformed in a nonlinear way through multiple hidden layers, before final outputs are computed in the output layer (Figure 2.2 (a)). Neurons in a hidden or output layer are connected to all neurons in the previous layer. Each neuron computes a weighted sum of its inputs, and applies a nonlinear activation function

to calculate its output (Figure 2.2 (b)). The most popular activation function is the Rectified linear unit (ReLU; Figure 2.2 (b)), since it allows faster learning compared to alternatives (e.g. sigmoid or tanh unit) [74]. The depth of a neural network corresponds to the number of hidden layers, and the width to the maximum number of neurons in one of its layers. As it became possible to train networks with larger numbers of hidden layers, artificial neural networks were rebranded to 'deep networks'.

The weights between neurons are free parameters that capture the model's representation of the data, and are learned from input/output samples. Learning minimizes a loss function that measures the fit of the model output to the true label of a sample (Figure 2.2 (a), bottom). This minimization is challenging, since the loss function is high dimensional and non-convex, similar to a landscape with many hills and valleys (Figure 2.2 (c)). It took several decades before the backward propagation algorithm was first applied to compute a loss function gradient via chain rule for derivatives [182], ultimately enabling efficient training of neural networks using stochastic gradient descent. During learning, the predicted label is compared with the true label to compute a loss for the current set of model weights. The loss is then backward propagated through the network to compute the gradients of the loss function and update (Figure 2.2 (a)). While learning in deep neural networks remains an active area of research, existing software packages (Table 2.1) can already be applied without knowledge of the mathematical details involved.

Several architectures have been developed for specific applications, which differ in the way neurons are arranged. These include the convolutional neural network for images, the recurrent neural network for sequential data [135, 211], or the restricted Boltzmann machine [87, 185] and autoencoder [3, 88, 115] for unsupervised learning. The multilayer perceptron, convolutional neural network, and recurrent neural network are most widely used in biology, and will be outlined in the following.

2.2.1 Multilayer perceptron

The multilayer perceptron (MLP) [181] is the most basic artificial neural network, which consists of a sequence of fully-connected layers. Specifically, all neurons in a layer are connected to all neurons in the previous layer (Figure 2.2 (a)). However, no connections exits between neurons within the same layer, as opposed to the recurrent neural network (Section 2.2.3). A MLP with many layers and neurons can be very powerful, but is limited by a

high number of parameters, which scales quadratically with the number of neurons per layer. Specialized architectures have been developed to reduce the number of parameters and avoid overfitting, which include the convolutional neural network.

2.2.2 Convolutional neural network

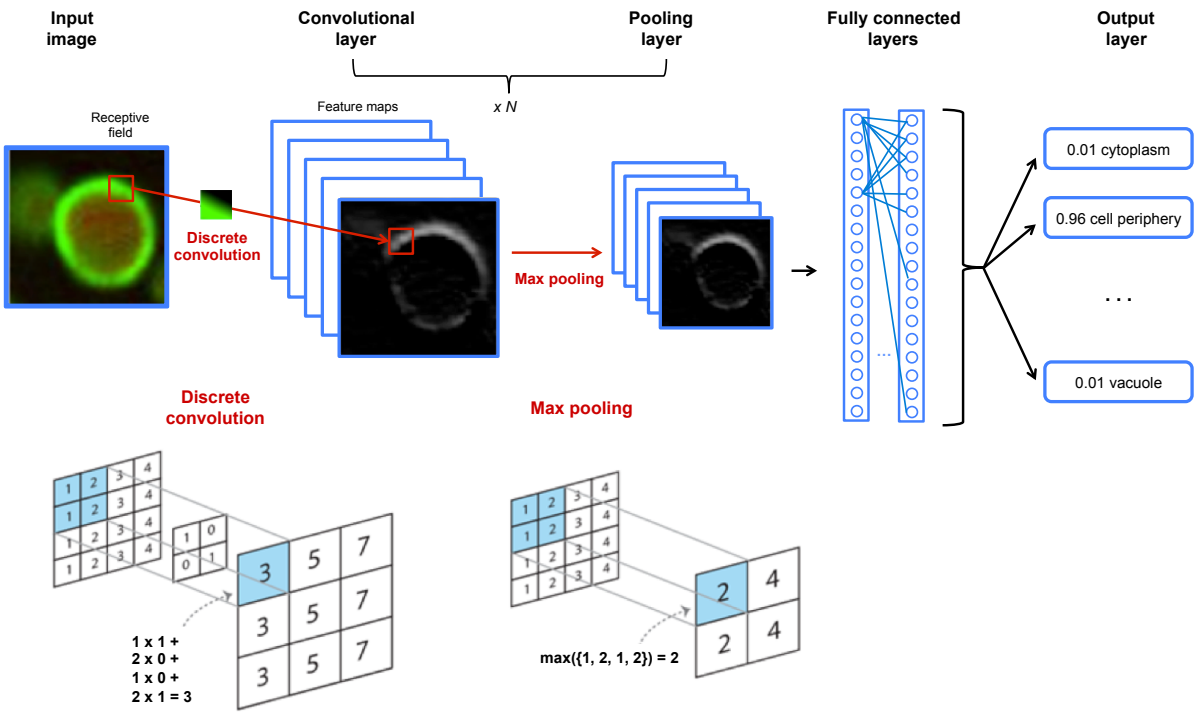


Figure 2.3 Convolutional neural network (CNN). (a) A typical CNN consists of a number of convolutional and pooling layers, two fully connected layers, and one output layer. Each convolutional layer consists of multiple feature maps, with neurons responding to a particular feature in a receptive field (red square). One feature map responding to the membrane of a cell at a particular angle is highlighted on the edge. (b) Neuron activities result from a discrete convolution of their receptive field. (c) Max pooling computes the maximum neuron activity over a small patch, reducing the dimension of a convolutional layer.

Convolutional neural networks (CNNs) were originally inspired by cognitive neuroscience and Hubel and Wiesel's seminal work on the cat's visual cortex, which was found to have simple neurons that respond to small motifs in the visual field, and complex neurons that respond to larger ones [96, 95].

CNNs are designed to model input data in the form of multi-dimensional arrays, such as two-dimensional images with three colour channels [83, 102, 119, 123, 216, 241], or one-

dimensional genomic sequences with one channel per nucleotide [5, 9, 112, 247]. The high dimensionality of these data (up to millions of pixels for high-resolution images) renders training a fully connected neural network challenging, as the number of parameters of such a model would typically exceed the number of training data to fit them. To circumvent this, CNNs make additional assumptions on the structure of the network, thereby reducing the effective number of parameters to learn.

A convolutional layer consists of multiple maps of neurons, so called feature maps or filters, with their size being equal to the dimension of the input image (Figure 2.3). Two concepts allow reducing the number of model parameters: local connectivity and parameter sharing. First, unlike in a fully connected network, each neuron within a feature map is only connected to a local patch of neurons in the previous layer, the so-called receptive field. Second, all neurons within a given feature map share the same parameters. Hence, all neurons within a feature map scan for the same feature in the previous layer, however at different locations. Different feature maps might, for example, detect edges of different orientation in an image, or sequence motifs in a genomic sequence. The activity of a neuron is obtained by computing a discrete convolution of its receptive field, i.e. computing the weighted sum of input neurons, and applying an activation function.

In most applications, the exact position and frequency of features is irrelevant for the final prediction, such as recognizing objects in an image. Using this assumption, the pooling layer summarizes adjacent neurons by computing, for example, the maximum or average over their activity, resulting in a smoother representation of feature activities. By applying the same pooling operation to small image patches that are shifted by more than one pixel, the input image is effectively down-sampled, thereby further reducing the number of model parameters.

A CNN typically consists of multiple convolutional and pooling layers, which allows learning more and more abstract features at increasing scales from small edges, to object parts, and finally entire objects. One or more fully connected layers can follow the last pooling layer. Model hyperparameters such as the number of convolutional layers, number of feature maps, or the size of receptive fields are application dependent and should be strictly selected on a validation data set (see below).

2.2.3 Recurrent neural network

Recurrent neural networks (RNNs) are designed to model input sequences of variable length. They have been successfully applied to modelling long-range dependencies in natural text [213, 13, 234], acoustic signals [54, 75, 86], and biological sequences [2, 199, 133, 174]. The input of a RNN is a sequence of vectors $x_1, \ldots, x_N$, such as the words of a sentence, nucleotides of a genomic sequence, or amino acids of a protein sequence. A RNN is called *recurrent*, because it applies the same operation at every time step:

$$h_t = \text{sigmoid}(W_{hx} x_t + W_{hh} h_{t-1} + b_h) \tag{2.1}$$

h_t is the hidden state vector of the RNN, which is sequentially updated based on the current input x_t and the previous hidden state h_{t-1}. h_t is also denoted as the *memory* of the network, since it memorizes the input sequence $x_1, \ldots, x_t$ up to time step t. The last hidden state h_T eventually memorises the entire input sequence. Importantly, the weights W_{hx}, W_{hh}, and b_h are shared across time steps, which allows the network to process sequences of variable length. The output y_t at time step t is conditioned on the hidden state h_t, and thereby the entire past sequence:

$$y_t = f(W_{yh} + b_y) \tag{2.2}$$

f is a task-specific activation function, such as the sigmoid function to model binary outputs or the softmax function to model categorical outputs. A RNN can either have a single output y_T at the end of the sequence, for example to predict the class of a protein, or it can have an output at every time step t, for example to predict the secondary structure for an entire protein sequence. The network parameters $\{W_{hx}, W_{hh}, b_h, W_{yh}, b_y\}$ are trained similarly to a feedforward neural network by comparing model outputs y_t with true output labels y'_t (Figure 2.2), and by backpropagating the error signal back over time. RNNs have long been considered as hard to train, since an incorrect parameter initialization can lead to vanishing or exploding gradients. While exploding gradients can be controlled by clipping [169], vanishing gradients can be circumvented by the use of advanced RNN architectures, such as the long short-term memory (LSTM [89]) or gated recurrent unit (GRU [45]) network. The core idea behind the LSTM and GRU is the use of additional gates to update the memory of the network only at certain time steps.

The outputs y_t of a RNN only depend on the past sequence, not the future sequence. In some applications, however, it is expedient to leverage information from the entire sequence. For

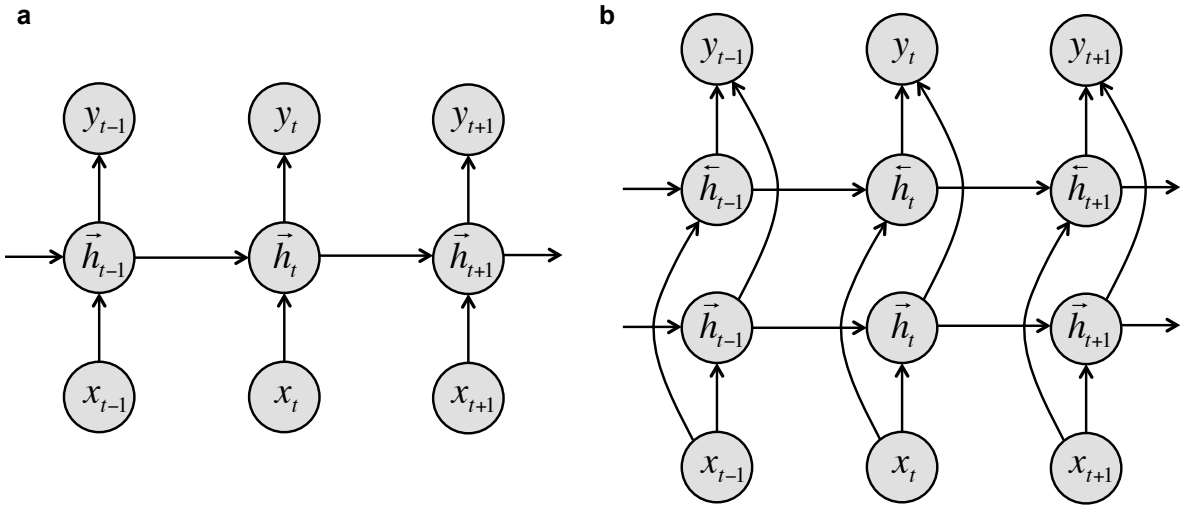


Figure 2.4 Recurrent neural network (RNN). (a) A recurrent neural network scans an input sequence x from left to right and updates at every time step t its hidden state vector h_t based on the previous hidden state h_{t-1} and the current input x_t. Outputs y_t are conditioned on h_t, which encodes the past sequence up to time step t. (b) A bidirectional RNN consists of a forward and backward RNN, which scan the input sequence in both directions. Outputs y_t are conditioned on both the hidden state vector $\overrightarrow{h}_t$ of the forward RNN, and the hidden state vector $\overleftarrow{h}_t$ of the backward RNN, thereby on the entire sequence.

example, the secondary structure of a protein sequence at position t can depend on both amino acids to the left and to the right of t. A bidirectional RNN conditions its outputs y_t on both the past and the future sequence. It consists of a forward and backward RNN, which scan the input sequence forward and backward, respectively, and concatenates their hidden state vectors:

$$\overrightarrow{h}_t = \text{sigmoid}(\overrightarrow{W}_{hx} x_t + \overrightarrow{W}_{hh} h_{t-1} + \overrightarrow{b}_h) \tag{2.3}$$

$$\overrightarrow{h}_t = \text{sigmoid}(\overrightarrow{W}_{hx} x_t + \overrightarrow{W}_{hh} h_{t-1} + \overrightarrow{b}_h) \tag{2.4}$$

$$h_t = [\overrightarrow{h}_t, \overleftarrow{h}_t] \tag{2.5}$$

$\overrightarrow{h}_t$ and $\overleftarrow{h}_t$ is the hidden state vector at time step t of the forward and backward RNN, which summarizes the input sequence before and after time step t, respectively. The concatenated hidden state vector h_t summarizes both the past and future input sequence, and thereby conditions y_t on the entire sequence.

2.3 Deep learning for regulatory genomics

Conventional approaches for regulatory genomics relate sequence variation to changes in molecular traits. One approach is to leverage variation between genetically diverse individuals to map quantitative trait loci (QTL). This principle has been applied to identify regulatory variants that affect gene expression levels [156, 171], DNA methylation [18, 71], histone marks [77, 228], and proteome variation [4, 16, 168, 223] (Figure 2.5 (a)). Better statistical methods have helped to increase the power to detect regulatory QTLs [107, 167, 177, 205], however any mapping approach is intrinsically limited to variation that is present in the training population. Thus, studying effects of rare mutations in particular requires extremely large datasets.

An alternative is to train models that use variation between regions within a genome (Figure 2.5 (a)). Splitting the sequence into windows centred on the trait of interest gives rise to tens of thousands of training examples for most molecular traits even when using a single individual. Even with large datasets, predicting molecular traits from DNA sequence is challenging due to multiple layers of abstraction between effect of individual DNA variants and the trait of interest, as well as the dependence of the molecular traits on a broad sequence context and interactions with distal regulatory elements.

The value of deep neural networks in this context is twofold. First, classical machine learning methods cannot operate on the sequence directly, and thus require predefining features that can be extracted from sequence based on prior knowledge (e.g. the presence of absence of single nucleotide variants (SNVs), k-mer frequencies, motif occurrences, conservation, known regulatory variants, or structural elements). Deep neural networks can help circumventing the manual extraction of features by learning them from data. Second, because of their representational richness, they can capture nonlinear dependencies in the sequence, interaction effects, and span wider sequence context at multiple genomic scales. Attesting to their utility, deep neural networks have been successfully applied to predict splicing activity [128, 235], specificities of DNA- and RNA binding proteins [5], or epigenetic marks and to study the effect of DNA sequence alterations [112, 247].

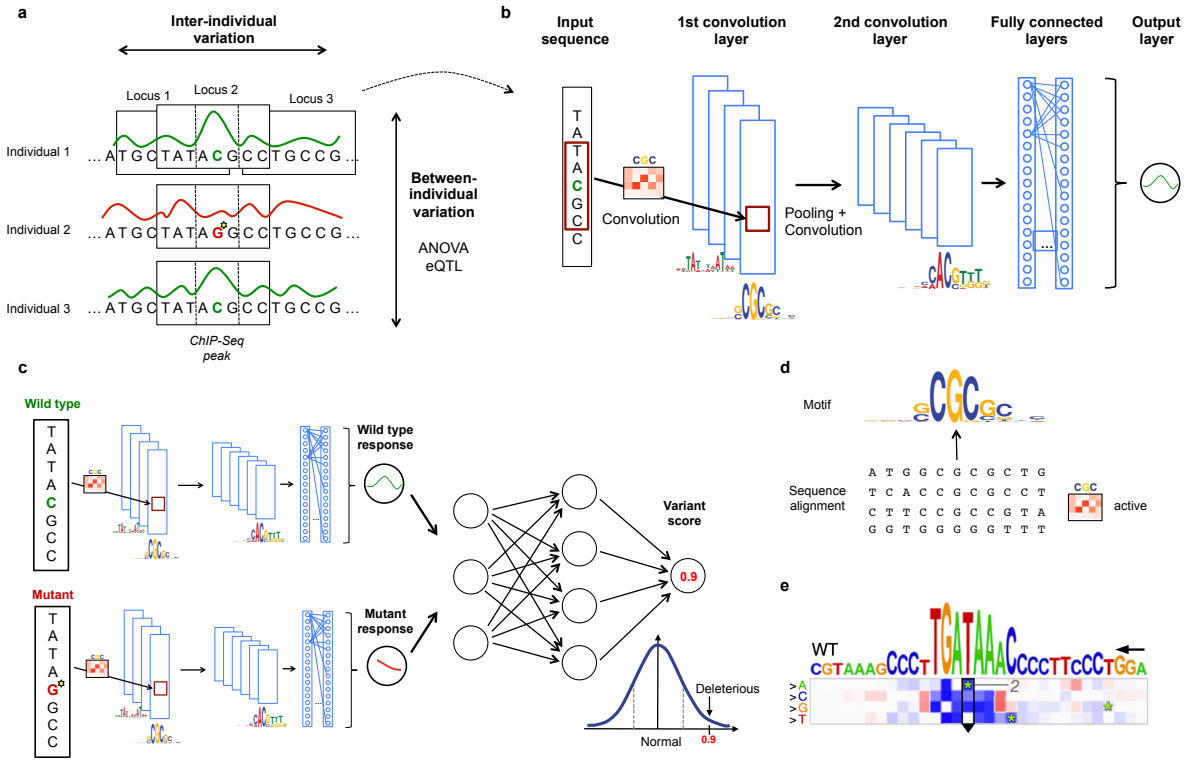


Figure 2.5 Principles of using neural networks for predicting molecular traits from DNA sequence. (a) DNA sequence and the molecular response variable along the genome for three individuals. Conventional approaches in regulatory genomics consider variations between individuals, whereas deep learning allows exploiting intra-individual variations by tiling the genome into sequence DNA windows centred on individual traits, resulting in large training datasets from a single sample. (b) One-dimensional convolutional neural network for predicting a molecular trait from the raw DNA sequence in a window. Filters of the first convolutional layer (example shown on the edge) scan for motifs in the input sequence. Subsequent pooling reduces the input dimension, and additional convolutional layers and can model interactions between motifs in the previous layer. (c) Response variable predicted by the neural network shown in (b) for a wild type and mutant sequence is used as input to an additional neural network that predicts a variant score and allows to discriminate normal from deleterious variants. (d) Visualization of a convolutional filter by aligning genetic sequences that maximally activate the filter and creating a sequence motif. (e) Mutation map of a sequence window. Rows correspond to the four possible base pair substitutions, columns to sequence positions. The predicted impact of any sequence change is colour coded. Letters on top denote the wild type sequence with the height of each nucleotide denoting the maximum effect across mutations (Figure panel adapted from Alipanahi et al. [5]).

2.3.1 Early applications of neural networks in regulatory genomics

The first successful applications of neural networks in regulatory genomics replaced a classical machine learning approach with a deep model, without changing the input features. For example, Xiong et al. [235] considered a fully connected feedforward neural network to predict the splicing activity of individual exons. The model was trained using more than 1000 pre-defined features extracted from the candidate exon and adjacent introns. Despite the relatively low number of 10700 training samples in combination with the model complexity, this method achieved substantially higher prediction accuracy of splicing activity compared to simpler approaches, and in particular was able to identify rare mutations implicated in splicing misregulation.

2.3.2 Convolutional designs

More recent work using convolutional neural networks (CNNs) allowed direct training on the DNA sequence, without the need to define features [5, 9, 112, 247]. The CNN architecture allows to greatly reduce the number of model parameters compared to a fully connected network by applying convolutional operations to only small regions of the input space and by sharing parameters between regions. The key advantage resulting from this approach is the ability to directly train the model on larger sequence windows (Section 2.2.2; Figure 2.5 (b); Figure 2.3).

Alipanahi et al. [5] considered convolutional network architectures to predict specificities of DNA- and RNA binding proteins. Their DeepBind model outperformed existing methods, was able to recover known and novel sequence motifs, and could quantify the effect of sequence alterations and identify functional SNVs. A key innovation that enabled training the model directly on the raw DNA sequence was the application of a one-dimensional convolutional layer. Intuitively, the neurons in the convolutional layer scan for motif sequences and combinations thereof, similar to conventional position-weight matrices [208]. The learning signal from deeper layers informs the convolutional layer which motifs are most relevant. The motifs recovered by the model can then be visualized as heatmaps or sequence logos (Figure 2.5 (d)).

2.3.3 In silico prediction of mutation effects

An important application of deep neural networks trained on the raw DNA sequence is to predict the effect of mutations in silico. Such model-based assessments of the effect of sequence changes complement methods based on QTL mapping, and can in particular help to uncover regulatory effects of rare SNVs or to fine-map likely causal genes. An intuitive approach for visualizing such predicted regulatory effects are mutation maps [5], whereby the effect of all possible mutations for a given input sequence is represented in a matrix view (Figure 2.5 (e)). The authors could further reliably identify deleterious SNVs, by training an additional neural network with predicted binding scores for a wild type and mutant sequence (Figure 2.5 (c)).

2.3.4 Joint prediction of multiple traits and further extensions

Following their initial successes, convolutional architectures have been extended and applied to a range of tasks in regulatory genomics. For example, Zhou and Troyanskaya [247] considered these architectures to predict chromatin marks from DNA sequence. The authors observed that the size of the input sequence window is a major determinant of model performance, where larger windows (now up to 1 kbp) coupled with multiple convolutional layers enabled capturing sequence features at different genomic length scales. A second innovation was to use neural network architectures with multiple output variables (so called multi-task neural networks), here to predict multiple chromatin states in parallel. Multi-task architectures allow learning shared features between outputs, thereby improving generalization performance, and markedly reducing the computational cost of model training compared to learning independent models for each trait [51].

In a similar vein, Kelley et al. [112] developed the open-source deep learning framework Basset, to predict DNase-I hypersensitivity across multiple cell types and to quantify the effect of SNVs on chromatin accessibility. Again, the model improved prediction performance compared to conventional methods and was able to retrieve both known and novel sequence motifs that are associated with DNase-I hypersensitivity. Most recently, Koh et al. [116] applied CNNs to de-noise genome-wide chromatin immunoprecipitation followed by sequencing data in order to obtain a more accurate prevalence estimate for different chromatin marks.

At present, CNNs are among the most widely used architectures to extract features from fixed sized DNA sequence windows. However, alternative architectures could also be considered. For example, RNNs are suited to model sequential data [135], and have been applied for modelling natural language and speech [41, 54, 75, 86, 213, 234], protein sequences [2, 199], clinical medical data [41], and to a limited extent DNA sequences [125]. RNNs are appealing for applications in regulatory genomics, because they allow modelling sequences of variable length, and to capture long-range interactions within the sequence and across multiple outputs. However, at present, RNNs are more difficult to train than CNNs, and additional work is needed to better understand the settings where one should be preferred over the other.

Complementary to supervised methods, unsupervised deep learning architectures learn low-dimensional feature representations from high-dimensional unlabelled data, similarly to classical principal components analysis or factor analysis, but using a non-linear model. Examples of such approaches are stacked autoencoders [223], restricted Boltzmann machines, and deep belief networks [88]. The learned features can be used to visualize data or as input for classical supervised learning tasks. For example, sparse autoencoders have been applied to classify cancer cases using gene-expression profiles [63], or to predict protein backbones [139]. Restricted Boltzmann machines can also be used for unsupervised pre-training of deep networks to subsequently train supervised models of protein secondary structures [200], disordered protein regions [60, 61], or amino-acid contacts [60]. Skip-gram neural networks have been applied to learn low-dimensional representations of protein sequences and improve protein classification [11]. In general, unsupervised models are a powerful approach if large quantities of unlabelled data are available to pre-train complex models. Once trained, these models can help to improve performance on classification tasks, for which smaller numbers of labelled examples are typically available.

2.4 Deep learning for biological image analysis

Historically, perhaps the most important successes of deep neural networks have been in image analysis. Deep architectures trained on millions of photographs can famously detect objects in pictures better than humans do [83]. All current state-of-the-art models in image classification, object detection, image retrieval, and semantic segmentation, make use of neural networks. The convolutional neural network (Section 2.2.2) is the most common network architecture for image analysis. Briefly, a CNN performs pattern matching (convolution) and aggregation

(pooling) operations. At a pixel level, the convolution operation scans the image with a given pattern, and calculates the strength of the match for every position. Pooling determines the presence of the pattern in a region, for example by calculating the maximum pattern match in smaller patches (max-pooling), thereby aggregating region information into a single number (Figure 2.3; Figure 2.6). The successive application of convolution and pooling operations is at the core of most network architectures used in image analysis.

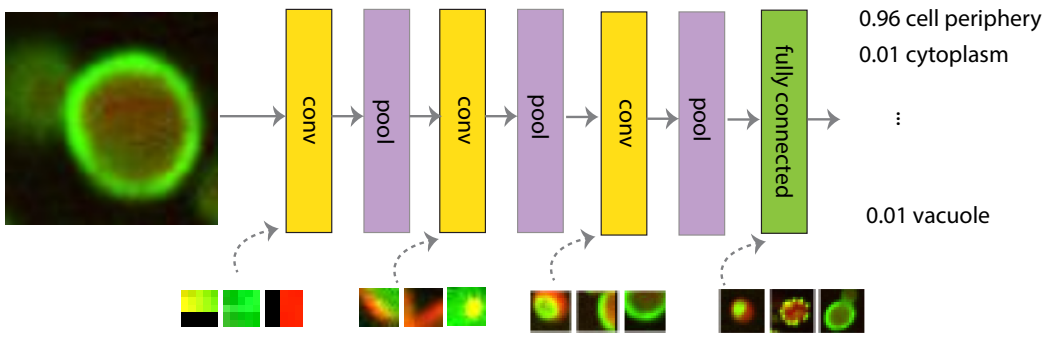


Figure 2.6 Convolution and pooling operators are stacked, thereby creating a deep network for image analysis. In standard applications, convolution layers are followed by a pooling layer (Section 2.2.2). In this example, the lowest level convolutional units operate on 3x3 patches, but deeper ones use and capture information from larger regions. These convolutional pattern-matching layers are followed by one or multiple fully connected layers to learn which features are most informative for classification. For each layer with learnable weights, three example images that maximize some neuron output are shown.

2.4.1 First applications in computational biology

The early applications of deep networks for biological images focused on pixel level tasks, with additional models building on the network outputs. For example, Ning et al. [163] applied convolutional neural networks in a study that predicted abnormal development in C. elegans embryo images. They trained a CNN on 40x40 pixel patches to classify the centre pixel to cell wall, cytoplasm, nucleus membrane, nucleus, or outside medium, using three convolutional and pooling layers, followed by a fully connected output layer. The model predictions were then fed into an energy-based model for further analysis. CNNs have outperformed standard methods, e.g. Markov random fields and conditional random fields [130] in such raw data analysis tasks, for example restoring noisy neural circuitry images [100].

Adding layers allows moving from clearing up pixel noise to modelling more abstract image features. Cireşan et al. [48] used five convolutional and pooling layers, followed by two fully connected layers, to find mitosis in breast histology images. This model won the mitosis detection challenge at the International Conference of Pattern Recognition 2012, outperforming competitors by a substantial margin. The same approach was also used to segment neuronal structures in electron microscopy images, classifying each pixel as membrane or non-membrane [47]. In these applications, while the CNNs were trained in an end-to-end manner, additional post-processing was required to obtain class probabilities from the outputs for new images.

Successive pooling operations lose information on localization, as only summaries are retained from larger and larger regions. To avoid this, skip links can be added to carry information from early, fine-grained layers forward to deeper ones. The currently best performing pixel level classification method for neuronal structures, U-Net [180]) employs an architecture in which neurons take inputs from lower layers to localize high resolution features, as well as to overcome the arbitrary choice of context size.

2.4.2 Analysis of whole cells, cell populations, and tissues

In many cases, pixel-level predictions are not required. For example, Xu et al. [238] classified colon histopathology images into cancerous and non-cancerous, finding that supervised feature learning with deep networks was superior to using hand-crafted features. Pärnamaa and Parts [166] used CNNs to classify pre-segmented image patches of individual yeast cells carrying a fluorescent protein to different subcellular localization patterns. Again, deep networks outperformed methods based on traditional features. Further, Kraus et al. [118] combined the segmentation and classification tasks into a single architecture that can be learned end-to-end, and applied the model to full resolution yeast microscopy images. This approach allowed classifying entire images without performing segmentation as a pre-processing step. CNNs have even been applied to count bacterial colonies in agar plates [67]. Since the early de-noising applications on the pixel level, the field has been moving towards end-to-end image analysis pipelines that make use of large bioimage datasets, and the representational power of CNNs.

2.4.3 Re-using trained models

Training convolutional neural networks requires large datasets. While biological data acquisition can be expensive, this does not mean that deep neural networks cannot be used when millions of images are not available. Regardless of image source, lower levels of the network tend to capture similar signal (edges, blobs) that are not specific to the training data and the application, but instead recur in perceptual tasks in general. Thus, convolutional neural networks can re-use pictures from a similar domain to help with learning, or even be pre-trained on other data, thereby requiring fewer images to fine-tune the model for the task of interest. Indeed, Donahue et al. [57] showed that features learned from millions of images to classify objects, can successfully be used in image retrieval, detection or classification on new domains where only hundreds of images are labelled. The effectiveness of such an approach depends on the similarity between the training data and the new domain [240].

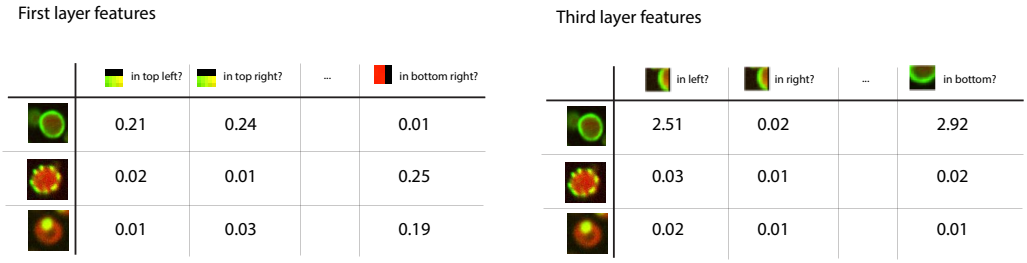


Figure 2.7 A pre-trained network can be used as a generic feature extractor. Feeding input into the first layer (left) gives a low-level feature representation in terms of patterns (left to right) present in smaller patches in every cell (top to bottom). Neuron activations extracted from deeper layers (right) give rise to more abstract features that capture information from a larger segment of the image.

The concept of transferring model parameters has also been successful in bioimage analysis. For example, Zhang et al. [244] showed that features learned from natural images can be transferred to biological data, improving the prediction of Drosophila melanogaster developmental stages from in situ hybridization images. The model was first pre-trained on data from the ImageNet [183], an open corpus of more than one million diverse images, to extract rich features at different scales. Xie et al. [233] further used synthetic images to train a CNN for automatic cell counting in microscopy images. We expect that network repositories that host pre-trained models will emerge for biological image analysis; such efforts already exist for general image processing tasks (see learning section below). These trained models could be

downloaded and used as feature extractors (Figure 2.7), or further fine-tuned and adapted to a particular task on small-scale data.

2.4.4 Interpreting and visualizing convolutional networks

Convolutional neural networks have been successful across many domains. In interpreting their performance, it is useful to understand the features they capture.

Visualizing input weights. One way to understand what a particular neuron represents is to look for inputs that maximally activate it. Under some mathematical constraints, these patterns are proportional to the incoming weights. Krizhevsky et al. [119] visualized weights in the first convolutional layer, and found that these maximally activating patterns correspond to colour blobs, edges at different orientations, and Gabor-like filters (Figure 2.7). Gabor filters are widely used pre-defined features in image analysis; neural networks rediscover them in a data driven way as a useful component of the image model. Higher layer weights can be visualized as well, but as the inputs are not pixels, their weights are more difficult to interpret.

Finding images that maximize neuron activity. To understand the deeper layers in terms of input pixels, Girshick et al. [72] and Simonyan et al. [194] generated images that maximize the output of individual neurons. While this approach yields no explicit representation, it can provide an overview of the type of features that differentiate images with large neuron activity from all others. Such visualizations tend to show that second layer features combine edges from the first layer, thereby detecting corners and angles, deeper layer neurons activate for specific object parts (e.g. noses, eyes), and the deepest layers detect whole objects (e.g. faces, cars). It is complicated to hand-engineer features that look specifically for noses, eyes, or faces, but neural networks can learn these features solely from input-output examples.

Hiding important image parts. To understand which image parts are important for determining the value of each feature, Zeiler and Fergus [241] occluded images with smaller grey boxes. The parts that are most influential will drastically change the feature value when occluded. In a similar vein, Simonyan et al. [194] and Springenberg et al. [201] visualised which individual pixels make the most difference in the feature, and Bach et al. [12] developed pixel relevance for individual classification decisions in a more general framework. This information can also be used for object localisation or segmentation, as the sensitive image pixels usually correctly

correspond to the true object. Kraus et al. [118] used this idea to effectively localize cells in large microscopy images.

Visualising similar inputs in two dimensions. Visualising the CNN representations can help gauge what inputs get mapped to similar feature vectors, and hence understand what the model has learned. Donahue et al. [57] projected CNN features into two dimensions to show that each subsequent layer transforms data to be more and more separable by a linear classifier. In general, different CNN visualisation methods show that higher layer features are more specific to the learning task, while low-level features tend to capture general aspects of images, such as edges, corners, etc.

2.5 Off-the-shelf tools and practical considerations

2.5.1 Deep learning frameworks

Deep learning frameworks have been developed to easily build neural networks from existing modules on a high level. The most popular ones are Caffe [103], Theano [15], Torch7 [49], and TensorFlow [1] (Table 2.1), which differ in modularity, ease of use and the way models are defined and trained.

Caffe [103] is developed by the Berkeley Vision and Learning Center and is written in C++. The network architecture is specified in a configuration file and models can be trained and used via command line, without writing code at all. Additionally, Python and MATLAB interfaces are available. Caffe offers one of the most efficient implementations for CNNs and provides multiple pre-trained models for image recognition, make it well suited for computer vision tasks. As a downside, custom models need to be implemented in C++, which can be difficult. Additionally, Caffe is not optimized for recurrent architectures.

Theano [15] is developed and maintained by the University of Montreal and written in Python and C++. Model definitions follow a declarative instead of an imperative programing paradigm, which means that the user specifics what needs to be done, not in which order. A neural network is declared as a computational graph, which is then compiled to native code and executed. This design allows Theano to optimize computational steps and to automatically derive gradients—one of its main strengths. Consequently, Theano is well suited for building

	Caffee	**Theano**	**Torch7**	**Tensorflow**
Core language	C++	Python, C++	LuaJIT	C++
Interfaces	Python, Matlab	Python	C	Python, R
Wrappers		Lasagne, Keras, sklearn-theano		Keras, Pretty Tensor, Scikit Flow
Programming paradigm	Imperative	Declarative	Imperative	Declarative
Well suited for	CNNs, Reusing existing models, Computer vision	Custom models, RNNs	Custom models, CNNs, Reusing existing models	Custom models, Parallelization, RNNs

Table 2.1 Overview of existing deep learning frameworks, comparing four widely used software solutions.

custom models and offers particularly efficient implementations for RNNs. Software wrappers such as Keras[1] or Lasagne[2] provide additional abstraction and allow building networks from existing components, and reuising pre-trained networks. The major drawback of Theano are frequently long compile times when building larger models.

Torch7 [49] was initially developed at the University of New York and is based on the scripting language LuaJIT. Networks can be easily built by stacking existing modules and are not compiled, hence making it more suited for fast prototyping than Theano. Torch7 offers an efficient CNN implementation and access to a range of pre-trained models. A possible downside is the need of the user to be familiar with the LuaJIT scripting language. Also, LuaJIT is less suited for building custom recurrent networks.

TensorFlow [1] is the most recent deep learning framework developed by Google. The software is written in C++ and offers interfaces to Python. Similar to Theano, a neural network is declared as a computational graph, which is optimized during compilation. However, the shorter compile time makes it more suited for prototyping. A key strength of TensorFlow is native support for parallelization across different devices, including central processing units (CPUs) and graphics processing units (GPUs), and compute nodes on a cluster. The accompanying tool TensorBoard allows to conveniently visualize networks in a web browser and to monitor training progress, e.g. learning curves or parameter updates. At present, TensorFlow provides the most efficient

[1] https://github.com/fchollet/keras
[2] https://github.com/Lasagne/Lasagne

implementation for RNNs. The software is recent, and under active development, hence only few pre-trained models are currently available.

2.5.2 Data preparation

Training data are the key for every machine learning applications. Since more data with informative features usually result in better performance, effort should be spent on collecting, labelling, cleaning, and normalizing data.

Required dataset sizes

Most of the successful applications of deep learning have been in supervised learning settings, where sufficient labelled training samples are available to fit complex models. As a rule of thumb, the number of training samples should be at least as high as the number of model parameters, although special architectures and model regularization (see below) can help to avoid overfitting if training data are scarce [19].

Central problems in regulatory genomics, e.g. predicting molecular traits from genotype, are limited in the number of training instances; hundreds to at most tens of thousands of training examples are typical. The strategy of considering a sequence windows centred on the trait of interest (e.g. splice site, transcription factor binding site, or epigenetic marks; Figure 2.5 (a)) is now a widely used approach and helps increasing the number of input-output pairs from a single individual.

In image analysis, data can be abundant, but manually curated and labelled training examples are typically difficult to obtain. In such instances, the training set can be augmented by scaling, rotating, or cropping the existing images, an approach that also enhances robustness [119]. Another strategy is to reuse a network that was pre-trained on a large dataset for image recognition (e.g. AlexNet [119], VGG [193], GoogleNet [216], or ResNet [83]), and to fine-tune its parameters on the data set of interest (e.g. microscopy images for a particular segmentation task). Such an approach exploits that different data sets share important characteristics and features, such as edges or curves, which can be transferred between them. Caffe, Lasagne, Torch, and to a limited extend TensorFlow provide repositories with pre-trained models.

Partitioning data into training, validation, and test set

Machine learning models need to be trained, selected, and tested on independent data sets to avoid over-fitting and assure that the model will generalize to unseen data. Holdout validation, partitioning the data into a training, validation, and test set, is the standard for deep neural networks (Figure 2.8 (c)). The training set is used to learn models with different hyperparameters, which are then assessed on the validation set. The model with best performance, e.g. prediction accuracy or mean squared error, is selected, and further evaluated on the test set to quantify the performance on unseen data and for comparison to other methods. Typical dataset proportions are 60% for training, 10% for validation, and 30% for model testing. If the dataset is small, k-fold cross validation or bootstrapping can be used instead [82].

Normalization of raw data

Appropriate choices for data normalization can help to accelerate training and the identification of a good local minimum.

Categorical features such as DNA nucleotides first need to be encoded numerically. They are typically represented as binary vectors with all but one entry set to zero, which indicates the category (one-hot coding). For example, DNA nucleotides (categories) are commonly encoded as A = (1 0 0 0), G = (0 1 0 0), C = (0 0 1 0), and T = (0 0 0 1) (Figure 2.8 (a)). A DNA sequence can then be represented as a binary string by concatenating the encoding nucleotides, and treating each nucleotide as an independent input feature of a feed forward neural network. In a CNN, the four bits of each encoded base are commonly considered analogously to colour channels of an image to preserve the entity of a nucleotide.

Numerical features are typically zero-centred by subtracting their mean value. Image pixels are usually not zero-centred individually, but jointly by subtracting the mean pixel intensity per colour channel [110]. An additional common normalization step is to standardize features to unit variance. Whiting can be used to decorrelate features (Figure 2.8 (b)), but can be computationally involved, since it requires computing the feature covariance matrix [82]. If the distribution of features is skewed due to few extreme values, log transformations or similar processing steps may be appropriate. Validation and test data need to be normalized consistently with the training data. For example, features of the validation data need to be zero-centred by subtracting the mean computed on the training data, not on the validation data.

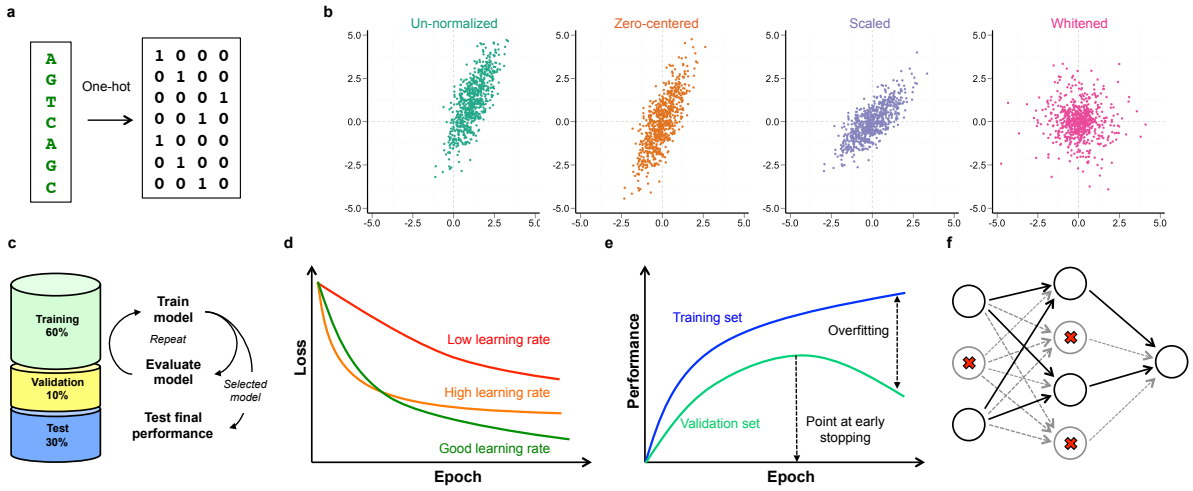


Figure 2.8 Data normalization and pre-processing for deep neural networks. (a) DNA sequence one-hot encoded as binary vectors using codes A = 1 0 0 0, G = 0 1 0 0, C = 0 0 1 0, and T = 0 0 0 1. (b) Continuous data (green) after zero-centring (orange), scaling to unit variance (blue), and whiting (purple). (c) Holdout validation partitions the full data set randomly into training (≈60%), validation (≈10%) and test set (≈30%). Models are trained with different hyperparameters on the training set, from which the model with the highest performance on the validation set is selected. The generalization performance of the model is assessed and compared with other machine learning methods on the test set. (d) The shape of the learning curve indicates if the learning rate is too low (red, shallow decay), too high (orange, steep decay followed by saturation), or appropriate for a particular learning task (green, gradual decay). (e) Large differences in the model performance on the training set (blue) and validation set (green) indicates overfitting. Stopping the training as soon as the validation set performance starts to drop (early stopping) can prevent overfitting. (f) Illustration of the dropout regularization. Shown is a feed forward neural network after randomly dropping out neurons (crossed out), which reduces the sensitivity of neurons to neurons in the previous layer due to non-existent inputs (greyed edges).

2.5.3 Model building

Choice of model architecture

After preparing the data, design choices about the model architectures need to be made. The default architecture is a feedforward neural network with fully connected hidden layers, which is an appropriate starting point for many problems. Convolutional architectures are well suited for multi- and high-dimensional data, such as two-dimensional images or abundant genomic data. Recurrent neural networks can capture long-range dependencies in sequential data of varying lengths, such as text, protein, or DNA sequences. More sophisticated models can be built by combining different architectures. To describe the content of an image, for example, a CNN can be combined with an RNN, where the CNN encodes the image and the RNN generates the corresponding image description [224, 236]. Most deep learning frameworks provide modules for different architectures and their combinations.

Determining the number of neurons in a network

The optimal number of hidden layers and hidden units is problem dependent and should be optimized on a validation set. One common heuristic is to maximize the number of layers and units without overfitting the data. More layers and units increase the number of representable functions and local optima, and empirical evidence show that it makes finding a good local optimum less sensitive to weight initialization [52].

Model training

The goal of model training is to find parameters w that minimize an objective function $L(w)$, which measures the fit between the predictions the model parameterized by w and the actual observations. The most common objective functions are the cross-entropy for classification and mean-squared error for regression. Minimizing $L(w)$ is challenging since it is high-dimensional and non-convex (Figure 2.8; Figure 2.2).

Stochastic Gradient Descent

Stochastic gradient descent is widely used to train deep models. Starting from an initial set of parameters w_0, the gradient Δw of L with respect to w is computed for a random batch of only few, e.g. 128, training samples. Δw points to the direction of steepest descent, towards which w is updated with step size eta, the learning rate (Figure 2.2 (c)). At each step, the parameters are updated into the direction of steepest descent until a minimum is reached, analogously to a ball running down a hill to a valley [19]. The training performance strongly depends on parameter initialization, learning rate, and batch size.

Parameter initialization

In general, model parameters should be initialized randomly to avoid local optima determined by a fixed initialization. Starting points for model parameters can be sampled independently from a normal distribution with small variance, or more commonly from a normal distribution with its variance scaled inversely by the number of hidden units in the input layer [73, 84].

Learning rate and batch size

The learning rate and batch size of stochastic gradient descent need to be chosen with care, since they can strongly impact training speed and model performance. Different learning rates are usually explored on a logarithmic scale such as 0.1, 0.01, or 0.001, with 0.01 as recommended default value [19]. A batch size of 128 training samples is suitable for most applications. The batch size can be increased to speed up training, or decreased to reduce memory usage, which can be important for training complex models on memory-limited GPUs. The optimum learning rate and batch size are connected, with larger batch sizes typically requiring smaller learning rates.

Learning rate decay

The learning rate can be gradually reduced during training, which is based on the idea that larger steps may be helpful in early training stages in order to overcome possible local optima, whereas smaller step sizes allowing exploring narrow parameter regions of the loss function in

advanced stages of training. Common approaches include to linearly reduce the learning rate by a constant factor such as 0.5 after the validation loss stops improving, or exponentially after every training iteration or epoch [19, 70].

Momentum

Vanilla stochastic gradient descent can be extended by 'momentum', which usually improves training [211]. Instead of updating the current parameter vector w_t at time t by the gradient vector Δw_{t+1} directly, a fraction of the previous update is added to the current one. With momentum rate ν, weights are updated by a momentum vector This approach can help to take larger steps in directions where gradients point consistently, and therefore speed up the convergence. The momentum rate ν can be set between $[0;1]$, and a typical value is 0.9. Nesterov momentum [160, 161] is a special form of the same concept, which sometimes provides additional advantages.

Per-parameter adaptive learning rate methods

To reduce the sensitivity to the specific choice of the learning rate, adaptive learning rate methods, such as RMSprop, Adagrad [58], and Adam [114], have been developed that one appropriately adapt the learning rate per parameter during. The most recent method, Adam, combines the strengths of previous methods RMSprop and Adagrad, and is generally recommended for many applications.

Batch normalization

Batch normalization [99] is a recently described approach to reduce the dependency of training to the parameter initialization, speed up training, and reduce overfitting. It is easy to implement, has marginal additional compute costs, and has hence become common practice. Batch normalization zero centres and normalizes data not only at the input layer, but also at hidden layers before the activation function. This approach allows using higher learning rates and hence also accelerates training.

Analyzing the learning curve

To validate the learning process, the loss should be monitored as a function of the number of training epochs, i.e. the number times the full training set has been traversed (Figure 2.8 (d)). If the learning curve decreases slowly, the learning rate may be too small and should be increased. If the loss decreases steeply at the beginning but saturates quickly, the learning rate may to too high. Extreme learning rates can result in an increasing or fluctuating learning curve [19, 110].

Monitoring training and validation performance

In parallel to the training loss, it is recommended to monitor the target performance such as the accuracy for both the training and validation set during training (Figure 2.8 (e)). A low or decreasing validation performance relative to the training performance indicates overfitting [19, 110].

Avoiding overfitting

Deep neural networks are notoriously difficult to train, and overfitting to data is a major challenge, since they are non-linear and have many parameters. Overfitting results from a too complex model relative to the size of the training set, and can thus be reduced by decreasing the model complexity, e.g. the number of hidden layers and units, or by increasing the size of the training set, e.g. via data augmentation (see above). The following training guidelines can help to avoid overfitting.

Dropout [203] is the most common regularization technique and often one of the key ingredients to train deep models. Here, the activation of some neurons is randomly set to zero ('dropped out') during training in each forward pass, which intuitively results in an ensemble of different networks whose predictions are averaged (Figure 2.8 (e)). The dropout rate corresponds to the probability that a neuron is dropped out, where 0.5 is a sensible default value. In addition to dropping out hidden units, input units can be dropped, however usually at a lower rate. Dropout is often combined with regularizing the magnitude or parameter values by the L2 norm, and less commonly the L1 norm.

Another popular regularization method is 'early stopping'. Here, training is stopped as soon as the validation performance starts to saturate or deteriorate, and the parameters with the best performance on the validation set chosen.

Layer-wise pre-training [20, 184] should be considered if the model overfits despite the mentioned regularization techniques. Instead of training the entire network at once, layers are first pre-trained unsupervised using autoencoders or restricted Boltzmann machines. Afterwards, the entire network is fine-tuned using the actual supervised learning objective.

Hyperparameter optimization

Table 2.2 summarizes recommendations and starting points for the most common hyperparameters, excluding architecture dependent hyperparameters such as the size and number of filters of a CNN. Since the best hyperparameter configuration is data and application dependent, models with different configurations should be trained and their performance be evaluated on a validation set. As the number of configurations grows exponentially with the number of hyperparameters, trying all of them is impossible in practice [19]. It is therefore recommended to optimize the most important hyperparameters such as the learning rate, batch size, or length of convolutional filters independently via line search, i.e. trying different values while keeping all other hyperparameters constant. The refined hyperparameter space can then be further explored by random sampling, and settings with the best performance on the validation set are chosen. Frameworks such as Spearmint [198], Hyperopt [23], or SMAC [98] allow to automatically explore the hyperparameter space using Bayesian optimization. However, although conceptually more powerful, they are at present more difficult to apply and parallelize than random sampling.

Training on GPUs

Training neural networks is more time consuming compared to shallow models, and can take hours, days, or even weeks, depending on the size of training set and model architecture. Training on GPUs can considerably reduce the training time (commonly by ten fold or more) and is therefore crucial for evaluating multiple models efficiently. The reason for this speedup is that learning deep networks requires large numbers of matrix multiplications, which can be parallelized efficiently on GPUs. All state of the art deep learning frameworks provide

Name	Range	Default value
Learning rate	0.1, 0.01, 0.001, 0.0001	0.01
Batch size	64, 128, 256	128
Momentum rate	0.8, 0.9, 0.95	0.9
Weight initialization	Normal, Uniform, Glorot uniform	Glorot uniform
Per-parameter adaptive learning rate methods	RMSprop, Adagrad, Adadelta, Adam	Adam
Batch normalization	yes, no	yes
Learning rate decay	None, linear, exponential	Linear (rate 0.5)
Activation function	Sigmoid, Tanh, ReLU, Softmax	ReLU
Dropout rate	0.1, 0.25, 0.5, 0.75	0.5
l1, l2 regularization	0, 0.01, 0.001	

Table 2.2 Important hyperparameters of deep neural networks and recommended default values.

support to train models on either CPUs or GPUs without requiring any knowledge about GPU programming. On desktop machines, the local GPU card can often be used if the framework supports the specific brand. Alternatively, commercial providers provide GPU cloud compute clusters.

Pitfalls

No single method is universally applicable, and the choice of whether and how to use deep learning approaches will be problem specific. Conventional analysis approaches will remain valid and have advantages when data are scarce or if the aim is to assess statistical significance, which is currently difficult using deep learning methods. Another limitation of deep learning is the increased training complexity, which applies both to model design and the required compute environment.

2.6 Discussion

Deep learning methods are a powerful complement to classical machine learning tools and other analysis strategies. Already, these approaches have found use in a number of applications in computational biology, including regulatory genomics and image analysis. The first publicly

available software frameworks have helped to reduce the overhead of model development, and provided a rich, accessible toolbox to practitioners. We expect that continued improvement of software infrastructure will make deep learning applicable to a growing range of biological problems.

Chapter 3

Protocols and analysis of single-cell DNA methylation data

Our understanding of DNA methylation has been revolutionized by the development of BS-seq, which offers single-cytosine resolution and absolute quantification of 5mC genome-wide. Recent advances have demonstrated the power of single-cell sequencing to deconvolve mixed cell populations [100, 55, 140]. Incorporating epigenetic information into this single-cell arsenal will provide insights into epigenetic heterogeneity and broaden our understanding of gene regulation.

In the first section of this chapter, we will describe the scBS-seq protocol for genome-wide profiling of DNA methylation in single cells, a statistical method for quantifying methylation variability between cells, and applications to mouse ESCs. The presented work is based on Smallwood et al. [196], which was joint work of Sebastien Smallwood, Heather Lee, Christof Angermueller, Felix Krueger, Heba Saadeh, Julian Peat, Simon Andrews, Oliver Stegle, and Wolf Reik.

> Individual contributions: Sebastien Smallwood and Heather Lee designed the study, prepared scBS-seq libraries, analysed data and wrote the manuscript. Felix Krueger, Heba Saadeh, and Sebastien Smallwood performed sequence mapping and analysed data. Julian Peat contributed to technical developments. Christof Angermueller and Oliver Stegle analysed the data.

In the second section, we will describe the scM&T-seq protocol for parallel profiling of DNA methylation and gene expression in single cells, methods for quantifying associations between DNA methylation and gene expression, and applications to mouse ESCs. The presented work is based on Angermueller et al. [7], which was joint work of Christof Angermueller, Stephen Clark, Heather Lee, Iain Macaulay, Mabel Teng, Tim Xiaoming Hu, Felix Krueger, Sebastien Smallwood, Chris Ponting, Thierry Voet, Gavin Kelsey, Oliver Stegle, and Wolf Reik.

Individual contributions: Christof Angermueller performed all statistical analyses of the data. Heather Lee, Iain Macaulay, Stephen Clark, and Sebastien Smallwood developed the protocol and performed experiments. Heather Lee, Iain Macaulay, Christof Angermueller, Stephen Clark, Oliver Stegle, Wolf Reik, and Chris Ponting interpreted the results. Mabel Teng contributed to method development. Tim Xiaoming Hu processed RNA-seq data. Felix Krueger processed BS-seq data. Wolf Reik, Gavin Kelsey, Iain Macaula, and Thierry Voet contributed protocols and reagents. Heather Lee, Iain Macaulay, Wolf Reik, and Thierry Voet conceived the project.

3.1 Estimating DNA methylation variability in embryonic stem cells

Several protocols have been developed for profiling average DNA methylation levels of in bulk populations of cells (Section 1.3). However, bulk profiling protocols are unable to directly estimate methylation heterogeneity between single cells, which is critical for studying embryonic development, cancer progression, and pluripotent stem cells. In the following, we will describe scBS-seq, an accurate and reproducible method for profiling DNA methylation in single cells, and a statistical model for estimating methylation heterogeneity in cell populations across the entire genome.

3.1.1 Single-cell bisulfite sequencing protocol

In commonly used BS-seq protocols, sequencing adaptors are ligated to fragmented DNA before bisulfite conversion, which results in a loss of information owing to DNA degradation by the bisulfite treatment. To minimize DNA loss from single cells, we developed a modification of post-bisulfite adaptor tagging [155]. In scBS-seq, bisulfite treatment is performed first,

which results in simultaneous DNA fragmentation and conversion of unmethylated cytosines to thymine (Figure 3.1). Then, synthesis of complementary strands is primed using oligonuleotides containing Illumina adaptor sequences and a 3' stretch of nine random nucleotides. This step is performed five times to maximize the number of tagged DNA strands and to generate multiple copies of each fragment. After capturing the tagged strands, a second adaptor is similarly integrated, and PCR amplification is performed with indexed primers.

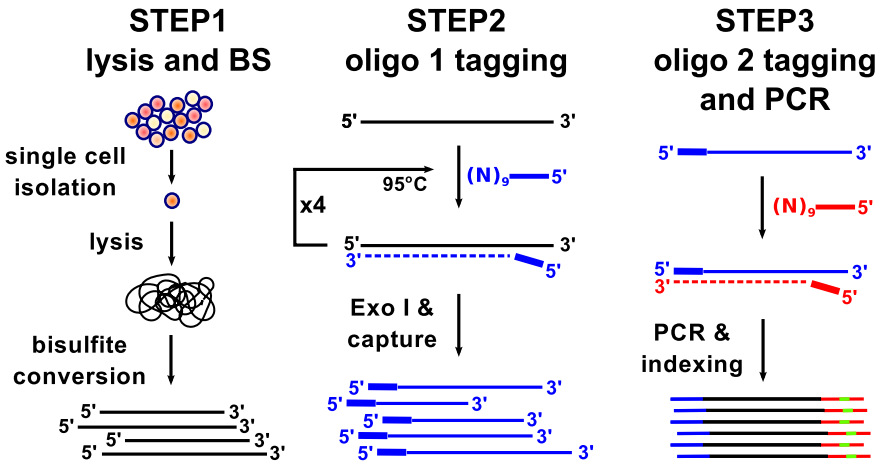


Figure 3.1 scBS-seq profiling protocol. scBS-seq library preparation consists of isolating and lysing single cells before bisulfite conversion ('BS'); performing five rounds of random priming and extension using oligo 1 (which carries the first sequencing adaptor) and purifying synthesized fragments; and performing a second random priming and extension step using oligo 2 (which carries the second sequencing adaptor) before amplifying the resulting fragments.

We assessed scBS-seq on ovulated metaphase II oocytes (MIIs) and mouse ESCs cultured either in 2i medium or serum conditions. MIIs are a suited model for technical assessment as they: (i) can be individually hand- picked to ensure that only one cell is processed; (ii) represent a highly homogeneous population, which allows discrimination between technical and biological variability; and (iii) present a distinct DNA methylome comprising large-scale hypermethylated and hypomethylated domains [191]. ESCs grown in serum conditions are characterized by a high heterogeneity in DNA methylation and gene expression and hence suited for the estimating of intercellular heterogenetiy [68]. We used ESCs grown in serum ('serum ESCs') and ESCs grown in 2i medium ('2i ESCs') to determine whether scBS-seq can reveal DNA methylation heterogeneity in single cells.

We sequenced 12 MII, 12 2i ESC, 20 serum ESC, and 7 negative controls using scBS-seq, and their bulk cell counterparts using BS-seq. We obtained the methylation state of on average 3.7 million CpG dinucleotides (CpGs; range, 1.8 M–7.7 M) corresponding to 17.7% of all

CpGs (range, 8.5–36.2%; Figure 3.2 (a)). A higher CpG coverage can be obtained by deeper sequencing. To validate this, we sequenced two MII libraries close to saturation, which resulted in 1.5-fold and 1.9-fold more CpGs captured. Altogether, we obtained up to 10.1 M CpGs, corresponding to a CpG coverage of 48.4%.

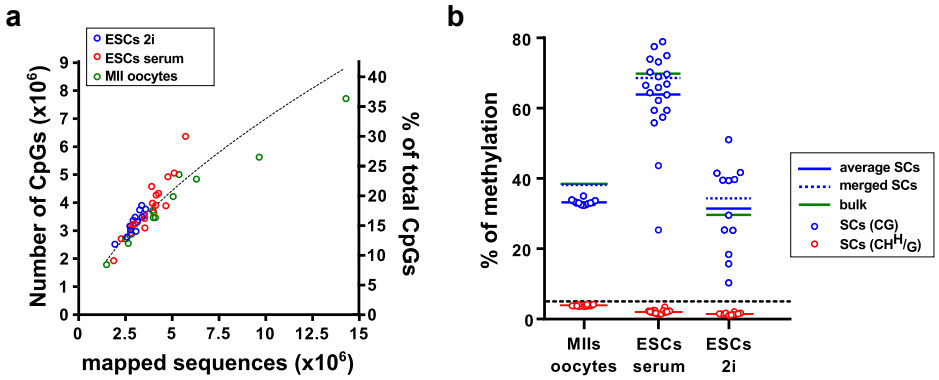


Figure 3.2 scBS-seq mapping efficiency and mean methylation levels. (a) Number of CpGs obtained by scBS-seq as a function of mapped sequences. (b) Global mean DNA methylation levels in CpG (CG) and non-CpG (CHH/G) context for single cells (SCs), in silico–merged, and bulk samples.

Next, we investigated the reproducibility and accuracy of scBS-seq. CpG sites in MIIs were overwhelmingly called methylated or unmethylated, which is consistent with a highly digitized output from single cells (Figure A.1). As expected, global methylation of MIIs was highly homogeneous ($33.1 \pm 0.8\%$) and 2i ESCs were hypomethylated compared to serum ESCs13. Yet both 2i ESCs and serum ESCs exhibited 5mC heterogeneity (serum, $63.9 \pm 12.4\%$; 2i medium, $31.3 \pm 12.6\%$; Figure 3.2 (b)). We determined the average pairwise concordance between individual CpGs across single oocyte libraries, which was 87.6% genome- wide (range, 85.3–88.9%) and 95.7% in unmethylated CGIs, a highly homogeneous genomic feature (Figure 3.3 (a)). CpG concordance in ESCs was lower (serum, 72.7%; 2i medium, 69.8%), which reflected the heterogeneity of these cells (Figure 3.3 (a)). At two kilo base pair (kbp) resolution, we observed high correlation between individual MIIs (Pearson's $r = 0.92$), and between individual MIIs and bulk (Pearson's $r = 0.95$) (Figure 3.3 (b)). We could largely reproduce the entire bulk methylation profile of oocytes using only 12 single cells (Figure 3.3 (b)). This capability is particularly beneficial for analyses of homogeneous cell populations and makes scBS-seq an important tool to investigate the 5mC landscape in very rare material.

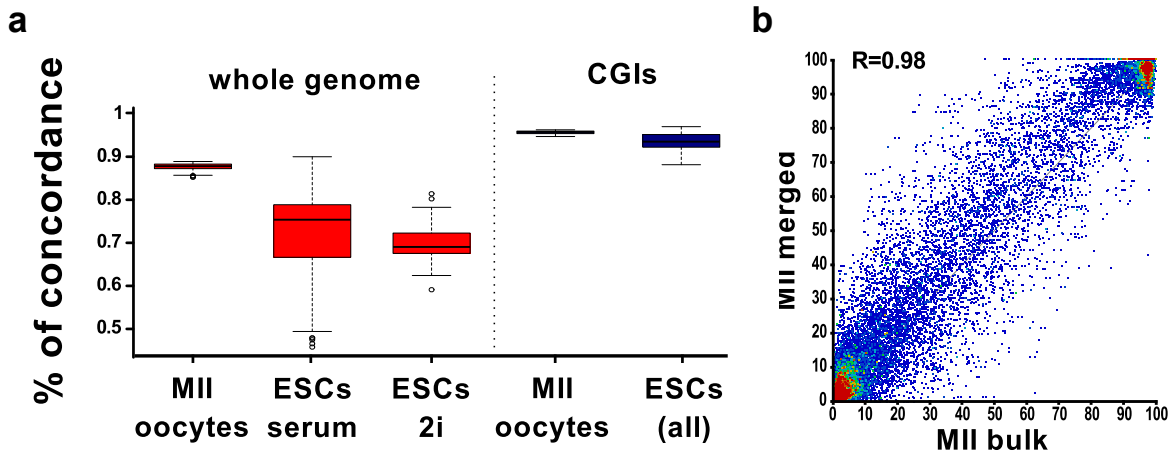


Figure 3.3 CpG concordance and reproduction of bulk data. (a) Pairwise analysis of CpG concordance genome-wide and in unmethylated CGIs. Boxplots represent the interquartile range, with the median; whiskers correspond to 1.5 times the interquartile range. (b) Pairwise correlation CpG methylation levels between MII-merged and MII-bulk data.

3.1.2 Method for estimating DNA methylation variability

The majority of CpG sites in single cells are either methylated or unmethylated. An exception is hemimethylation, where the cytosine is methylated on only one DNA strand. Consistent with this, scBS-seq called CpG sites overwhelmingly methylated or unmethylated (Figure A.1). Since hemimethylation is rare and currently hard to detect by scBS-seq owing to low CpG coverage, we did not consider hemimethylation in our analysis. Instead, we modelled the methylation state of CpG sites in single-cells as a Bernoulli variable, and represented methylation profiles as a binary matrix (Figure A.1). As a consequence of the limited CpG coverage of only $\approx 10-30\%$, the methylation state of most CpG sites is unobserved, which renders downstream analyses challenging. We therefore developed a method that aggregates information from adjacent CpG sites and estimates mean methylation levels as well as cell-to-cell heterogeneity for windows instead of single CpG sites. Our method yields uncertainty estimates, which is critical in regions of low CpG coverage. Our method is further computational efficient, thereby applicable genome-wide on over 20 million CpG sites.

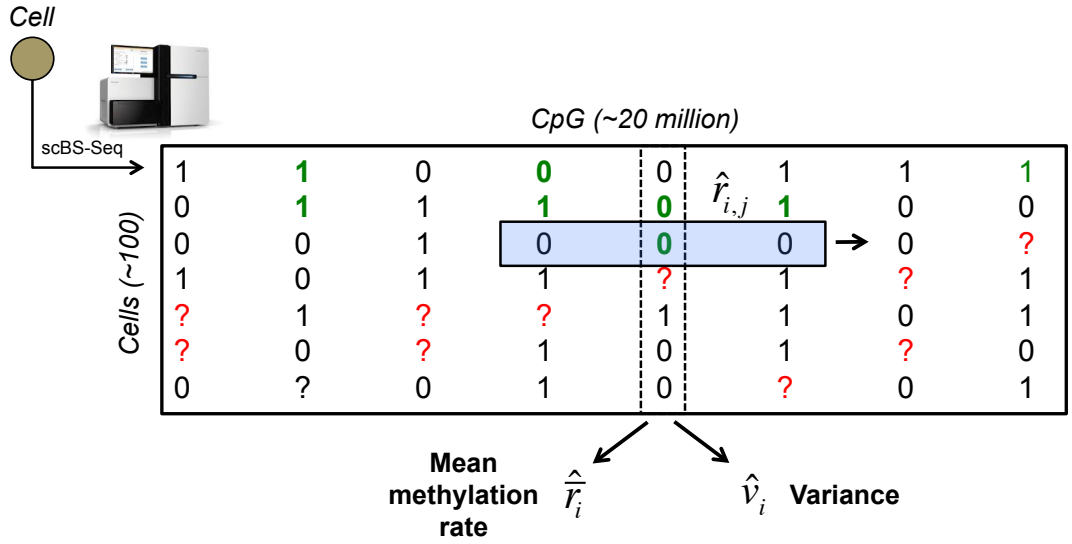


Figure 3.4 Representation of single-cell methylation data. Binary matrix M with rows corresponding to cells and columns to CpG sites. $M_{i,j}$ represent the methylation state of CpG site i in cell j, which is one if the CpG site is methylated and zero otherwise. Question marks denote sites with unobserved methylation state. A sliding window (blue) is used to first estimate the methylation rate $\hat{r}_{i,j}$ (green) for each cell and window, and the mean methylation rate $\hat{\bar{r}}_i$ and variance $\hat{v}_i$ across cells afterwards.

Estimating cell-specific methylation rates

To increase the coverage across cells, we employed a sliding window approach, which is conceptually similar to approaches that have been used for bulk BS-Seq [31, 131]. With window size $w = 3000$ bp and step size 600 bp, we computed for each cell j the sum of methylated $c^+_{i,j}$ and unmethylated $c^-_{i,j}$ read counts in window i:

$$s^+_{i,j} = \sum_{k=-w/2}^{+w/2} c^+_{i+k,j} \qquad s^-_{i,j} = \sum_{k=-w/2}^{+w/2} c^-_{i+k,j} \tag{3.1}$$

To estimate methylation rates, we modelled the sum $S^+_{i,j}$ of methylated read counts as a Binomial random variable with methylation rate $r_{i,j}$:

$$S^+_{i,j} \sim \mathrm{Bin}(s^+_{i,j} + s^-_{i,j}, r_{i,j}) \tag{3.2}$$

Assuming $r_{i,j} \sim \text{Beta}(1,1)$, leads to the maximum a posteriori estimator $\hat{r}_{i,j}$ of the methylation rate in cell j and window i:

$$\hat{r}_{i,j} = \frac{s^+_{i,j} + 1}{s^+_{i,j} + s^-_{i,j} + 2} \tag{3.3}$$

To account for the limited CpG coverage of scBS-seq (Section 3.1.1), we quantified prediction uncertainty by approximating the standard error of the rate estimator using the Wald method [206]:

$$\text{SE}[\hat{r}_{i,j}]^2 = \frac{\hat{r}_{i,j}(1-\hat{r}_{i,j})}{s^+_{i,j} + s^-_{i,j}} \tag{3.4}$$

Estimating cell-to-cell variability

We used the estimated cell-specific methylation rates $\hat{r}_{i,j}$ to estimate the mean methylation rate and variance across all cells. We modelled the mean methylation rate r_i in window i as a Gaussian random variable with mean $\bar{r}_i$ and variance v_i:

$$r_i \sim N(\bar{r}_i, v_i) \tag{3.5}$$

To account for differences in the standard errors $\text{SE}[\hat{r}_{i,j}]$, we weighted cell j and position i by $w_{i,j} = \text{SE}[\hat{r}_{i,j}]^{-2}$, and used the weighted maximum likelihood estimator

$$\hat{\bar{r}}_i = \frac{1}{\sum_j w_{i,j}} \sum_j w_{i,j} \hat{r}_{i,j} \tag{3.6}$$

to estimate $\bar{r}_i$. Its standard error is given by

$$\text{SE}[\hat{\bar{r}}_i]^2 = \frac{1}{\sum_j w_{i,j}}. \tag{3.7}$$

The maximum likelihood estimator of the variance v_i is

$$\hat{v}_i = \frac{\sum_j w_{i,j}}{\left(\sum_j w_{i,j}\right)^2 - \sum_j w^2_{i,j}} \sum_j w_{i,j} \left(\hat{r}_{i,j} - \hat{\bar{r}}_i\right)^2, \tag{3.8}$$

which is the unbiased weighted cell-to-cell variance. The chi-squared confidence interval of the variance estimator with significance level α is

$$[\hat{v}_i^l, \hat{v}_i^u] = [\frac{n_i \hat{v}_i}{\chi^2_{1-\frac{\alpha}{2},n_i}}, \frac{n_i \hat{v}_i}{\chi^2_{\frac{\alpha}{2},n_i}}]. \tag{3.9}$$

Here, χ^2_{p,n_i} is the p-quantile of the chi-squared distribution with n_i degrees for freedom, where n_i is the sum of cell weights:

$$n_i^2 = \frac{\sum_j w_{i,j}}{\left(\sum_j w_{i,j}\right)^2 - \sum_j w_{i,j}^2} \tag{3.10}$$

To determine highly variable methylated sites, we ranked these by the lower bound $\hat{v}_i^l$ of the chi-squared confidence interval and defined the top k sites as the most variable sites. This approach is selecting sites with large estimates of cell-to-cell variance while penalizing for uncertainty of these estimates due to low CpG coverage.

Cluster analysis

To cluster cells and sites, we considered a complete linkage clustering, and employed the weighted Euclidean norm as distance measure for comparing cell j with cell j':

$$d(j,j') = \sqrt{\sum_{i=1}^{d} w_i^{j,j'} \left(\hat{r}_{i,j} - \hat{r}_{i,j'}\right)^2} \tag{3.11}$$

We defined the weight $w_i^{j,j'}$ at position i as

$$w_i^{j,j'} \propto \sqrt{w_{i,j} w_{i,j'}}, \tag{3.12}$$

and normalized weights to sum up to the total number of positions d. This distance measure places most emphasis on positions that are covered by both cells.

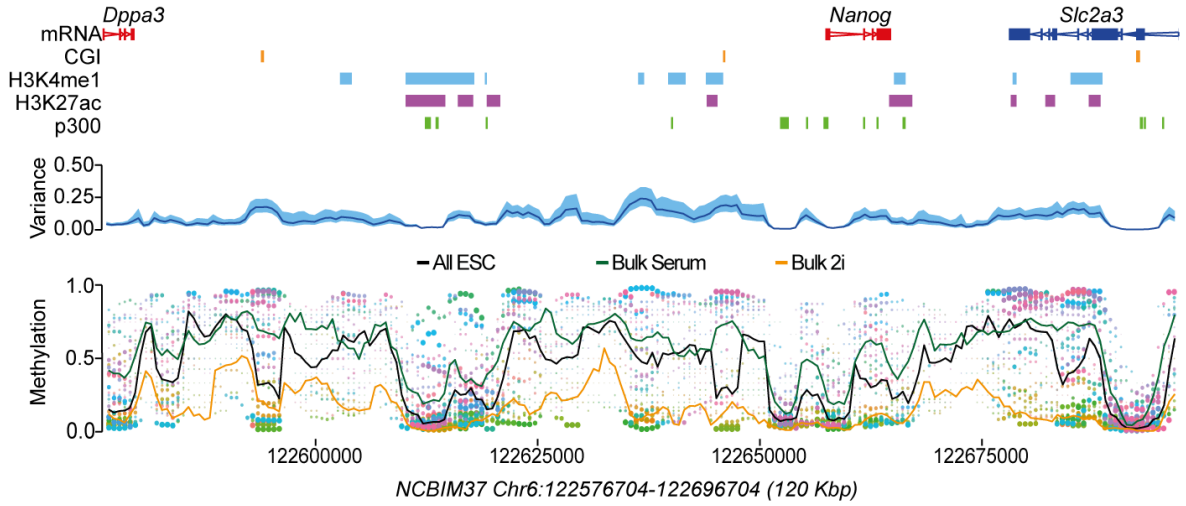


Figure 3.5 Estimated DNA methylation rates using a sliding window in an example region containing the Nanog locus with some annotated features. Each single ESC is represented by a different color (bottom), and dot size is the inverse of estimation error. Mean methylation rate estimates across cells (black line, bottom) and cell-to-cell variance (blue line, middle; 95% confidence interval in light blue) are shown. Methylation rates for 'bulk serum' (green line) and 'bulk 2i' (orange line) are superimposed (bottom).

3.1.3 Methylation variability in different genomic contexts

We applied our method to estimate methylation rates in each ESC genome as well as the mean methylation rate and variance across all ESCs (Figure 3.5). By using our method for weighted clustering (Equation 3.11), we identified two distinct clusters that represented the majority of 2i ESCs and serum ESCs (Figure 3.7 (a)). Two outlier cells from the serum condition clustered with 2i ESCs, which implies that serum cultures contain '2i-like' ESCs and demonstrates the ability of scBS-seq to identify rare cell types in populations. To examine 5mC heterogeneity in ESCs in greater detail, we ranked sites by the estimated cell-to-cell variance (Equation 3.8) and repeated the cluster analysis for the 300 most variable sites (Figure 3.6). The structure of the resulting clusters was broadly similar to what was observed based on genome-wide analysis, and all 300 variable sites followed the global trend of being more highly methylated in serum than 2i ESCs with high similarity between sites (Figure 3.2 (b); Figure 3.6). This observation is consistent with the genome-wide hypomethylation observed in ESCs grown in 2i medium [68] and indicates that a major determinant of ESC heterogeneity is global methylation.

Our method also identified sites whose methylation varied more than the genome average, including sites with marked heterogeneity even among cells from the same growth condition

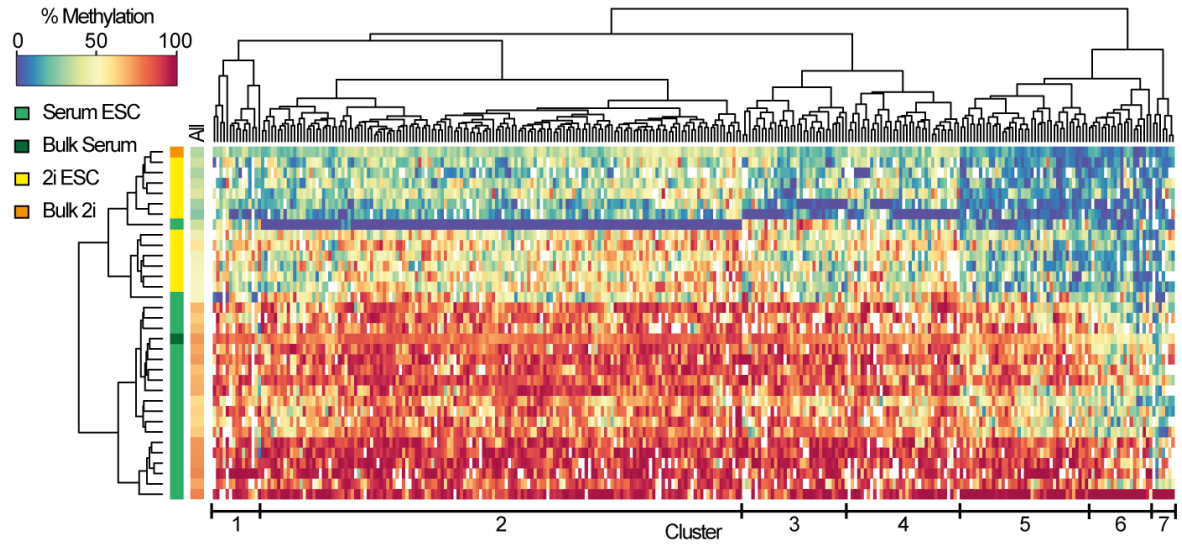


Figure 3.6 Heatmap for methylation rates of the 300 most variable sites among single-cell ESC samples. Cluster dendrograms for samples (left) and sites (top) are shown. The genome-wide average methylation rate is displayed in the left track ('all'). The main clusters of variable sites are indicated at the bottom.

(e.g. clusters 5 and 6 in serum ESCs; Figure 3.6). Regions containing H3K4me1 and H3K27ac, marks associated with active enhancers, had the greatest variance in 5mC, whereas CGIs and intracisternal A-particle repeats had lower variance than the genome average (Figure 3.7 (b)). These findings are consistent with observations that distal regulatory elements are differentially methylated between tissues and throughout development [204, 250, 91]

3.2 Estimating associations between DNA methylation and gene expression

We have shown that scBS-seq enables the exploration of intercellular heterogeneity in DNA methylation genome-wide. To further study the relationship between heterogeneity in DNA methylation and gene expression, we developed scM&T-seq, a protocol for parallel profiling of DNA methylation and gene expression in single cells. In combination with scM&T-seq, we developed methods for estimating associations between DNA methylation and gene expression in single cells, which will be described in the following.

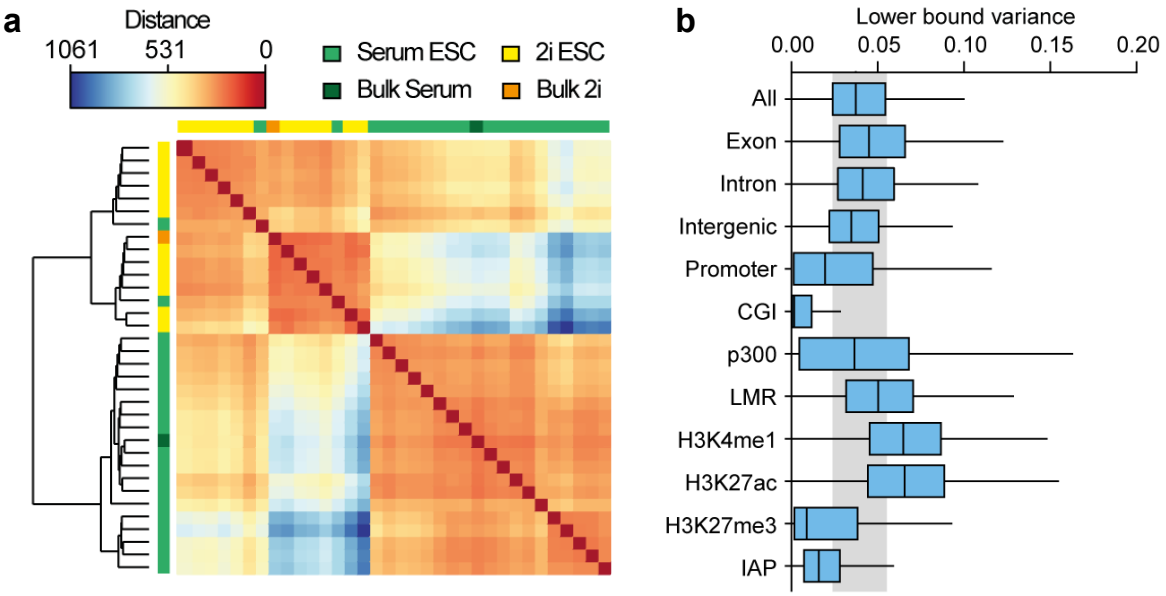


Figure 3.7 Genome-wide clustering and estimated variance in genomic contexts. (a) Genome-wide cluster dendrogram and distance matrix for all ESCs and bulk samples based on estimated methylation rates. Distance refers to the weighted Euclidean norm between estimated rates. (b) Variance of sites located in different genomic contexts. Boxes represent interquartile range with the median; whiskers correspond to 1.5 times the interquartile range. The shaded gray region indicates the interquartile range for all genome-wide sites.

3.2.1 Parallel single-cell DNA methylation and gene expression profiling

Macaulay et al. [141] have recently developed G&T-seq, a method for parallel genome and transcriptome sequencing within single cells. Importantly, G&T-seq utilizes physical separation of RNA and DNA allowing bisulfite conversion of DNA without affecting the transcriptome. We now apply scBS-seq to genomic DNA purified according to the G&T-seq protocol. Our scM&T-seq protocol consists of isolating single cells, followed by physical separating DNA and RNA using the G&T protocol (Figure 3.8). DNA is bisfulfite treated and sequenced using scBS-seq to quantify DNA methylation of individual cells, and RNA is sequenced using scRNA-seq [101] to quantify gene expression.

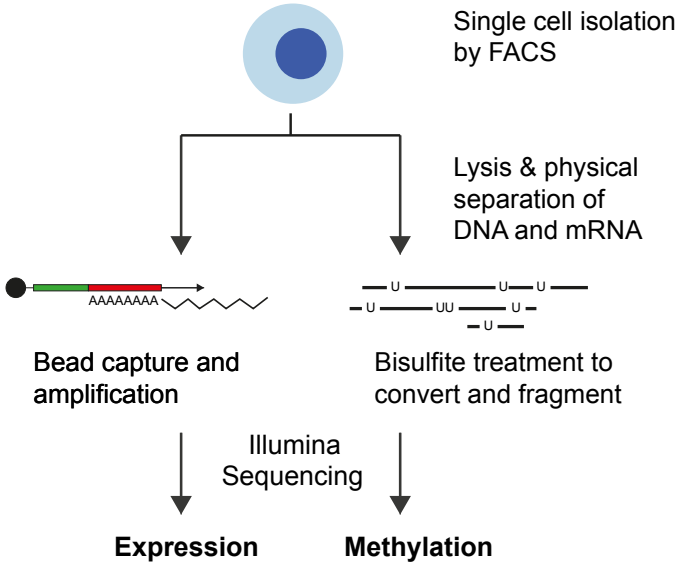


Figure 3.8 Schematic overview of the scM&T-seq protocol. Single-cells are isolated by FACS sorting and extracted DNA separated from RNA using G&T-seq. DNA is treated with bisulfite and sequenced using scBS-seq to quantify DNA methylation; RNA is amplified and sequences to quantify gene expression.

We evaluated scM&T-seq on mouse ESCs. In the presence of serum, these cells are characterized by high transcriptional and epigenetic heterogeneity. To investigate the link between epigenetic and transcriptional heterogeneity in ESCs, we performed scM&T-seq on 76 individual serum ESCs and 16 ESCs grown in '2i' media, which induces genome-wide DNA hypomethylation.

To assess the quality of the scBS-seq data, we compared the resulting single-cell methylomes with the 20 serum and 12 2i ESCs that we profiled with stand-alone scBS-seq (Section 3.1.1). The genome-wide CpG coverage at matched sequencing depth was consistent across scM&T-seq and scBS-seq (Figure 3.9 (a)) and we found that scM&T-seq covered a large proportion of sites in different genomic contexts with sufficient frequency to enable the analysis of

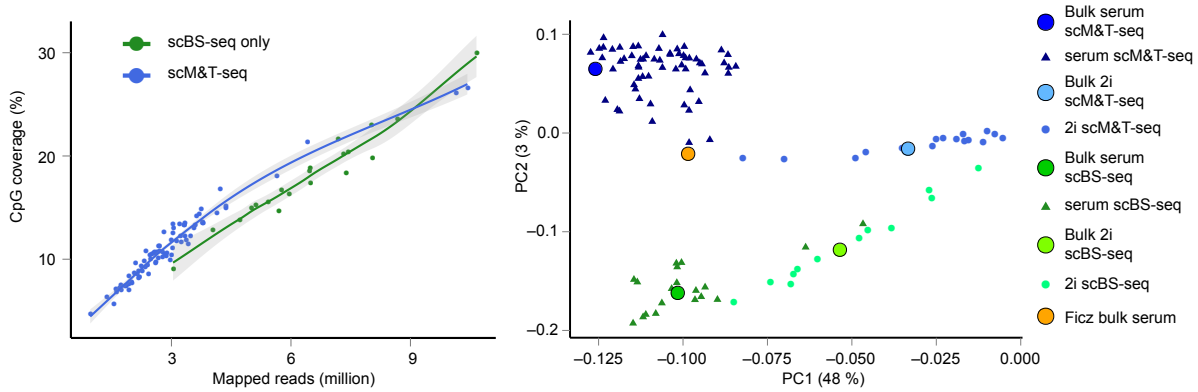


Figure 3.9 Quality control of the scM&T-seq protocol. (a) CpG coverage of single cells as a function of the number of mapped sequencing reads. Green: stand-alone scBS-seq, Blue: scM&T-seq. (b) Joint principal component analysis of the methylomes (gene body methylation) of 61 serum ESCs (dark blue) and 16 2i ESCs (light blue) obtained using scM&T-seq, as well as 20 serum ESCs (green) and 12 2i ESCs (yellow) sequenced using stand-alone scBS-seq. The solid circles correspond to synthetic bulk datasets form the same cells. For comparison, we also included a bulk serum ESC DNA methylation dataset [68] (orange). Cell type explained a substantially larger proportion of variance (PC1, 48%) than protocol (PC2, 3%).

epigenome heterogeneity across cells. As additional validation, we assessed the discrimination of serum and 2i ESCs by both stand-alone scBS-seq and scM&T-seq, finding a similar degree of separation that was consistent with bulk datasets published previously [68] (Figure 3.9 (b)), with similar conclusions when using a joint hierarchical clustering across all cells (Figure A.4). Notably, the difference between protocols and biological batches had a substantially smaller effect (PC2, 3% variance) than cell type differences (PC1, 48% variance), and by combining data across cells, we found that both protocols yield genome-wide methylation profiles that accurately recapitulate bulk methylation profiles in the same cell type (Figure A.5). Finally, we compared estimates of methylation heterogeneity in different genomic contexts, again finding good agreement between protocols (Figure A.2). Taken together, these analyses provide confidence that the parallel scM&T-seq method yields results that are in agreement with data from stand-alone scBS-seq.

Macaulay et al. [141] has previously shown that the scRNA-seq data generated by the G&T-seq method is of similar quality to that generated using the scRNA-seq protocol. We obtained an average of 2.7 million scRNA-seq reads per cell, and we excluded cells with fewer than 2 million mapped reads. In ESCs that met scRNA-seq quality-control criteria, we detected transcripts from between 4000 and 8000 genes exceeding one transcript per million, consistent with previous measurements made using the method (Figure A.3).

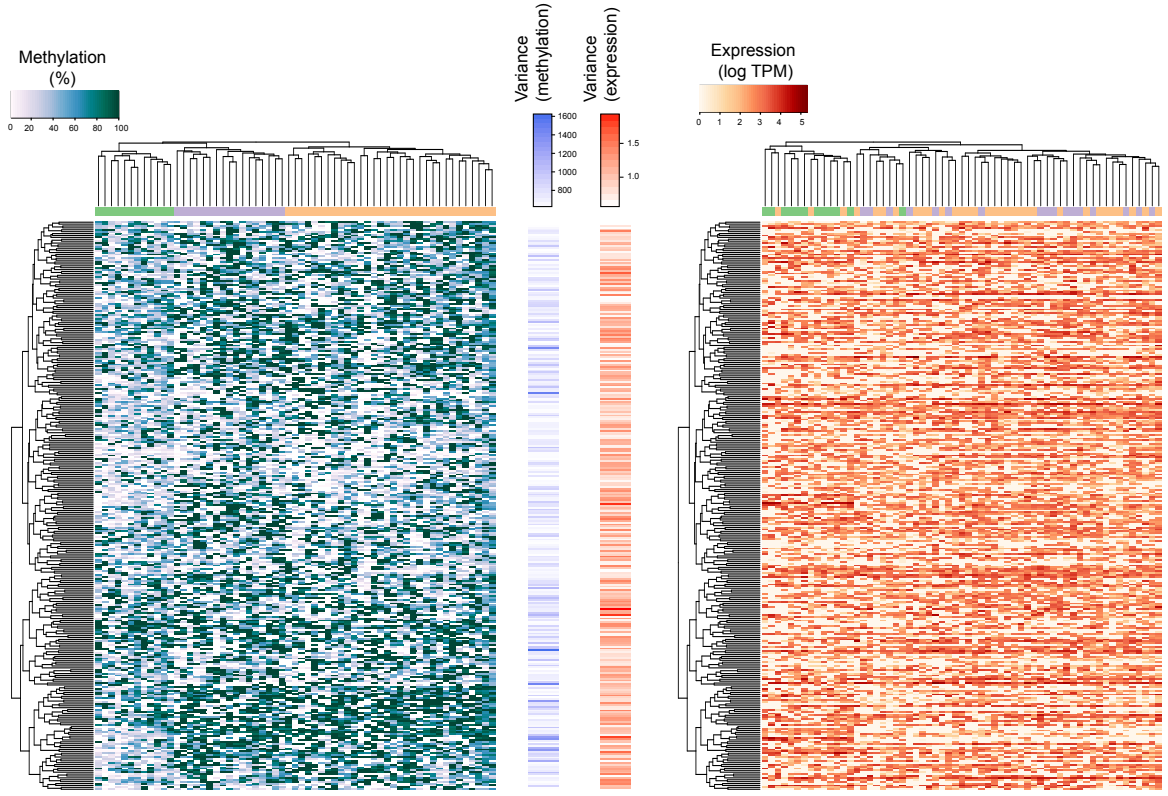


Figure 3.10 Clustering analysis of transcriptome and methylome data from 61 serum ESCs, considering gene body methylation (left) and gene expression (right) for the 300 most heterogeneous genes (based on gene body methylation). The order of genes was taken from an individual clustering analysis based on gene body methylation whereas cells were clustered separately either using DNA methylation or expression data, and coloured by methylation cluster. The bar plots in the center show the heterogeneity in DNA methylation (left) and gene expression (right).

For subsequent analyses, we focused on serum ESCs only since transcription and DNA methylation are uncoupled in 2i ESCs [68, 80]. A comparison of the principal components derived from gene body methylation and gene expression revealed associations between some factors of variations of both data modalities (Figure A.6; Figure A.7). However, a hierarchical clustering analysis of gene body methylation and gene expression for the 300 most variable genes revealed distinct clustering of cells when using either source of information (Figure 3.10). This suggests that global methylome and transcriptome profiles yield complementary but distinct aspects of cell state. This is also consistent with previous observations that the transcriptome and methylome are partially uncoupled in serum ESCs [68].

3.2.2 Methods for estimating associations between DNA methylation and gene expression

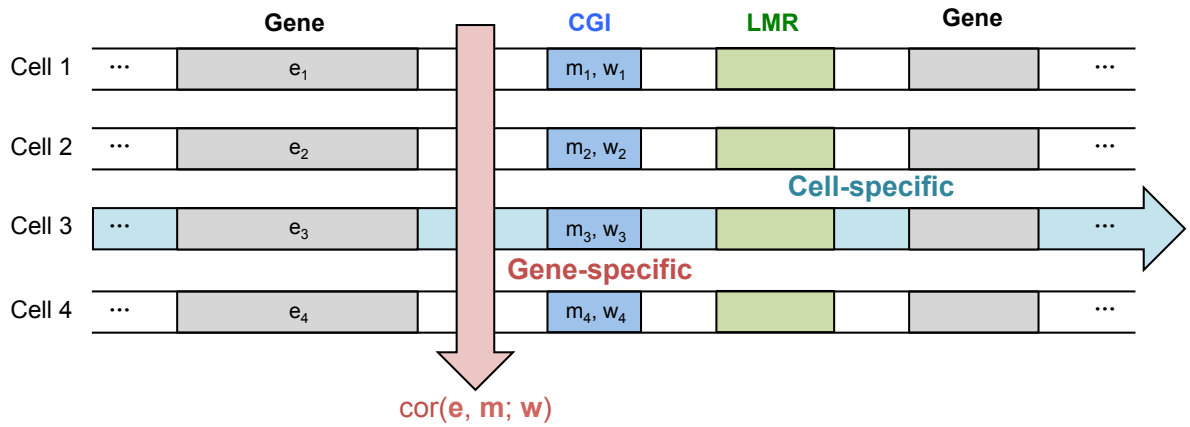


Figure 3.11 Schematic representation of cell-specific and gene-specific correlation analysis between methylome and transcriptome. Cell-specific analysis is performed for a single cell or a bulk population of cells across multiple genes. Gene-specific analysis is performed for a single gene across multiple cells. The vector e represents the expression rates of the considered gene for all cells, m the mean methylation rates in the corresponding genomic context, and w the number of covered CpG sites within that context. Associations were estimated by the weighted Pearson correlation $\mathrm{cor}(e, m; w)$ to account for differences in CpG coverage between cells.

Previous studies on data from bulk sequencing protocols estimated correlations between methylome and transcriptome in a bulk population of cells across multiple genes (Figure 3.11). In contrast, scM&T-seq separates individual cells and hence enables estimating associations for a particular gene across multiple cells. Let e be a vector with expression rates of cells for a particular gene, m be methylation rates of the associated region, and w be weights corresponding

to the number of covered CpGs sites within the region. Then we estimated associations using weighted Pearson correlation $\mathrm{cor}(e,m;w)$ between gene-expression e and methylation m:

$$\mathrm{cor}(e,m;w) = \frac{\mathrm{cov}(e,m;w)}{\sqrt{\mathrm{cov}(e,e;w)\,\mathrm{cov}(m,m;w)}} \tag{3.13}$$

Here, $\mathrm{cov}(x,y;w)$ is the weighted covariance

$$\mathrm{cov}(x,y;w) = \sum_i \frac{w_i(x_i - m(x;w))(y_i - m(y;w))}{\sum_i w_i}, \tag{3.14}$$

and $m(x;w)$ the weighted arithmetic mean:

$$m(x;w) = \frac{\sum_i x_i w_i}{\sum_i w_i} \tag{3.15}$$

By computing weighted correlations, we accounted for differences in CpG coverage between cells. We considered all possible relationships between genes and methylated regions within 10 kbp of the gene (upstream and downstream of gene start or stop). We performed two-sided Student's t-tests to test for non-zero correlations, and adjusted p-values for multiple testing for each context using the Benjamini-Hochberg procedure.

We considered gene expression levels on a logarithmic scale using log10 normalized TPM counts. We estimated binary single-base pair CpG methylation states by the ratio of methylated read counts to total read counts. We further estimated the methylation rate in different genomic contexts, such as gene body, promoter, or enhancer annotations, as the mean CpG methylation rate within the region defined by the context.

We discarded genes with low expression levels or low expression and methylation variability between cells, following the rational of independent filtering [35]. First, a minimum expression level (at least 10 TPM counts) in at least 10% of all cells was required. From these, the 7500 most variable genes were considered for analysis. Second, methylated regions were required to be covered by at least one read in at least 50% of all cells.

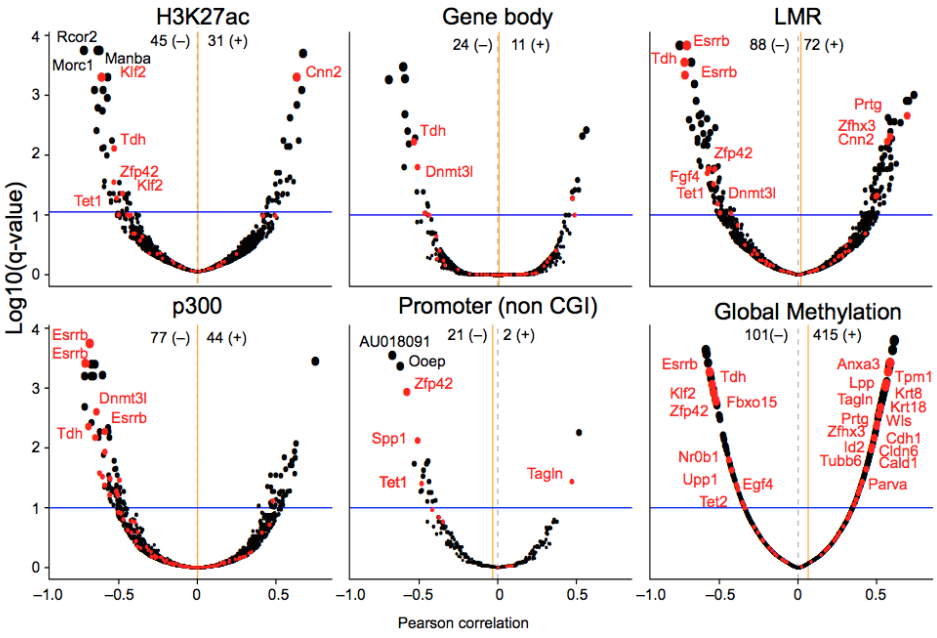


Figure 3.12 Volcano plots of correlation coefficients (Pearson's r^2) from association tests between gene expression heterogeneity of individual genes and DNA methylation heterogeneity in alternative genomic contexts. Shown is the correlation coefficient for every gene (x-axis) versus the adjusted p-value (using Benjamini-Hochberg correction; y-axis). The size of dots corresponds to the adjusted p-value. A set of 86 known pluripotency and differentiation genes are highlighted in red. The blue horizontal line corresponds to the FDR $= 0.1$ significance threshold. The total number of significant positive (+) and negative (–) correlations (FDR < 0.1) for each annotation is shown in the header of each panel. The orange vertical bar corresponds to the average correlation coefficient across all genes for a given context.

3.2.3 Associations between DNA methylation and gene expression in different genomic contexts

Using weighted Pearson correlation (Equation 3.13), we tested for associations between expression of individual genes and DNA methylation variation at several genomic contexts. We identified a total of 1493 associations (FDR < 0.1; Figure 3.12), which were robust when using a bootstrapping approach to subsample the set of cells. We found both positive and negative associations, highlighting the complexity of interactions between the methylome and transcriptome [56]. While methylation of non-CGI promoters is known to be associated with transcriptional repression, the role of enhancer methylation is less clear. Accordingly, negative correlations between DNA methylation and gene expression were predominant for non-CGI promoters, whereas positive and negative associations were more balanced in distal regulatory elements including LMRs (Figure 3.12; Figure A.8; Figure A.9). Interestingly, associated genes were enriched for known pluripotency and differentiation genes [117] (FDR < 0.01, Fisher's exact test). Our results provide the first evidence that heterogeneous methylation of distal regulatory elements, e.g. LMRs, accompanies heterogeneous expression of key pluripotency factors in stem cell populations [127].

As an example, figure 3.13 shows the association map of Esrrb–a known key regulator gene in pluripotency networks [165]. Expression of Esrrb negatively correlated with the methylation of several LMR and p300 sites overlapping 'super enhancers' in the genomic neighbourhood [232], providing evidence for the regulatory importance of Esrrb. We also found 516 genes whose expression correlated with the overall methylation level (FDR < 0.1), indicating substantial links between transcriptional heterogeneity and global methylation levels (Figure 3.12).

In addition to between-cell analyses, scM&T-seq can be used to correlate the methylome and transcriptome between genes in individual cells (Figure 3.11; Figure 3.14), analogously to studies in cell populations. We found that correlation between methylation and gene expression varied substantially between cells but was consistent in direction with matched RNA-seq and BS-seq data from a population of cells [68]. Again, this attests to scM&T-seq being sufficiently accurate to reliably study epigenome-transcriptome linkages. Our results also point to the possibility of heterogeneity between cells in the degree of coupling between the methylome and the transcriptome. Although we have ruled out obvious confounding factors, such as average methylation rate and sequence coverage (Figure A.10; Figure A.11), more data will be required to understand possible technical components in these linkages.

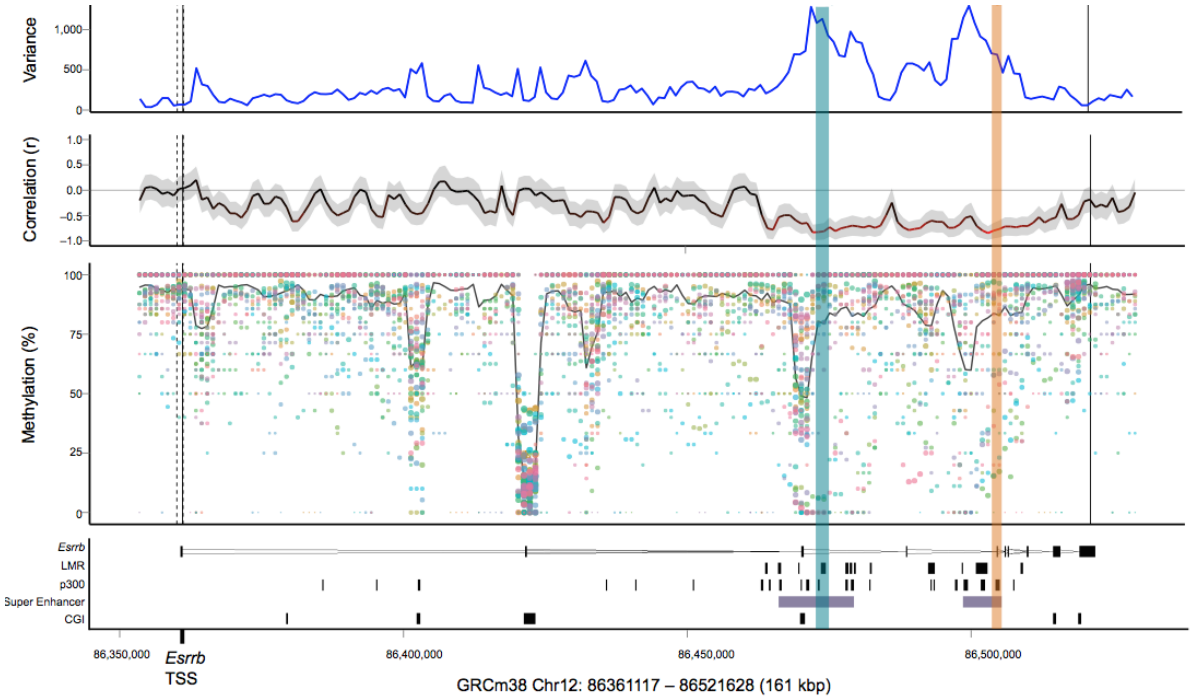


Figure 3.13 Representative zoom-in view for the gene Esrrb. From bottom to top, shown is: the annotation of the Esrrb locus with LMR, p300, super enhancer and CGI sites indicated; the estimated methylation rate of 3 *kbp* windows for each cell with the size of dots representing the CpG coverage and the solid line indicating the weighted mean methylation rate across all cells; the correlation between the methylation rate and Esrrb expression for each region coloured by the strength of the correlation and with the shaded area corresponding to the 95% confidence interval of the correlation coefficient; and the estimated weighted DNA methylation variance between cells. The vertical bars denote the location of a p300 (yellow) and LMR (blue) region, in which DNA methylation is significantly associated with gene expression.

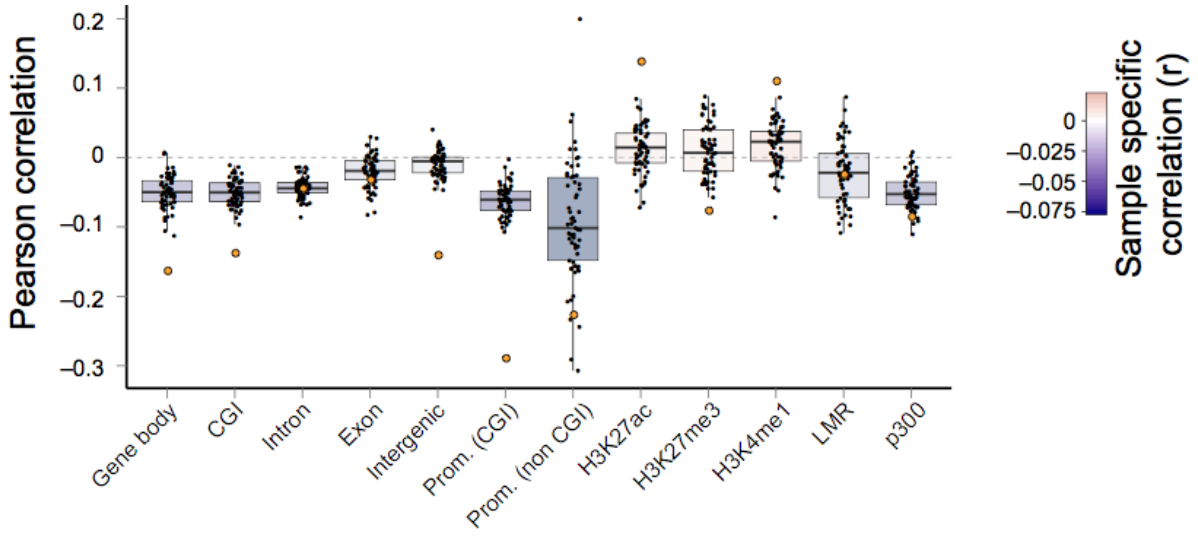


Figure 3.14 Cell-specific association analysis, estimating correlations between DNA methylation in different genomic contexts and gene expression in individual cells. For each annotation, shown are box plots of methylation-expression correlations for all variable genes in single cells, with the correlation obtained from matched RNA-seq and BS-seq of a bulk cell population superimposed (orange circles).

3.3 Discussion

We have first presented scBS-seq for profiling DNA methylation in single cells and a statistical model for quantifying methylation heterogeneity at a genome-wide scale. We demonstrated that our approach detects rare cell types and identifies genomic regions with high methylation heterogeneity that are functionally important during cell differentiation.

We further extended our protocol for parallel profiling of the methylome and transcriptome from the same single cell, thereby allowing to probe regulatory relationships between DNA methylation and expression. We both confirmed previously known negative associations between non-CGI promoter methylation and transcription in single cells, and identified novel positive and negative associations at distal regulatory regions. We found the expression levels of many pluripotency factors, e.g. Esrrb, to be negatively associated with DNA methylation. Furthermore, we found evidence that the strength of associations between methylome and transcriptome can vary from cell to cell.

We used principal component analysis to investigate the factors of variations of DNA methylation and gene expression. To better understand the relation between these factors and to distinguish between factors that drive variations either in DNA methylation or gene expres-

sion, future studies may consider advanced statistical methods, such as canonical correlation analysis [81, 217], or collective matrix factorization [195, 32, 39].

In our study, we quantified local associations between genes and genomic regions that were separated by at most 10 kbp. An important area of future research is to also investigate trans-associations beyond 10 kbp. Appealing for this purpose are chromatin contact information from Hi-C experiments [53], which can help to identify distal genomic regions that are spatially close to a certain gene.

We estimated mean methylation rates in genomic contexts by averaging and used weighted Pearson correlation to account for incomplete CpG coverage. More sophisticated methods for estimating the mean methylation rate in genomic contexts from low-coverage data may reveal stronger associations between DNA methylation and gene expression. These include methods for imputing CpG methylation in single cells, which will be discussed in the following chapter.

Chapter 4

Deep neural networks for predicting DNA methylation

Protocols for profiling DNA methylation in single cells are powerful for studying intercellular differences. However, they are limited by incomplete CpG coverage, which renders downstream analyses challenging. We therefore developed a deep neural network, *DeepCpG*, for imputing incomplete DNA methylation profiles in single cells. In this chapter, we will describe the architecture of DeepCpG and show that it yields considerably more accurate predictions than existing methods across alternative cell types and sequencing protocols. In chapter 5, we will then present methods for analysing DNA methylation in single cells using DeepCpG. The presented work is based on Angermueller et al. [9].

4.1 Motivation

Single-cell DNA profiling technologies enabled the fine-grained study of DNA methylation in single cells. However, neither genome-wide (scBS-seq; Section 3.1) nor reduced representation (scRRBS-seq; Section 1.3) protocols cover all CpG sites per cell. scBS-seq is limited by its efficiency to capture DNA and read-mapping biases, resulting in a CpG coverage of 10-30% per cell. scRRBS-seq is limited to genomic regions of high CpG density by design, resulting in a CpG coverage of 1-10% per cell. Unlike scBS-seq, scRRBS-seq is systematically biased

towards CpG dense regions, making it impossible to obtain genome-wide CpG coverage by pooling cells.

The low CpG coverage renders several downstream analyses challenging, for example quantifying DNA methylation variability between cells (Section 3.1.2), clustering cells (Section 3.1.2), or correlating DNA methylation with gene expression. Whereas data of low CpG coverage can already be sufficient to reliably estimate average DNA methylation levels in large or CpG dense regions such as gene bodies or CGIs, analysing small or CpG poor regions is challenging. This holds true for many functional relevant regions such as enhancers or repressors, which we have found to be associated with high methylation variability between cells (Section 3.1.3) and to be linked with changes in gene expression levels (Section 3.2.3).

Methods for imputing incomplete DNA methylation profiles are therefore critical for analysing CpG methylation genome-wide.

4.2 Existing methods and limitations

While the problem of imputing single-cell methylation profiles has previously not been considered, methods have been developed for imputing average methylation levels in bulk populations of cells.

Some of these methods restrict predictions to a particular genomic context, for example CGIs. The methylation level of CGIs is relatively easy to predict since they are mainly hypomethylated, unlike the rest of the genome, which is mainly hypermethylated in somatic cells. For example, Bock et al. [30] trained a support vector machine on DNA sequence-derived features and context annotations to predict CGI methylation in human lymphocytes. In addition to DNA sequence-derived features, Zheng et al. [246] found histone methylation- and histone acetylation marks to be important for predicting CGI methylation in multiple cell types.

Predicting methylation of individual CpG sites is more challenging since their methylation levels can vary considerably both within and across genomic contexts. For example, gene promoters that are overlapped by a CGI (CGI promoters) tend to be hypomethylated at the centre but hypermethylated at flanking regions, resulting in a bimodal distribution of methylation levels. In contrast, non-CGI promoters are mainly hypermethylated and can have alternative methylation patterns. Existing methods usually represent the methylation state of a CpG site as a binary or

continuous variable, and use features extracted from the local neighbourhood of the CpG site to train a conventional machine learning classifier, such as logistic regression, support vector machine, random forest, naïve Bayes, or k-nearest neighbour. Features can be categorized into i) DNA sequence features, including the frequency of k-mers in a window centred on the target CpG site, GC-content, transcription factor binding sites, DNA conservation, single nucleotide polymorphism, and repeat elements, ii) context annotations, such as the type and function of nearby genomic regions, iii) structural information, including predicted structural elements and nucleosome positioning information, iv) histone modification marks, including histone methylation- and histone acetylation marks, and v) information about observed neighbouring CpG sites. For example, Malousi et al. [144] trained a support vector machine only using DNA sequence features, including the frequency of di- and trinucleotides in 401 bp windows, known DNA sequence motifs, and Fourier-extracted periodicity signals. Zhang et al. [243] proposed a random forest classifier trained on DNA sequence features, transcription factor binding sites, histone modification marks, as well as the methylation state and distance of the two closest neighbouring CpG sites.

To reduce the number of features and understand which features have the strongest influence on methylation levels, feature selection methods have been applied, including principal component analysis ([245, 246]), maximum-relevant-minimum redundancy ([137]), sequential minimization optimization ([144]), and genetic algorithms [132].

Although existing methods have helped to better characterise DNA methylation in bulk populations of cells, they have multiple limitations. First, they separate feature extraction from model training. Instead of learning features from the raw data directly, features are pre-defined and manually extracted before model training. This makes it hard to discover new features since it is often a priori unclear which features are most relevant for predicting DNA methylation. Furthermore, feature extraction can be time-consuming and certain features might be unavailable for the cell type of interest. Second, most existing methods predict methylation for a single methylation profile without taking correlations between methylation profiles into account, for example from multiple cells. However, sharing information between related methylation profiles can improve prediction accuracy, in particular in domains where the methylation level of CpG sites is missing in some cells but available in others. ChromImpute [62] is a notable exception, a method that uses information from multiple epigenomic profiles, which, however, was not explicitly designed to predict DNA methylation. Third, some conventional machine learning models scale poorly to large datasets and cannot be trained online, which makes it impossible to adapt pre-trained models on new data. Some methods further do not support

training on multiple tasks with missing labels, which is required for learning models on multiple cells with partially observed methylation levels.

4.3 DeepCpG model architecture

To address the aforementioned limitations, we developed DeepCpG, a computational method based on deep neural networks (Chapter 2) for predicting single-cell methylation states and for modelling the sources of DNA methylation variability. DeepCpG leverages associations between DNA sequence patterns and methylation states, as well as between neighbouring CpG sites, both within individual cells and across cells. Unlike previous methods [25, 132, 136, 137, 207, 243, 248], DeepCpG does not separate the extraction of informative features and model training. Instead, DeepCpG is based on a modular architecture and learns predictive DNA sequence- and methylation patterns in a data-driven manner. Furthermore, DeepCpG is trained by online gradient descent, which enables to efficiently adapt or 'fine-tune' pre-trained models on new data.

DeepCpG predicts binary CpG methylation states from local DNA sequence windows and observed neighbouring methylation states (Figure 4.1 (a)). A major feature of the model is its modular architecture, consisting of a CpG module to account for correlations between CpG sites within and across cells, a DNA module to detect informative sequence patterns, and a joint module that integrates the evidence from the CpG and DNA module to predict methylation states at target CpG sites (Figure 4.1 (b)).

4.3.1 DNA module

The DNA module is a CNN (Section 2.2.2) consisting of a stack of convolutional and pooling layers, which is followed by one fully connected hidden layer. CNNs are designed to extract features from high-dimensional inputs while keeping the number of model parameters tractable by applying a series of convolutional and pooling operations. Unless stated otherwise, the DNA module takes as input a 1001 bp long DNA sequence centred on a target CpG site n, which is represented as a binary matrix s_n by one-hot encoding the $D = 4$ nucleotides as binary vectors $A = [1,0,0,0]$, $T = [0,1,0,0]$, $G = [0,0,1,0]$, and $C = [0,0,0,1]$. The input matrix s_n is first transformed by a one-dimensional convolutional layer, which computes the activation a_{nfi} of

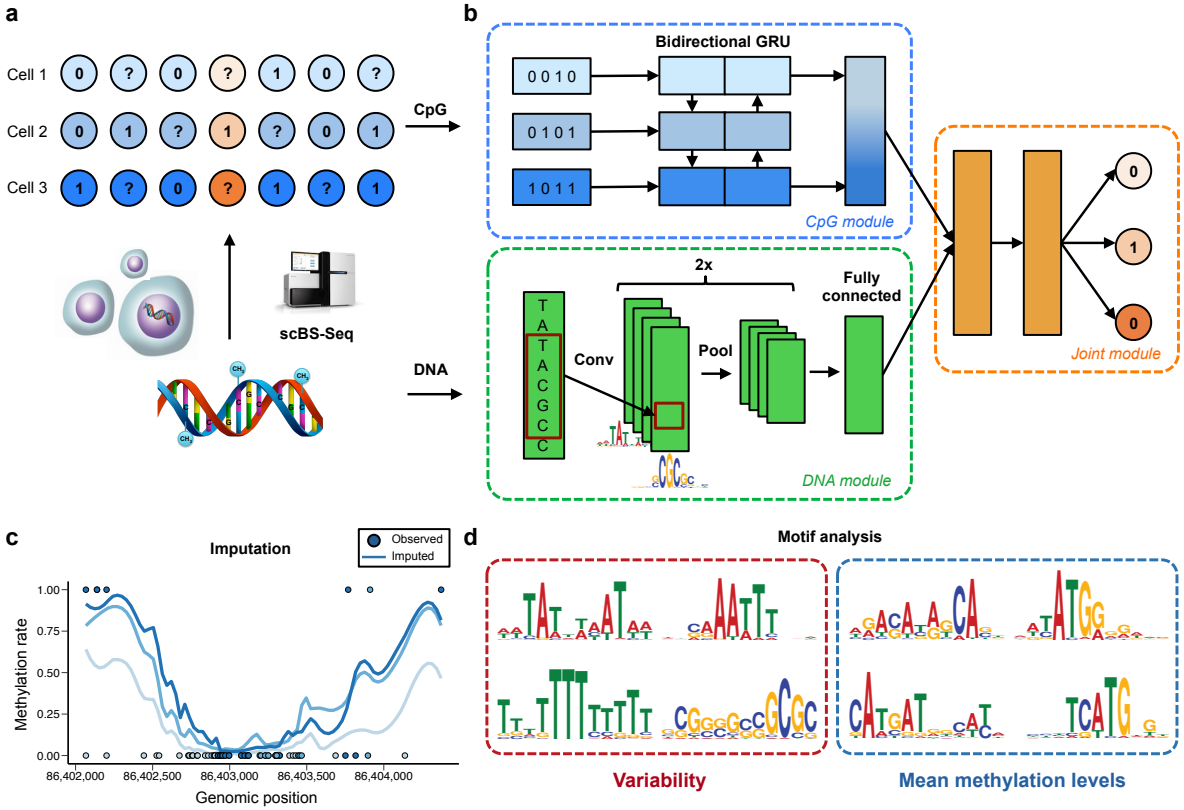


Figure 4.1 DeepCpG model training and applications. (a) Sparse single-cell CpG profiles as obtained from scBS-seq (Section 3.1) or scRRBS-seq [65, 79, 92]. Methylated CpG sites are denoted by ones, un-methylated CpG sites by zeros, and question marks denote CpG sites with unknown methylation state (missing data). (b) Modular architecture of DeepCpG. The DNA module consists of two convolutional and pooling layers to identify predictive motifs from the local sequence context, and one fully connected layer to model motif interactions. The CpG module scans the CpG neighbourhood of multiple cells (rows in b), using a bidirectional GRU (Section 2.2.3), yielding compressed features in a vector of constant size. The joint module learns interactions between higher-level features derived from the DNA- and CpG module to predict methylation states in all cells. (c, d) The trained DeepCpG model can be used for different downstream analyses, including genome-wide imputation of missing CpG sites (c) and the discovery of DNA sequence motifs that are associated with DNA methylation levels or cell-to-cell variability (d).

multiple convolutional filters f at every position i within the DNA sequence window:

$$a_{nfi} = \text{ReLU}(\sum_{l=1}^{L}\sum_{d=1}^{D} w_{fld} s_{n,i+l,d}) \tag{4.1}$$

Here, w_f are the parameters or weights of convolutional filter f of length L. These can be interpreted similarly to position weight matrices, which are matched against the input DNA sequence window s_n at every position i to recognize distinct motifs. The $\text{ReLU}(x) = \max(0,x)$ activation function sets negative values to zero, such that a_{nfi} corresponds to the evidence that the motif represented by w_f occurs at position i.

A pooling layer is used to summarize the activations of P adjacent neurons by their maximum value:

$$p_{nfi} = \max_{|k|<P/2} (a_{nf,i+k}) \tag{4.2}$$

Non-overlapping pooling is applied with step size P to decrease the dimension of the input sequence and hence the number of model parameters. The DNA module has multiple pairs of convolutional-pooling layers to learn higher-level interactions between sequence motifs, which are followed by one final fully connected layer with a ReLU activation function. The number of convolutional-pooling layers is a hyperparameter, which was selected by optimizing the prediction performance on the validation set (Section 2.5.3).

4.3.2 CpG module

The CpG module consists of a non-linear embedding layer to model dependencies between CpG sites within cells, which is followed by a bidirectional GRU (Section 2.2.3) to model dependencies between cells. Inputs are $100d$ vectors $x_1, \ldots, x_T$, where x_t represents the methylation state and distance of $K = 25$ CpG sites to the left and to the right of a target CpG site in cell t. Distances were transformed to relative ranges by dividing by the maximum genome-wide distance. The embedding layer is fully connected and transforms x_t into a $256d$ vector $\bar{x}_t$, which allows learning possible interactions between methylation states and distances within cell t:

$$\bar{x}_t = \text{ReLU}(Wx_t + b) \tag{4.3}$$

The sequence of vectors $\bar{x}_t$ are then fed into a bidirectional GRU, which scans input sequence vectors $\bar{x}_1, \ldots, \bar{x}_T$ from left to right, and encodes them into fixed-size hidden state vectors $h_1, \ldots, h_T$:

$$
\begin{aligned}
r_t &= \text{sigmoid}(W_{rx}\bar{x}_t + W_{rh}h_{t-1} + b_r) \\
u_t &= \text{sigmoid}(W_{ux}\bar{x}_t + W_{uh}h_{t-1} + b_u) \\
\tilde{h}_t &= \tanh\left(W_{\tilde{h}x}\bar{x}_t + W_{\tilde{h}h}(r_t \odot h_{t-1}) + b_{\tilde{h}}\right) \\
h_t &= (1 - u_t) \odot h_{t-1} + u_t \odot \tilde{h}_t
\end{aligned} \tag{4.4}
$$

The reset gate r_t and update gate u_t determine the relative weight of the previous hidden state h_{t-1} and the current input $\bar{x}_t$ for updating the current hidden state h_t. The last hidden state h_T summarizes the sequence as a fixed-size vector. Importantly, the set of parameters W and b are independent of the sequence length T, which enables to fine-tune a pre-trained CpG module on a new dataset with a different number of cells.

By using a bidirectional GRU, cell-to-cell dependencies are encoded independently of the order of cells. It consists of a forward and backward GRU with $256d$ hidden state vectors h_t, which scan the input sequence from the left and right, respectively. The last hidden state vector of the forward and backward GRU are concatenated into a $512d$, which forms the output of the CpG module.

4.3.3 Joint module

The joint module takes as input the concatenated last hidden vectors of the DNA and CpG module, and models interactions between the extracted DNA sequence and CpG neighbourhood features via two fully connected hidden layers with 512 neurons and ReLU activation function. This enables context-dependent smoothing of neighbouring CpG sites and hence modelling of alternatively shaped methylation profiles, for example in promoter or enhancer regions. The output layer of the joint module contains T sigmoid neurons to predict the methylation rate $\hat{y}_{nt} \in (0;1)$ of CpG site n in cell t:

$$
\hat{y}_{nt} = \text{sigmoid}(x) = \frac{1}{1 + e^{-x}} \tag{4.5}
$$

4.3.4 Model training

Model parameters were learned on the training set by minimizing the following loss function:

$$L(w) = \mathrm{NLL}_w(\hat{y}, y) + \lambda_2 \|w\|_2 \tag{4.6}$$

Here, the weight-decay hyperparameter λ_2 penalises large model weights quantified by the L2 norm, and $\mathrm{NLL}_w(\hat{y}, y)$ denotes the negative log-likelihood, which measures how well the predicted methylation rates $\hat{y}_{nt}$ fit to observed binary methylation states $y_{nt} \in \{0, 1\}$:

$$\mathrm{NLL}_w(\hat{y}, y) = \sum_{n=1}^{N} \sum_{t=1}^{T} o_{nt} \left[y_{nt} \log(\hat{y}_{nt}) + (1 - y_{nt}) \log(1 - \hat{y}_{nt}) \right] \tag{4.7}$$

The binary indicator o_{nt} is set to one if the methylation state y_{nt} is observed for CpG site n in cell t, and zero otherwise. Dropout ([203]; Section 2.5.3) with different dropout rates for the sequence, CpG, and joint module was used for additional regularization. Model parameters were initialized randomly following the approach in Glorot and Bengio [73]. The loss function was optimized by mini-batch stochastic gradient descent with a batch size of 128 and a global learning rate of 0.0001. The learning rate was adapted by Adam ([114]; Section 2.5.3), and decayed by a factor of 0.95 after each epoch. Learning was terminated if the validation loss did not improve over ten consecutive epochs (early stopping). The DNA and CpG module were pre-trained independently to predict methylation from the DNA sequence (DeepCpG DNA) or the CpG neighbourhood (DeepCpG CpG). For training the joint module, only the parameters of the hidden layers and the output layers were optimized, whiling keeping the parameters of the pre-trained DNA and CpG module fixed. Training DeepCpG on 18 serum mESCs using a single NVIDIA Tesla K20 GPU took approximately 24 hours for the DNA module, 12 hours for the CpG module, and 4 hours for the joint module. Model hyperparameters were optimized on the validation set by random sampling ([19]; Section 2.5.3).

4.3.5 Software availability

DeepCpG is publicly available on Github[1] and implemented in Python using Theano [15] and Keras [44]. Our software includes documentation and interactive tutorials on training, evalu-

[1] https://github.com/cangermueller/deepcpg

ating, and analysing DeepCpG. We also provide pre-trained models, which can be efficiently fine-tuned on new data.

4.4 Prediction performance evaluation

First, we assessed the ability of DeepCpG to predict single-cell methylation states and compared the model to existing imputation strategies for DNA methylation. As a baseline approach, we considered local averaging of the observed methylation states, either in 3 kb windows centred on the target site of the same cell (WinAvg; Section 3.1.2) or across cells at the target site (CpGAvg). Additionally, we compared DeepCpG to a random forest classifier [36] trained on individual cells using the DNA sequence information and neighbouring CpG states as input (RF). Finally, we evaluated a recently proposed random forest classifier for predicting methylation rates in bulk populations of cells [243], which takes comprehensive DNA annotations into account, including genomic contexts, and tissue-specific regular annotations, such as DNase1 hypersensitivity sites, histone modification marks, and transcription factor binding sites (RF Zhang). All methods were trained, selected, and tested on distinct chromosomes via holdout validation (Section 2.5.2). Specifically, we used chromosomes 1, 3, 5, 7, 9, and 11 as training set, chromosomes 2, 4, 6, 8, 10, and 12 as test set, and the remaining chromosomes as validation set. For each cell type, models were fit on the training set, hyperparameters were optimized on the validation set, and the final model performance and interpretations were exclusively reported on the test set.

Since the proportion of methylated versus unmethylated CpG sites can be unbalanced in globally hypo- or hypermethylated cells, we used the area under the receiver operating characteristics curve (AUC) to quantify the prediction performance of different models. We have also considered a range of alternative metrics, including precision-recall curves, F1 score [173], and Matthews correlation coefficient [147], resulting in overall consistent conclusions (Figure B.1; Figure B.2).

Initially, we applied all methods to 18 serum-cultured mouse embryonic stem cells (mESCs; average CpG coverage 17.7%; Figure B.3), profiled using scBS-seq (Section 3.1).

DeepCpG yielded more accurate predictions than any of the alternative methods, both genome-wide and in different genomic contexts (Figure 4.2). Notably, DeepCpG was consistently more accurate than RF Zhang, a model that relies on genomic annotations. These results

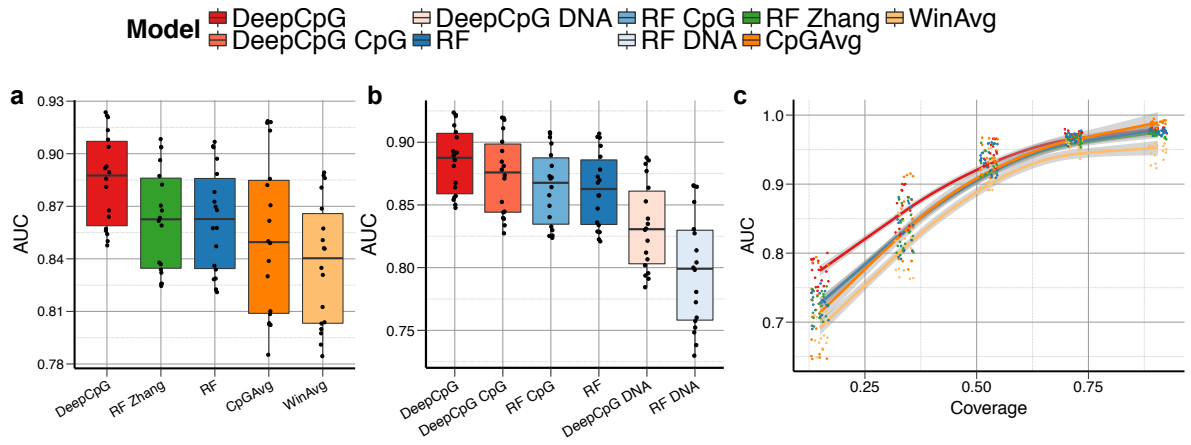


Figure 4.2 Prediction performance of DeepCpG. (a) Genome-wide prediction performance for 18 serum-grown mouse embryonic stem cells (mESCs) profiled using scBS-seq. Performance is measured by the area under the receiver-operating characteristic curve (AUC), using holdout validation. Considered were DeepCpG, random forest classifiers trained either using DNA sequence and CpG features (RF) or trained using additional annotations from corresponding cell types (RF Zhang). Additionally, two baseline methods were considered, which estimate methylation states by averaging observed methylation states, either across consecutive 3 kb regions within individual cells (WinAvg) or across cells at a single CpG site (CpGAvg). (b) Performance breakdown of DeepCpG and RF, comparing the full models to models trained either only using methylation features (DeepCpG CpG, RF CpG) or DNA features (DeepCpG DNA, RF DNA). (c) AUC of the methods as in (a) stratified by genomic contexts with increasing CpG coverage across cells. Trend lines were fit using local polynomial regression (LOESS); shaded areas denote 95% confidence intervals.

indicate that DeepCpG can automatically learn higher-level features from the DNA sequence. To investigate this, we tested for associations between the activity of convolutional filters in the DNA module and known sequence annotations (Section 5.1), finding both positive and negative correlations with several annotation contexts, including DNase1 hypersensitive sites, histone modification marks, and CpG-rich genomic contexts (Figure B.4). The ability to extract higher-level features from the DNA sequence is particularly important for analysing single-cell datasets, where individual cells may be from different cell types and states, making it difficult to derive appropriate annotations.

To assess the relative importance of DNA sequence features compared to neighbouring CpG sites, we trained the same models, however, either exclusively using DNA sequence features (DeepCpG DNA, RF Seq) or neighbouring methylation states (DeepCpG CpG, RF CpG). Consistently with previous studies in bulk populations [243], methylation states were more predictive than DNA features, and models trained with both CpG and DNA features performed best (Figure 4.2 (b)). Notably, DeepCpG trained with CpG features alone outperformed a random forest classifier trained with both CpG and DNA features. A likely explanation for the accuracy of the CpG module is its recurrent network architecture, which enables the module to effectively transfer information from neighbouring CpG sites across different cells (Figure B.5).

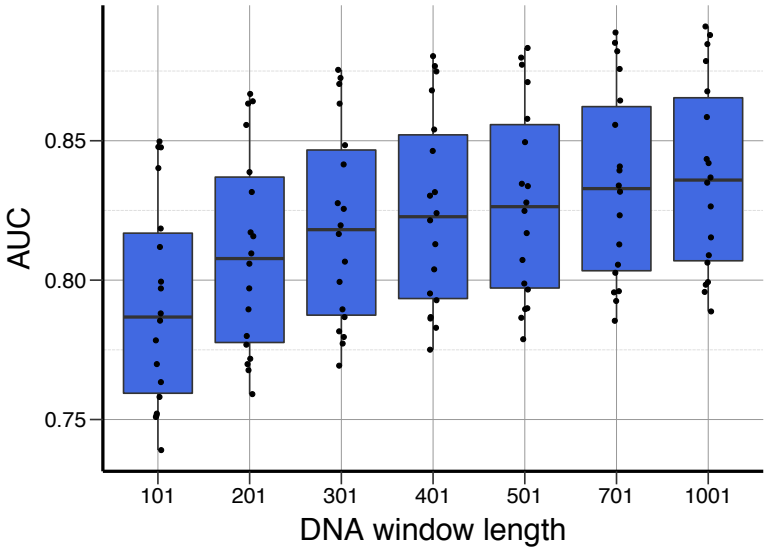


Figure 4.3 Prediction performance of the DeepCpG DNA module for DNA sequence windows of length 101 bp up to 1001 bp.

The largest relative gains between RF and DeepCpG were observed when training both models with DNA sequence information only (AUC 0.83 versus 0.80; Figure 4.2 (b)). This demonstrates the strength of the DeepCpG DNA module to extract predictive sequence features from large DNA sequence windows of up to 1001 bp (Figure 4.3), which is in particular critical for accurate predictions from DNA in uncovered genomic regions, for example when using reduced

representation sequencing data [65, 78, 92]. Consistent with this, the relative performance gain of DeepCpG compared to other methods was highest in contexts with low CpG coverage (Figure 4.2 (c); Figure B.6).

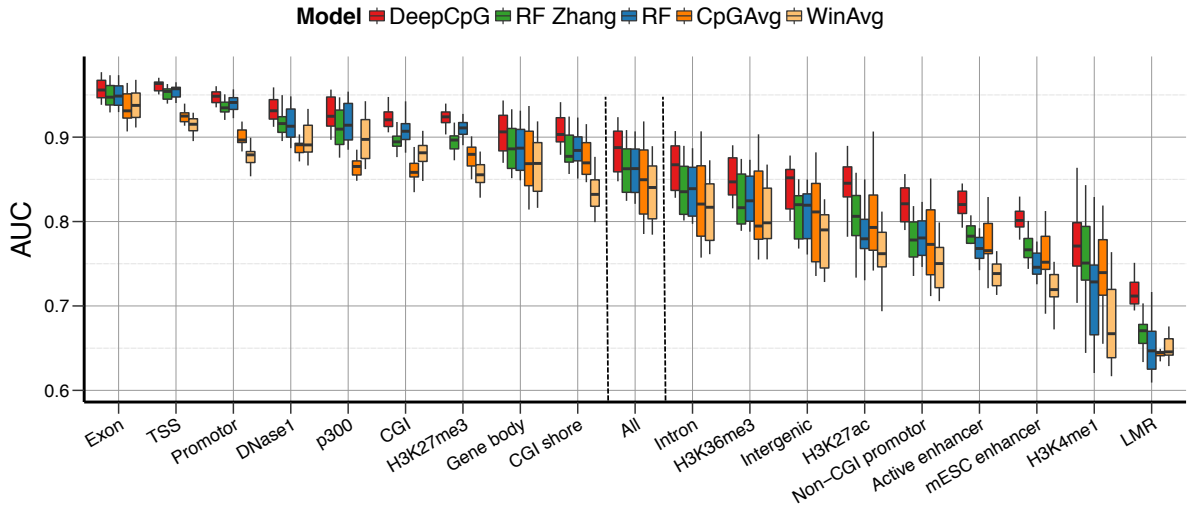


Figure 4.4 Prediction performance of DeepCpG for alternative genomic contexts, with 'All' corresponding to genome-wide performances.

Next, we explored the prediction performance of all models in different genomic contexts. In line with previous findings [207, 243], all models performed best in GC-rich contexts (Figure 4.4). However, DeepCpG offered most advantages in GC-poor genomic contexts, including non-CGI promoters, enhancer regions, and histone modification marks (H3K4me1, H3K27ac)–contexts that we found to be associated with higher methylation variability between cells (Section 3.1.3).

We also applied DeepCpG to 12 2i-cultured mESCs profiled using scBS-seq and to data from three cell types profiled using scRRBS-seq [92], including 25 human hepatocellular carcinoma cells (HCC), 6 human heptoplastoma-derived (HepG2) cells, and an additional set of 6 mESCs. Notably, in contrast to the serum cells, the human cell types are globally hypomethylated (Figure B.3). Across all cell types, DeepCpG yielded substantially more accurate predictions than alternative methods (Figure 4.5; Figure B.2), demonstrating the broad applicability of the model, including to hypo- and hypermethylated cells, as well to data generated using different sequencing protocols.

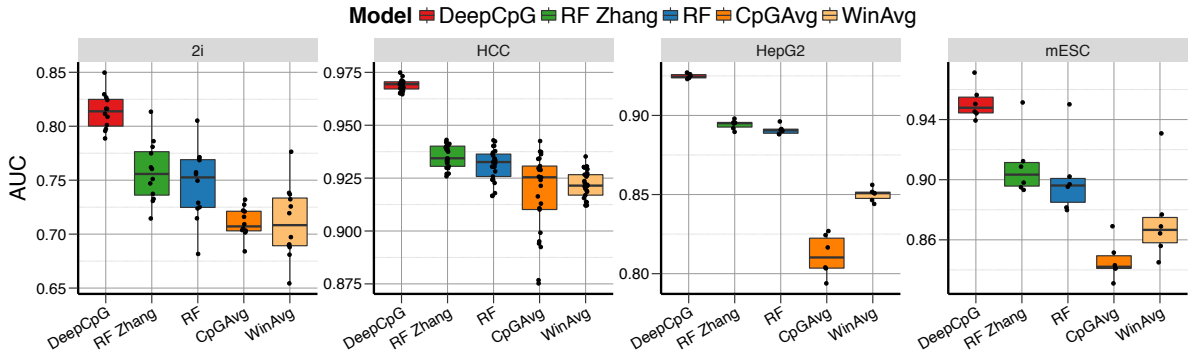


Figure 4.5 Prediction performance of DeepCpG for alternative datasets. Shown are genome-wide performances for 12 2i-grown mESCs profiled using scBS-seq, as well as three cell types profiled using scRRBS-seq, including 25 human HCC cells, 6 HepG2 cells, and 6 additional mESCs.

4.5 Discussion

We have developed a deep neural network, *DeepCpG*, for predicting DNA methylation in single cells. We have demonstrated that our model enables accurate imputation of missing methylation states, thereby facilitating genome-wide downstream analyses. DeepCpG offers major advantages in shallow sequenced cells as well as in sparsely covered sequence contexts with increased methylation variability between cells. Our method may also help to reduce the required sequencing depth in single-cell bisulfite sequencing studies, thereby enabling the analysis of larger numbers of cells at reduced cost.

The increased accuracy of DeepCpG comes at the expense of higher compute costs. Training DeepCpG genome-wide on a single GPU takes about two days, which is considerably longer compared with conventional machine learning models. However, parallelising computations across multiple GPUs and compute nodes can drastically reduce the training time. Since DeepCpG processes all cells simultaneous, it further shares computations across cells and hence scales better than existing methods to data sets with many cells. Faster training is also possible by only fine-tuning a model that was pre-trained on a related cell type instead of training a new model from scratch.

An important area of future work is to simultaneously impute multiple bulk and single-cell methylation profiles, and to integrate different data modalities from parallel-profiling methods, which are now becoming increasingly available for multiple molecular layers.

Chapter 5

Analysis of deep neural networks for predicting DNA methylation

Deep neural networks are often criticised as hard to interpret 'black-box' models. However, model interpretability is critical in multiple fields, including biology and health care. Hence, several approaches have been developed to interpret the parameters of neural networks (Section 2.4.4; Section 2.3) and to obtain insights into learned features. We adapted some of these approaches to analyse DeepCpG (Section 4.3). In particular, we show that DeepCpG can be applied to discover DNA sequence motifs that are associated with methylation states, to identify variance-associated motifs, and to estimate the effect of single nucleotide mutations on DNA methylation. The presented work is based on Angermueller et al. [9].

5.1 Discovering DNA sequence motifs

As described in section 4.3, DeepCpG consists of a DNA, CpG, and joint module. The DNA module has a convolutional architecture with a series of convolutional and max-pooling layers. The filters of the first convolutional layer recognize DNA sequence motifs similarly to conventional position weight matrices and can be visualised as sequence logos (Section 2.3). Existing approaches for visualizing convolutional filters are either *alignment-based* or *optimization-based*. Alignment-based approaches [5, 112, 174] align DNA sequence fragments that maximally activate a certain convolutional filter and visualize the resulting alignment as sequence

logo. Optimization-based approaches [121, 122] optimize the input DNA sequence by gradient descent to maximize the activation of a certain convolutional filter. Our approach is alignment-based. Specifically, we computed the activations a_{nfi} of all filters f of the first convolutional layer for input sequences s_n at every position i. We selected sequence window $s_{n,i-L}, \ldots, s_{n,i+L}$, if the activation a_{nfi} of filter f with length L at position i (Equation 4.1) was greater than 0.5 of the maximum activation of f over all sequences, i.e. $a_{nfi} > 0.5 \max_{ni}(a_{nfi})$. We then aligned all selected sequence fragments and visualized filter alignments as sequence logos using WebLogo [50].

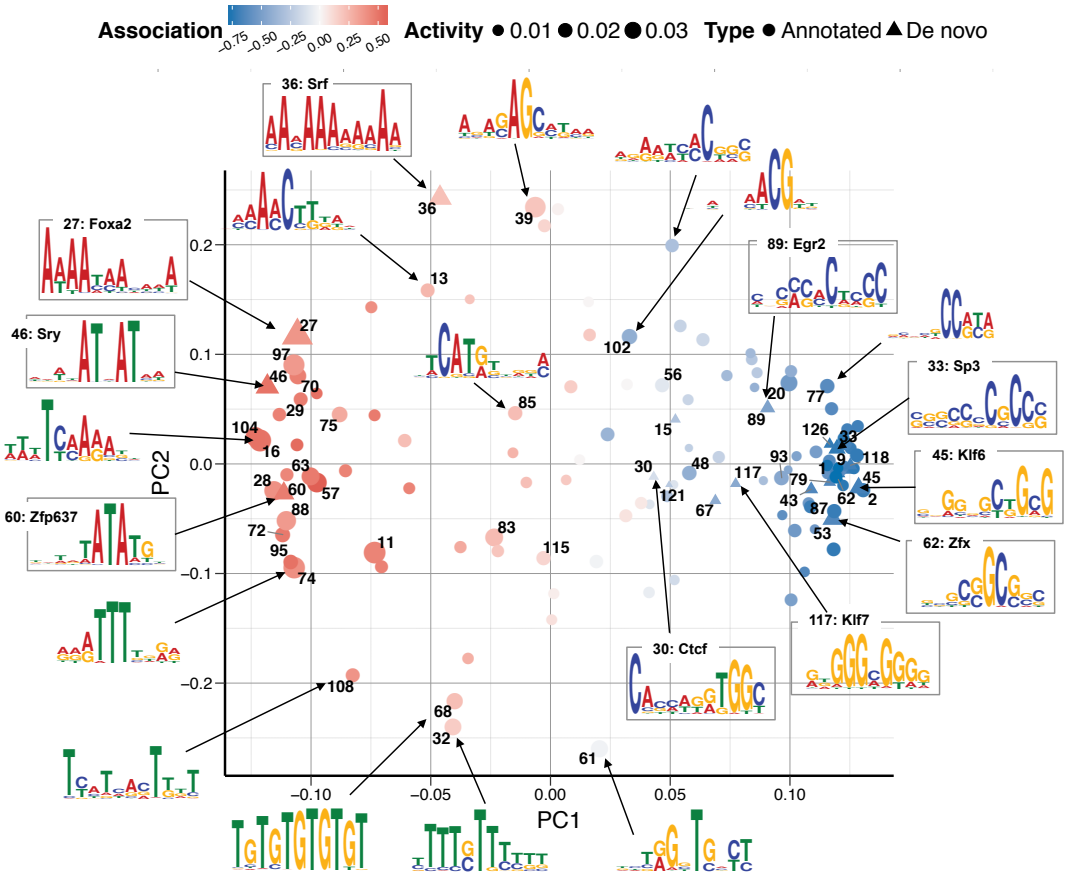


Figure 5.1 Principal component analysis of discovered DNA sequence motifs. Clustering of 128 motifs discovered by DeepCpG. Shown are the first two principal components of the motif occurrence frequencies in sequence windows (activity). Triangles denote motifs with significant (FDR < 0.05) similarity to annotated motif in the CIS-BP or UniPROPE database. Marker size indicates the average activity; the estimated motif effect on methylation level is shown in colour. Sequence logos are shown for representative motifs with larger effects, including 10 annotated motifs.

The number of motifs that the model learns corresponds to the number of convolutional filters, which is a pre-defined hyperparameter. Depending on the choice of this hyperparameter and the nature of the underlying data, learned motifs can be redundant or only weakly associated with DNA methylation. We therefore quantified the importance of learned motifs by two metrics: their activity (occurrence frequency) and their influence on model predictions (association). Specifically, for a set of sequences, e.g. within a certain genomic context, we computed the average of mean sequence activities $\bar{a}_{nf}$, where $\bar{a}_{nf}$ denotes the weighted mean of activities a_{nfi} across all positions i in the sequence window of length W:

$$\bar{a}_{nf} = \frac{1}{\sum_{i=1}^{W} w_i} \sum_{i=1}^{W} w_i a_{nfi} \qquad w_i = 1 - \frac{|i - 0.5W|}{0.5W} \tag{5.1}$$

w_i are linear weights that are highest close the centre position ($i = 0.5W$) of the sequence window. We computed the influence of filter f on the predicted methylation states $\hat{y}_{nt}$ of cell t (Equation 4.5) as the Pearson correlation $r_{ft} = \text{cor}_n(\bar{a}_{nf}, \hat{y}_{nt})$ over CpG sites n, and the mean influence r_f over all cells by averaging r_{ft}.

To investigate the co-occurrence of motifs across sequence windows, we applied principal component analysis (Figure 5.1) and hierarchical clustering (Figure 5.2) to motif activities. Motifs with similar nucleotide composition tended to co-occur in the same sequence windows, where two major motif clusters were associated with increased or decreased methylation levels (Figure 5.1; Figure C.2). Consistent with previous findings [153, 218, 231], we observed that motifs associated with decreased methylation tended to be CG rich and were most active in CG rich promoter regions, transcription start sites, as well as in contexts with active promoter marks, such as H3K4me3 or p300 sites (Figure 5.2). Conversely, motifs associated with increased methylation levels tended to be AT rich and were most active in CG poor genomic contexts (Figure 5.2).

We further compared learned motifs, i.e. the weights of the first convolutional layer, to known motifs using Tomtom [14]. 20 out of the 128 learned motifs matched significantly (FDR < 0.05) motifs annotated in the Mus Musculus CIS-BP [229] and UniPROBE [162] database. 17 of these motifs were transcription factors with a known implication in DNA methylation [85, 138, 231], including CTCF [113], E2f [219], and members of the Sp/KLF family [66]–transcription factors and regulators of cell differentiation. 13 out of the 20 annotated motifs had been shown to interact with DNMT3a and DNMT3b [85], two major DNA methylation enzymes. Three motifs have no clear associations with DNA methylation. These included Foxa2 [126, 226] and Srf [10, 145], which are implicated in cell differentiation and embryonic development, as well

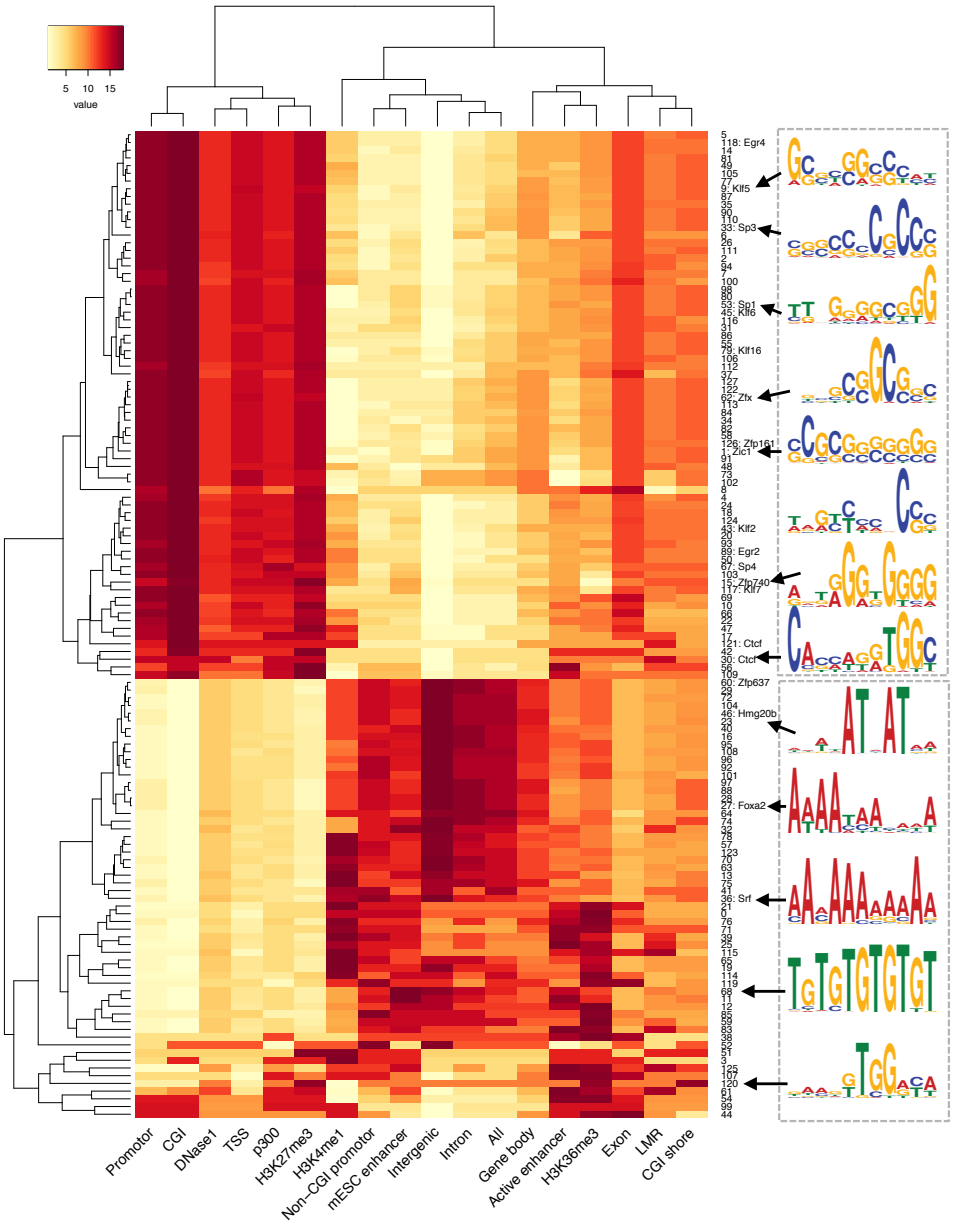


Figure 5.2 Activity of DNA sequence motifs in genomic contexts. Rank of motif activities in different genomic contexts on test chromosomes. The highest activity of CG rich motifs is observed in GC-rich regions such as promotors and CpG islands (CGI). Conversely, the highest activity of AT rich motifs is observed in GC-poor regions such as intergenic or active enhancer regions.

as Zfp637 [93, 175], a zinc finger protein that has recently been linked to spermatogenesis in mouse.

5.2 Identifying variance-associated motifs

DNA sequence motifs can either have similar effects in all cells and thereby influence the mean methylation rate across cells, or they can have cell-specific effects and hence account for the variance between cells. Of particular interests for understanding intercellular differences are variance-associated motifs, which we discerned from motifs that are primarily associated with mean methylation levels.

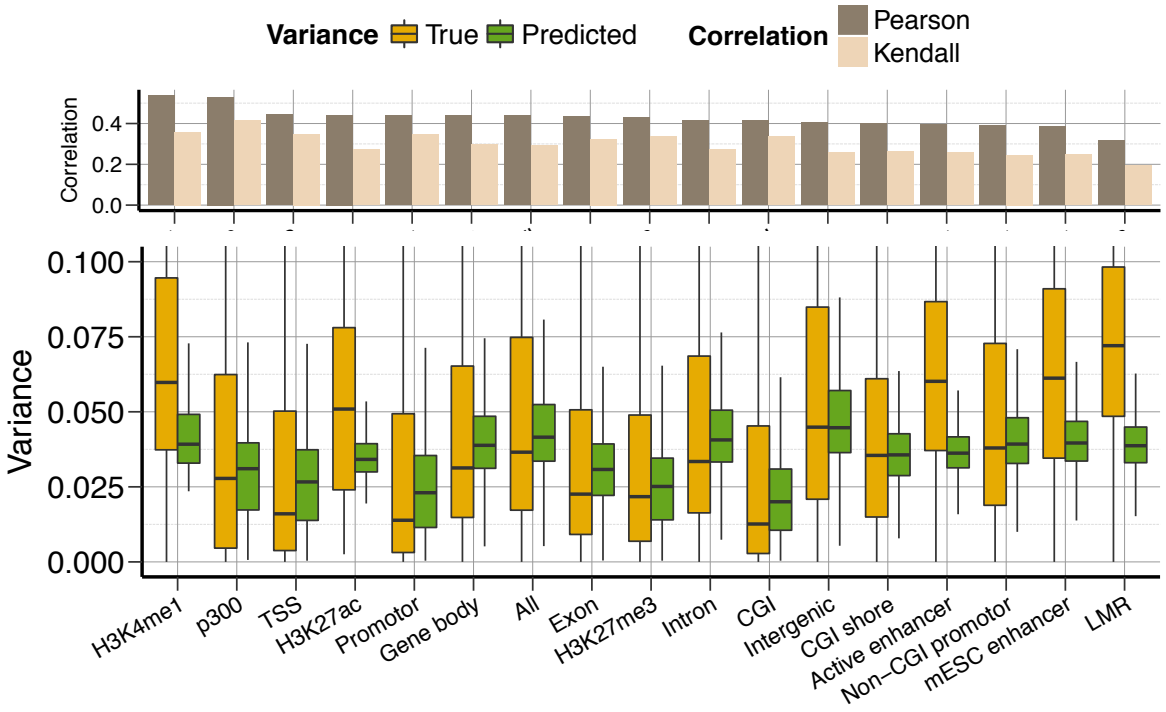


Figure 5.3 Performance of DeepCpG DNA module to predict methylation variability. Boxplot represent predicted (green) and the observed (orange) methylation variability in 3 kbp windows centred on individual CpG sites for different genomic contexts on held out test chromosomes. Barplot represent Pearson and Kendall correlation coefficients.

For this purpose, we trained a second neural network that had the same architecture and in particular reused the motifs from the DNA module of DeepCpG. However, this network was now trained to predict for single CpG sites the variability across cells and the mean methylation

rate. Specifically, we replaced output neurons by neurons with a sigmoid activation function to predict for CpG site n both the mean methylation rate $\hat{m}_{ns}$ and cell-to-cell variance $\hat{v}_{ns}$ within a window of size $s \in \{1000, 2000, 3000, 4000, 5000\}$ bp. We used multiple window sizes to obtain predictions at different scales, thereby mitigating the uncertainty of mean- and variance estimates in low-coverage regions. For training the resulting model, we initialized and fine-tuned parameters with the corresponding parameters of the DNA module, except for motif parameters of the first convolutional layer. The training objective was

$$L(w) = \mathrm{MSE}_w(\hat{m}, m, \hat{v}, v) + \lambda_2 \|w\|_2 , \tag{5.2}$$

where MSE the is mean squared error between model predictions and training labels:

$$\mathrm{MSE}_w(\hat{m}, m, \hat{v}, v) = \sum_{n=1}^{N} \sum_{s=1}^{S} (m_{ns} - \hat{m}_{ns})^2 + (v_{ns} - \hat{v}_{ns})^2 \tag{5.3}$$

m_{ns} is the estimated mean methylation rate for a window centred on target site n of a certain size indexed by s:

$$m_{ns} = \frac{1}{T} \sum_{t=1}^{T} m_{nst} \tag{5.4}$$

Here, m_{nst} denotes the estimated mean methylation rate of cell t computed by averaging the binary methylation state y_{it} in the set of all observed CpG sites Y_{nst} in window s

$$m_{nst} = \frac{1}{|Y_{nst}|} \sum_{i \in Y_{nst}} y_{it}, \tag{5.5}$$

where v_{ns} denotes the estimated cell-to-cell variance:

$$v_{ns} = \frac{1}{T} \sum_{t=1}^{T} (m_{nst} - m_{ns})^2 \tag{5.6}$$

We evaluated this model by holdout validation as we did before (Section 4.4). Notably, the model could predict both global changes in mean methylation levels (Pearson's $R = 0.80$; MAD $= 0.01$; Figure C.3), as well as cell-to-cell variability (Pearson's $R = 0.44$; MAD $= 0.03$, figure 5.3; Kendall's $R = 0.29$, figure C.4).

There is an intrinsic mean-variance relationship of single-cell methylation states (Figure C.5), and hence the separation of the motif impact on mean methylation and methylation variance is partially confounded. To mitigate this effect, we used a scoring approach that separates the

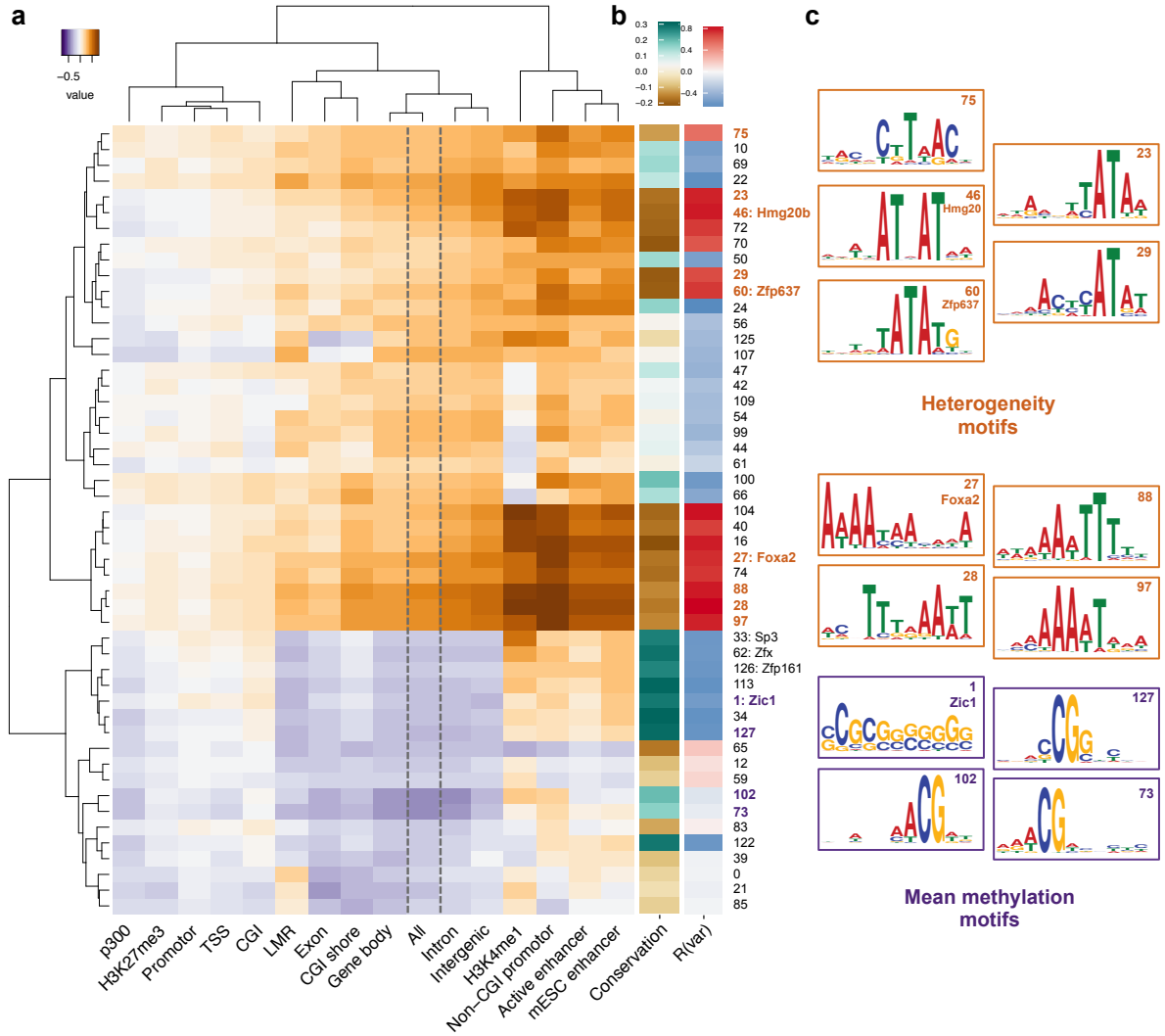


Figure 5.4 Identification of motifs associated with mean methylation levels and cell-to-cell variability. (a) Difference of motif effect on cell-to-cell variability and methylation levels for different genomic contexts. Motifs associated with increased cell-to-cell variability are highlighted in brown; motifs that were primarily associated with changes in methylation level are shown in purple. (b) Genome-wide correlation coefficients between motif activity and DNA sequence conservation (left), as well as cell-to-cell variability (right). (c) Sequence logos for selected motifs identified in (a), which are highlighted by colour in (b).

effect of individual motifs on cell-to-cell variability and mean methylation levels. Specifically, we computed the influence r^v_{fs} of filter f on cell-to-cell variability in widows of size s as the Pearson correlation between mean sequence filter activities $\bar{a}_{nf}$ and predicted variance levels $\hat{v}_{ns}$ of sites n:

$$r^v_{fs} = \text{cor}_n(\bar{a}_{nf}, \hat{v}_{ns}) \tag{5.7}$$

Analogously, we computed the influence r^m_{fs} on predicted mean methylation levels $\hat{m}_{ns}$. We then used the difference $r^d_{fs} = |r^v_{fs}| - |r^m_{fs}|$ between the absolute value of the influence on variance and mean methylation levels to identify motifs that were primarily associated with cell-to-cell variance $r^d_{fs} > 0.25$, or with changes in mean methylation levels $r^d_{fs} < -0.25$.

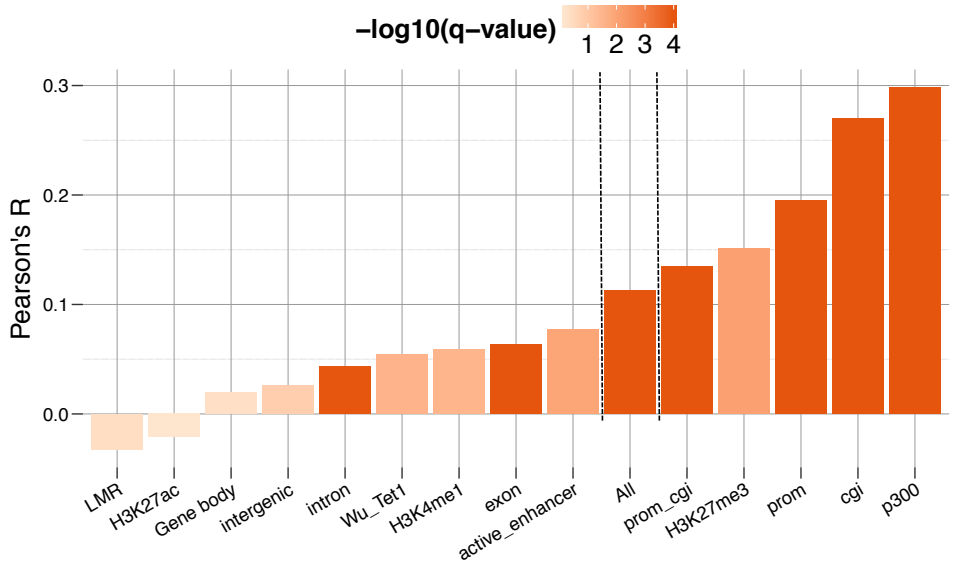


Figure 5.5 Functional assessment of predicted cell-to-cell variability. Pearson correlation coefficient between methylome-transcriptome linkage as reported in section 3.2, and the predicted cell-to-cell variability for test chromosomes. Colour denotes statistical significance (q-value, Benjamini Hochberg adjusted). Significant correlations (FDR < 0.01) were observed genome-wide ('All') and in most genomic contexts.

The approach identified 22 motifs that were primarily associated with cell-to-cell variance (Figure 5.4). These motifs tended to be active in CG-poor and active enhancer regions–sequence contexts with increased epigenetic variability (Section 3.1.3). 12 of the identified motifs were AT-rich and associated with increased variability, including the differentiation factors Foxa2 [126, 226], Hmg20b [210], and Zfp637 [93, 175]. Notably, variance-increasing motifs were more frequent in un-conserved regions such as active enhancers, in contrast to

variance-decreasing motifs, which were enriched in evolutionary conserved regions such as gene promoters (Figure 5.4 (b); Figure C.6). Our analysis also revealed 4 motifs that were primarily associated with mean methylation levels, which were in contrast CG rich and most active in conserved regions.

To explore whether the model predictions for variable sites are functionally relevant, we overlaid predictions with methylome-transcriptome linkages from our previous study (Section 3.2). The rationale behind this approach is that regions with increased methylation variability are more likely to harbour associations with gene expression. Consistent with this hypothesis, we observed a weak but globally significant association (Pearson's $R = 0.11$; $P = 5.72 \times 10^{-}16$; Figure 5.5).

5.3 Estimating the effect of DNA mutations

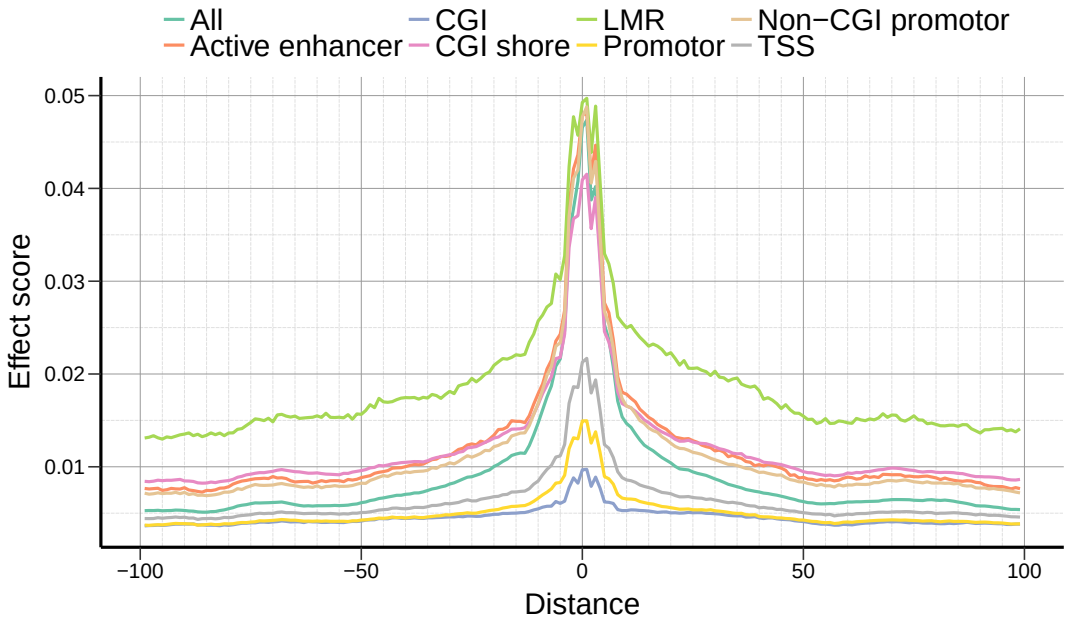


Figure 5.6 Average genome-wide effect of single nucleotide mutations on CpG methylation estimated using DeepCpG, depending on the distance to the CpG site and the genomic context.

The trained DeepCpG model can also be used to estimate the effect of single nucleotide mutations on CpG methylation. For this purpose, we adapted a gradient-based approach [194] to estimate mutational effects in a computationally efficient manner, thereby greatly reducing the compute cost compared to previous methods [5, 247, 247]. Specifically, let $\hat{y}_n(s_n) = \frac{1}{T}\sum_t \hat{y}_{nt}(s_n)$

be the mean predicted methylation rate across cells t for an input sequence s_n. Then we quantified the effect e^s_{nid} of changing nucleotide d at position i as:

$$e^s_{nid} = \frac{\Delta \hat{y}_{nt}(s_n)}{\Delta s_{nid}}(1 - s_{nid}) \tag{5.8}$$

Here, the first term is the first-order gradient of $\hat{y}_n$ with respect to s_{nid}, and the second term sets the effect of wild-type nucleotides ($s_{nid} = 1$) to zero. We quantified the overall effect score e^s_{ni} at position i as the maximum absolute effect over all nucleotide changes, i.e. $e^s_{ni} = \max_d(|e^s_{nid}|)$.

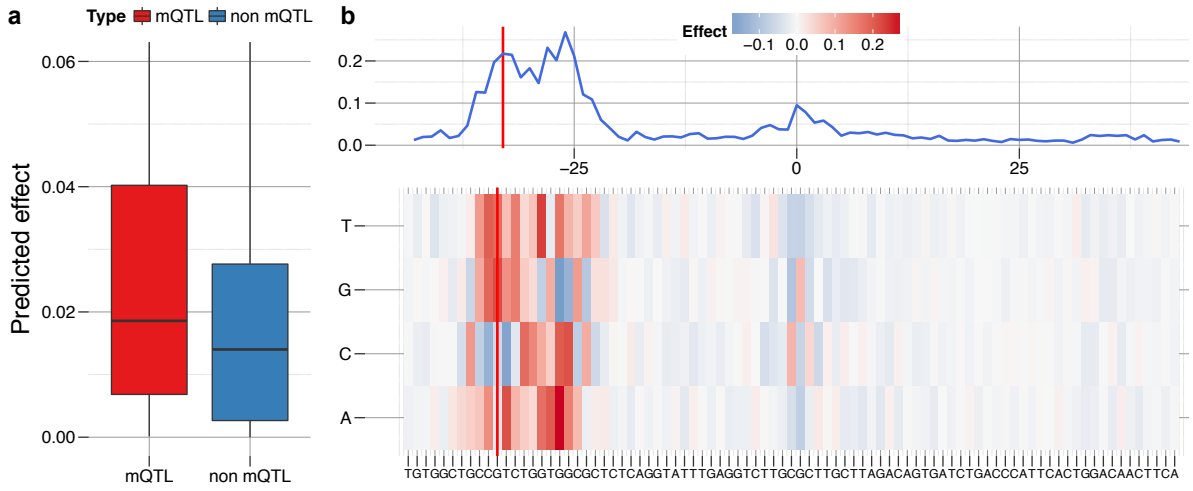


Figure 5.7 Analysis and example visualization of estimated single nucleotide mutation effects for methylation QTLs (mQTLs). (a) Distribution of the mutation effect for 2379 mQTLs variants from Kaplow et al. [108], compared to distance-matched random variants (non mQTL). (b) Visualization of mutation effects for an example CpG site and the corresponding mQTL. Shown are effects in a window centred on the example CpG site (chromosome 1; position 159791497). The position of the corresponding mQTL from Kaplow et al. [108] (rs60205880; position 159791464) is indicated by the vertical red line. The heat map in the lower panels shows effect sizes for individual nucleotides, and line plot in the upper panel the maximum absolute effect across nucleotides.

As expected, mutations in the direct vicinity of the target CpG site had the largest effects (Figure 5.6). Mutations in CG dense regions such as CGIs or promoters tended to have smaller effects, suggesting that DNA methylation in these genomic contexts is more robust to single nucleotide mutations. Globally, we observed a negative correlation between mutational effects and DNA sequence conservation ($P < 1.0 \times 10^{-15}$; Figure C.7), providing evidence that estimated single nucleotide effects capture genuine effects.

We further evaluated mutational effects by comparing the effect for 2379 methylation QTLs (mQTLs) from Kaplow et al. [108] with the effect for matched non-mQTL variants. Since Kaplow et al. [108] mapped mQTLs in human cells, we used the DeepCpG model trained on HepG2 cells for this experiments. To adjust of differences in effect due to the distance to the CpG site (Figure 5.6), we randomly sampled non-QTLs at distances matched to mQTLs. Known mQTL variants had significantly larger effects than matched random variants ($P < 1.0 \times 10^{-15}$, Wilcoxon rank sum test; Figure 5.7), providing evidence that DeepCpG can be used to identify functional variants from the DNA sequence alone.

5.4 Discussion

We have developed alternative approaches for interpreting the parameters of DeepCpG and for obtaining insights into the learned features. We have used these approaches to discover motifs that are associated with DNA methylation states, to identify variance-associated motifs, and to estimate the effect of single nucleotide mutations on DNA methylation.

Although our approaches helped to better understand DeepCpG, they are not free of limitations. First, the randomness that is involved in the training of DeepCpG affects which specific motifs are learned. This applies in particular to the random initialization of the parameters of the first convolutional layer. One way to address this problem is to initialize a fraction of convolutional filters by known DNA sequence motifs, which, however, can introduce biases and decrease prediction accuracy.

Although our approaches helped to better understand DeepCpG, they are not free of limitations. First, the randomness that is involved in the training of DeepCpG affects which specific motifs are learned. This applies in particular to the random initialization of the parameters of the first convolutional layer. One way to address this problem is to initialize a fraction of convolutional filters by known DNA sequence motifs, which, however, requires prior knowledge about methylation-associated motifs for the cell type of interest and can introduce biases.

Second, the model can learn redundant and overlapping motifs depending on the choice of the number of convolutional filters and the dropout rate. More distinct motifs can be obtained by penalizing the number motifs with non-zero weights, analogously to the Bayesian automatic relevance determination framework [142, 159].

Third, our scoring approach to identify variance-associated motifs does not fully disentangle the mean-variance relationship of binomial-distributed methylation states in single cells. One approach to more clearly separate between mean and variance motifs would be to model methylation states using a beta-binomial distribution, where mean and over-dispersion parameters are predicted by a deep neural network, similar to mixture density networks [28].

Forth, our gradient-based approach to quantify the effect of single nucleotide mutations is problematic because the ReLU activation function can have a gradient of zero but still transfer information. This problem has recently been addressed by comparing neuron activations to reference activations [192]–a promising strategy also in the context of DNA methylation. Another avenue of future research is to quantify statistical significance of estimated effect scores, which is important for prioritizing genomic variants.

Chapter 6

Summary and future research

The dogmatic view that organisms are predetermined by their genetic code has been shattered by the discovery of epigenetic factors, including DNA methylation, histone modifications, and small non-coding RNA sequences. DNA methylation has been the topic of this thesis, which is involved in important biological processes, including gene regulation, imprinting, X-chromosome inactivation, and the repression of retroviruses. Single-cell DNA methylation profiling protocols are promising for studying DNA methylation at a single-cell resolution and for understanding differences between cells. However, current protocols are limited by incomplete CpG coverage and high-levels of technical noise, which renders downstream analyses challenging.

In this thesis, we presented computational methods for the analysis of single-cell methylation data. In chapter 3, we described computational methods for the analysis DNA methylation in single cells, which we developed jointly with seminal single-cell profiling protocols. We described scBS-seq, a protocol (Section 3.1) for genome-wide profiling of DNA methylation in single cells, and a statistical model for estimating methylation heterogeneity in different genomic contexts. We further presented scM&T-seq, a protocol (Section 3.2) for parallel profiling of DNA methylation and gene expression in the same cell, and methods for estimating associations between the methylome and transcriptome. By identifying both previously known and novel regulatory associations between DNA methylation and gene expression, we demonstrated that scM&T-seq is a powerful approach for investigating the poorly understood relationship between transcriptional and epigenetic heterogeneity in single cells.

The low CpG coverage of single-cell methylation profiles renders genome-wide analyses challenging. Hence, methods for imputing methylation profiles are critical. In chapter 4, we described DeepCpG, a computational approach based on deep neural networks for predicting methylation states in single cells. Unlike previous methods, DeepCpG is based on a modular architecture to learn informative features in a data-driven manner and leverages information from multiple cells. By evaluating DeepCpG on single-cell methylation data from five cell types generated using alternative sequencing protocols, we demonstrated that DeepCpG yields substantially more accurate predictions than previous methods and is widely applicable.

In chapter 5, we described approaches for interpreting the learned parameters of DeepCpG. By interpreting the filters of the first convolutional layer as sequence motifs, we have shown that DeepCpG learns both previously known and potentially new motifs that are associated with DNA methylation states. We have further developed a scoring approach to discern motifs that are either primarily associated with cell-to-cell variance or mean DNA methylation levels. Third, we have presented a gradient-based approach to efficiently estimate the effect of single nucleotide mutations on DNA methylation, and have shown that estimated effects are significantly increased for experimentally validated mQTLs.

There are many open leads for future research. DeepCpG does not estimate predictive uncertainty, which makes it hard to decide if predictions are accurate enough for a particular downstream analysis. Extending DeepCpG to estimate predictive uncertainty is one important future research direction. Inspiration might be drawn from Bayesian neural networks [159], which are, however, limited by high computational costs. Recently, Gal and Ghahramani [69] described the use of dropout for performing approximate Bayesian inference and representing uncertainty in a computationally efficient manner. This approach is straightforward to implement in DeepCpG since it already uses dropout for regularization.

Another important avenue of future research is to further improve the interpretability of DeepCpG. We have shown that the filters of the first convolutional layer of the DeepCpG DNA module can be interpreted as sequence motifs, and hence DeepCpG be used for motif discovery. It is interesting to also analyse higher-level convolutional features, which may reveal more complex sequence patterns that are composed of elementary motifs. Higher-level features can be visualized, for example, by activation maximization [143], a technique that uses the gradient signal to find the model inputs that maximally activate a certain feature. An alternative way to improve the interpretability is attention [43]. Attention mechanisms have been used to identify the most salient input features, such as regions in an image that are most relevant for a certain prediction task [43, 237, 190]. In particular soft-attention is straightforward to implement in

DeepCpG and provides the potential to better understand which regions in the DNA sequence window or which cells are most relevant for predicting DNA methylation.

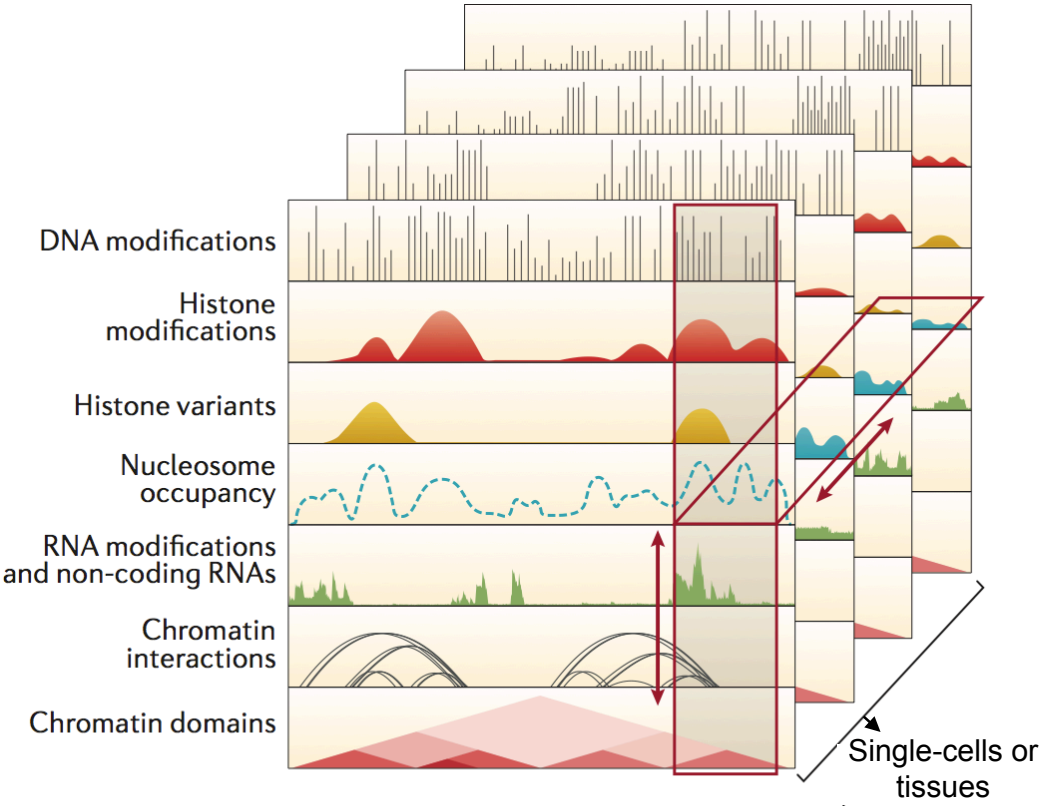


Figure 6.1 Imputation of multiple molecular layers in multiple cells. Missing layers are imputed by sharing information both between molecular layers and between cells. Source: Stricker et al. [209].

DeepCpG predicts methylation states independently for each CpG site and cell. Extending DeepCpG to take dependencies between its predictions into account is likely to further improve prediction accuracy. Appealing for this purpose are conditional generative models, including generative recurrent neural networks [76, 46], pixel convolutional neural networks [222, 106, 221], and restricted Boltzmann machines [212, 33, 34, 17]. These models have been successfully used for generating handwriting, natural text, and audio sequences, and can be adapted to also generate a sequence of methylation states conditioned on partially observed methylation states and the DNA sequence.

DeepCpG is trained using information from the DNA sequence and incomplete DNA methylation profiles. An important extension of DeepCpG is to integrate additional data modalities profiled in the same cells, for example gene expression and copy number variations. These

data modalities can be integrated by additional modules that are specific for each modality. Alternatively, data modalities can be processed jointly by representing them analogously to the colour channels of a multi-dimensional image tensor.

DeepCpG has been designed for imputing a single data modality (DNA methylation) in multiple cells from one or more input data modalities. More general are models for imputing multiple data modalities in multiple cells (Figure 6.1). These models would be more broadly applicable to sets of cells that were profiled using different parallel profiling protocols. In this settings, some data modalities, e.g. DNA methylation and gene expression, might be available in a subset of cells but missing or only partially be available in a subset of cells that was processed using a different protocol, e.g. for profiling DNA accessibility and histone modifications. Missing data modalities can be imputed by sharing information both between data modalities and between cells. Based on this idea, Ernst and Kellis [62] proposed an ensemble of regression trees for imputing multiple epigenomic profiles in bulk populations of cells. Appealing deep neural network architectures for learning from multi-modal data are, for example, Boltzmann machines [202], autoencoders [40, 176], variational autoencoders [164, 214, 188], and deep canonical correlation analysis [6, 22]. These architectures exchange information between data modalities by shared latent representations, which are inferred from the observed data and used to impute missing data.

Multi-modal latent variable models are also well suited for analysing the factors of variations that underlie epigenetic data. Inferring latent variables that are either shared across or specific to individual cells and molecular layers can help to better characterize cells and molecular layers as well as their relationships. Inferring latent variables of individual cells enables clustering cells, detecting outlier cells, and interpreting the type and state of cells. Inferring latent variables that are shared across and specific to data modalities enables investigating the relationships between molecular layers, for example associations between gene expression and DNA methylation. While latent variable models are promising for imputing and understanding multi-modal epigenetic data, they are often limited by high computational costs. Developing models that are computational efficient is critical for genome-wide analyses of large datasets and practical applications. Another challenge is the integration data that are profiled at different scales, for example per base pair, per CpG sites, or per gene. Finally, methods must be user-friendly and easy to use by non-experts.

Appendix A

Protocols and analysis of single-cell DNA methylation data

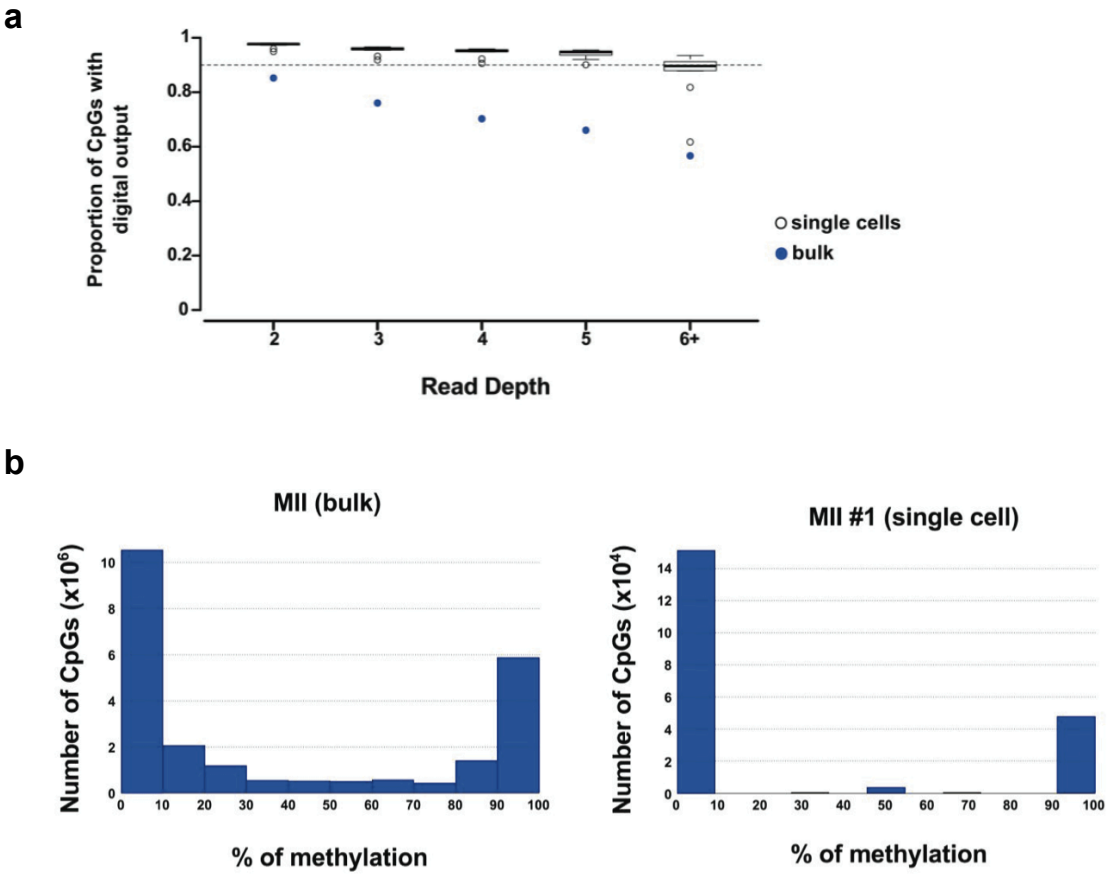


Figure A.1 scBS-seq generates a digital output of DNA methylation. (a) For each single MII BS-Seq library, and for the bulk MII sample, CpGs were grouped based on their read depth. The proportion of CpGs in each group with a methylation value of either 0% or 100% (digital output) was calculated for each sample. The boxplot represents the results from all 12 single MII libraries. The results from the bulk MII sample are superimposed as solid blue circles. As expected, the proportion of digital CpGs in the scBS-Seq libraries was very high ($>$ 90% for read depth 2-5 in all cells, dashed line). In contrast, the bulk sample had fewer digital CpGs (66% at read depth 5) due to cell-to-cell variability within the population. (b) Histograms of the distribution of CpG methylation values for MII bulk and MII single cells for CpGs with at least 2 reads.

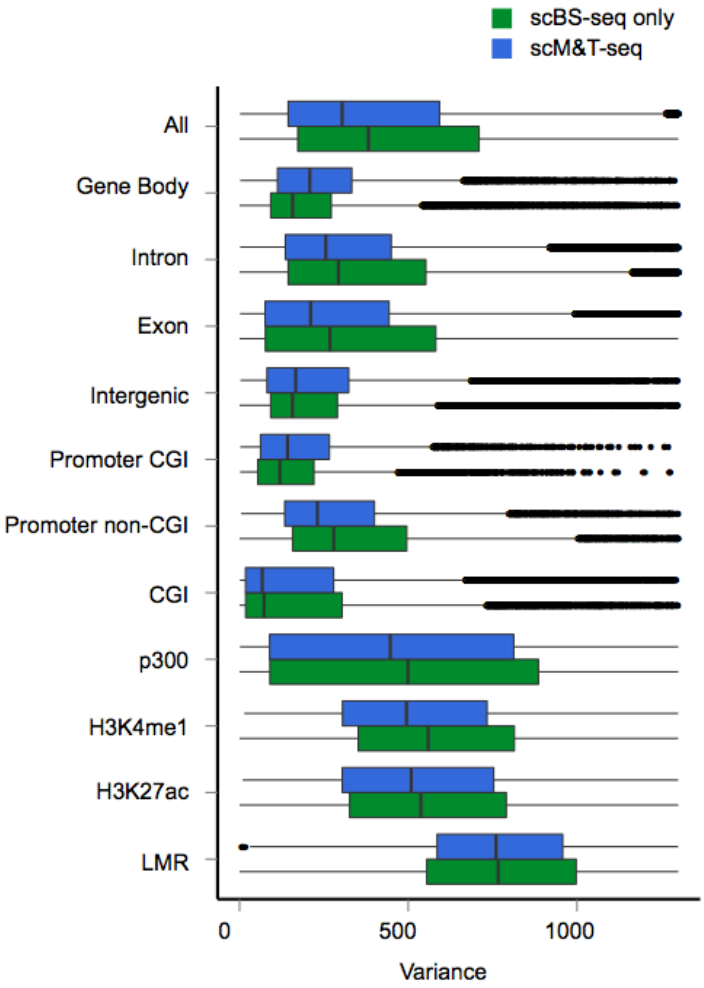


Figure A.2 Comparison of epigenetic heterogeneity in different genomic contexts, considering 61 serum ESCs obtained using scM&T-seq and 20 serum ESCs sequenced using stand-alone scBS-seq.

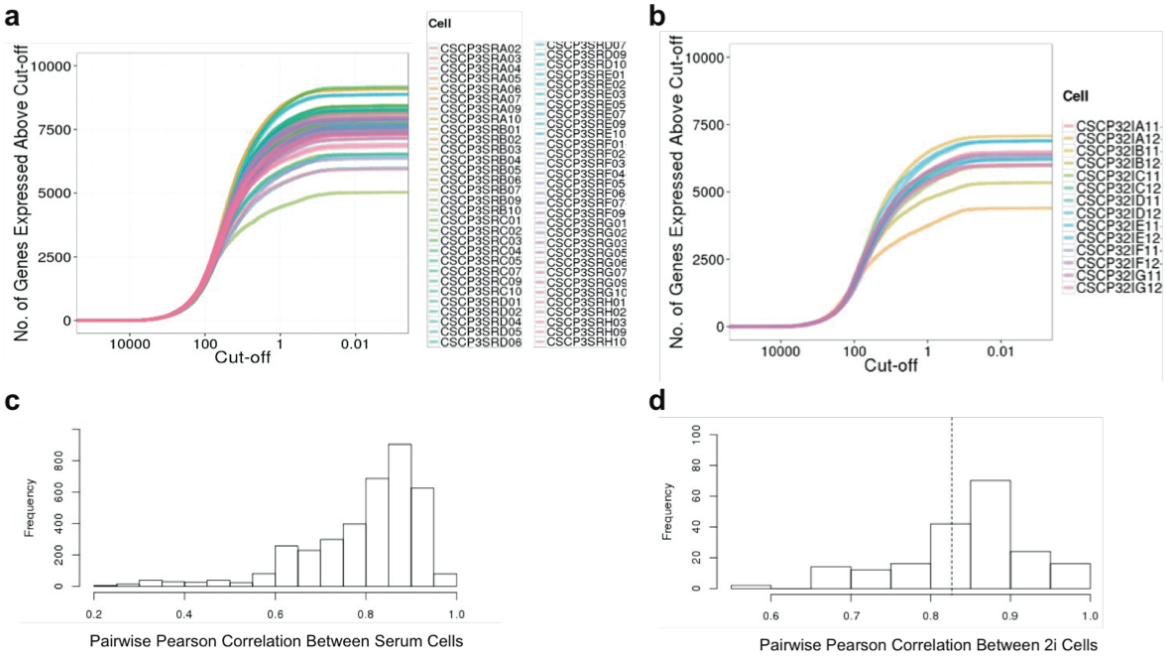


Figure A.3 Quality metrics of scRNA-seq data obtained from mouse ESCs profiled using scM&T-seq. (a,b) Number of genes detected on (Y-axis) as a function of the expression cut off (x-axis). In each cell, between 4,000 and 8,000 genes were expressed (TPM>1) (the dashed line drawn at X=1). High quality cells generally have about 5,000 genes detectable at the cut-off of TPM>1, indicating a high level of quality among the 61 serum ESCs (or the 14 2i ESCs). (c,d) Distribution of Pearson correlation coefficient calculated pairwise on the 61 serum ESCs (or the 14 2i ESCs). The observed correlation coefficient tended to be between 0.7-0.99, indicating a high degree of technical consistency in the measured transcriptome of the cells considered, and attesting high quality of scRNA-seq data.

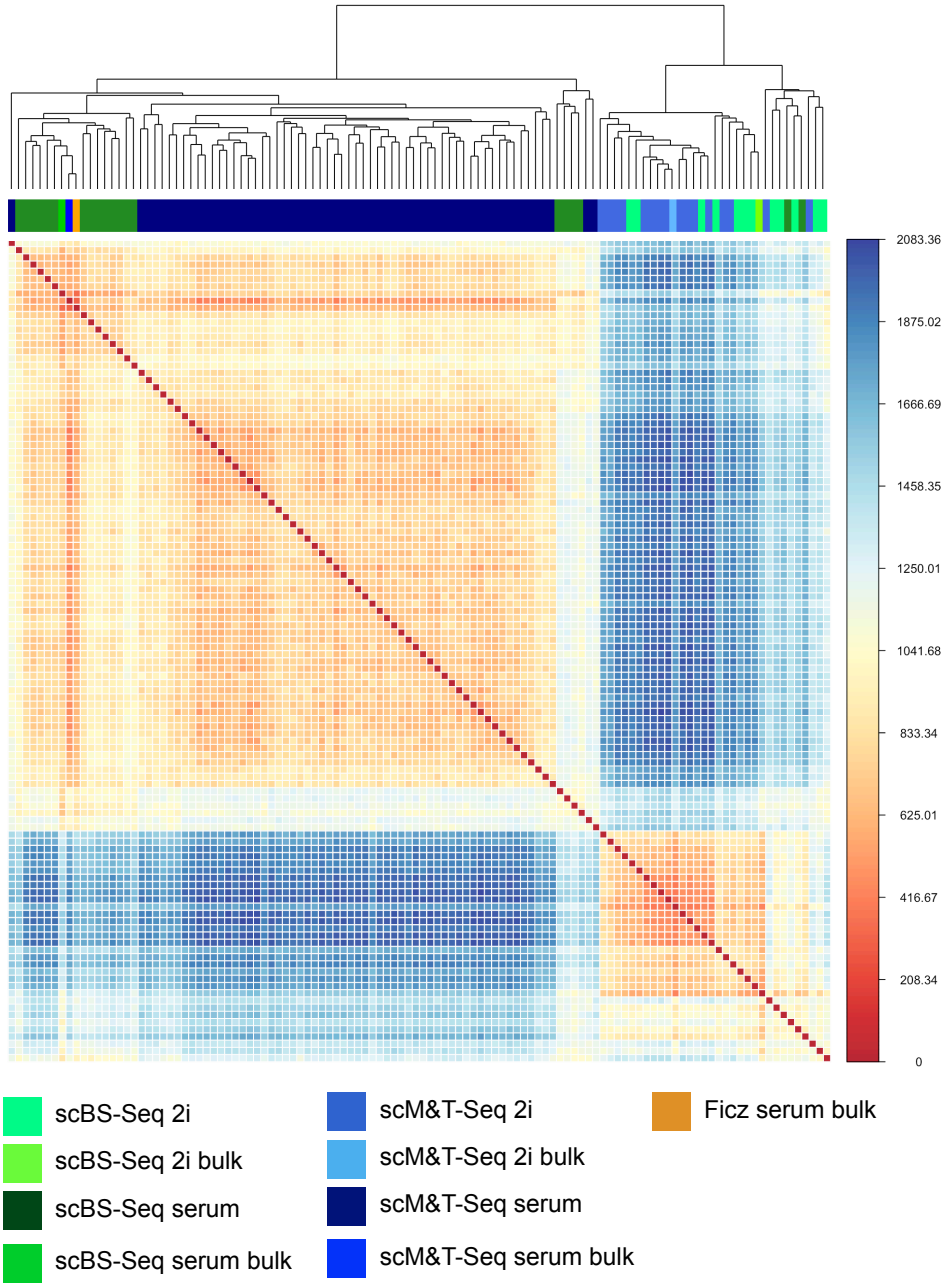


Figure A.4 Shown s a joint hierarchical clustering from 61 serum and 16 2i cells profiled using scM&T-seq, as well as 20 serum and 12 2i ESCs profiled by scBS-seq (Smallwood et al. 2014), as well as corresponding synthetic bulk samples and an independent bulk BS-seq sample from serum ESCs (Ficz et al. 2013). The clustering analysis was performed on gene body methylation of the 500 genes with the largest epigenome heterogeneity.

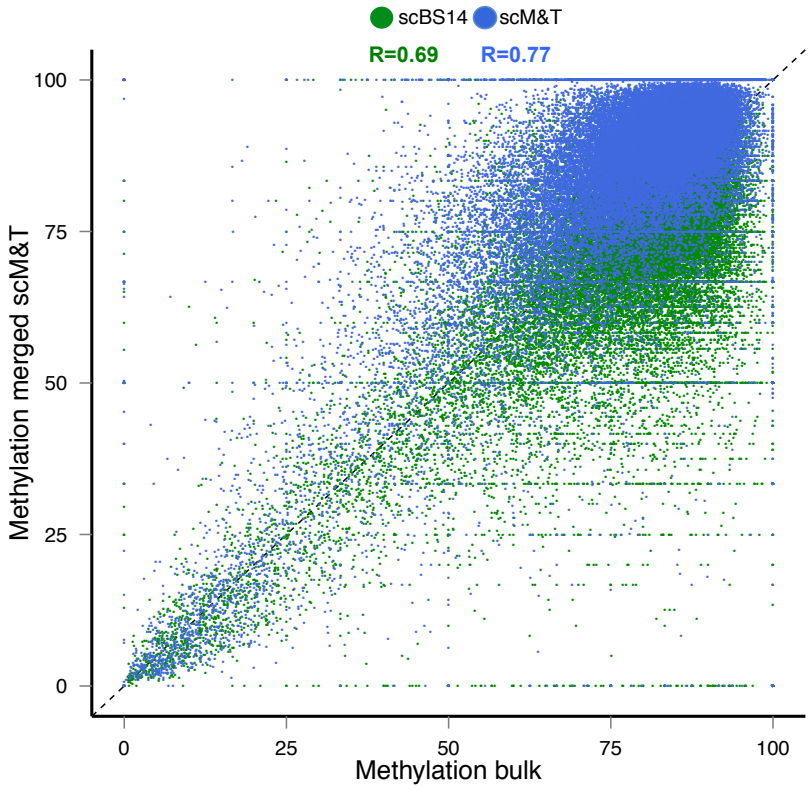


Figure A.5 Shown is a scatter plot, relating bulk gene body methylation (Ficz et al. 2013) on the x-axis, versus synthetic bulk estimates of gene- body methylation derived using either scBS-seq (Smallwood et al. 2014, green) or scM&T-seq (blue) on the y-axis. Synthetic bulk methylation profiles are derived form averages of the single-cell methylation profiles. The true bulk methylation profile is concordant with both single-cell profiles, where the scM&T-seq bulk estimates correlate slightly better (R=0.77) than the scBS-seq bulk (R=0.69).

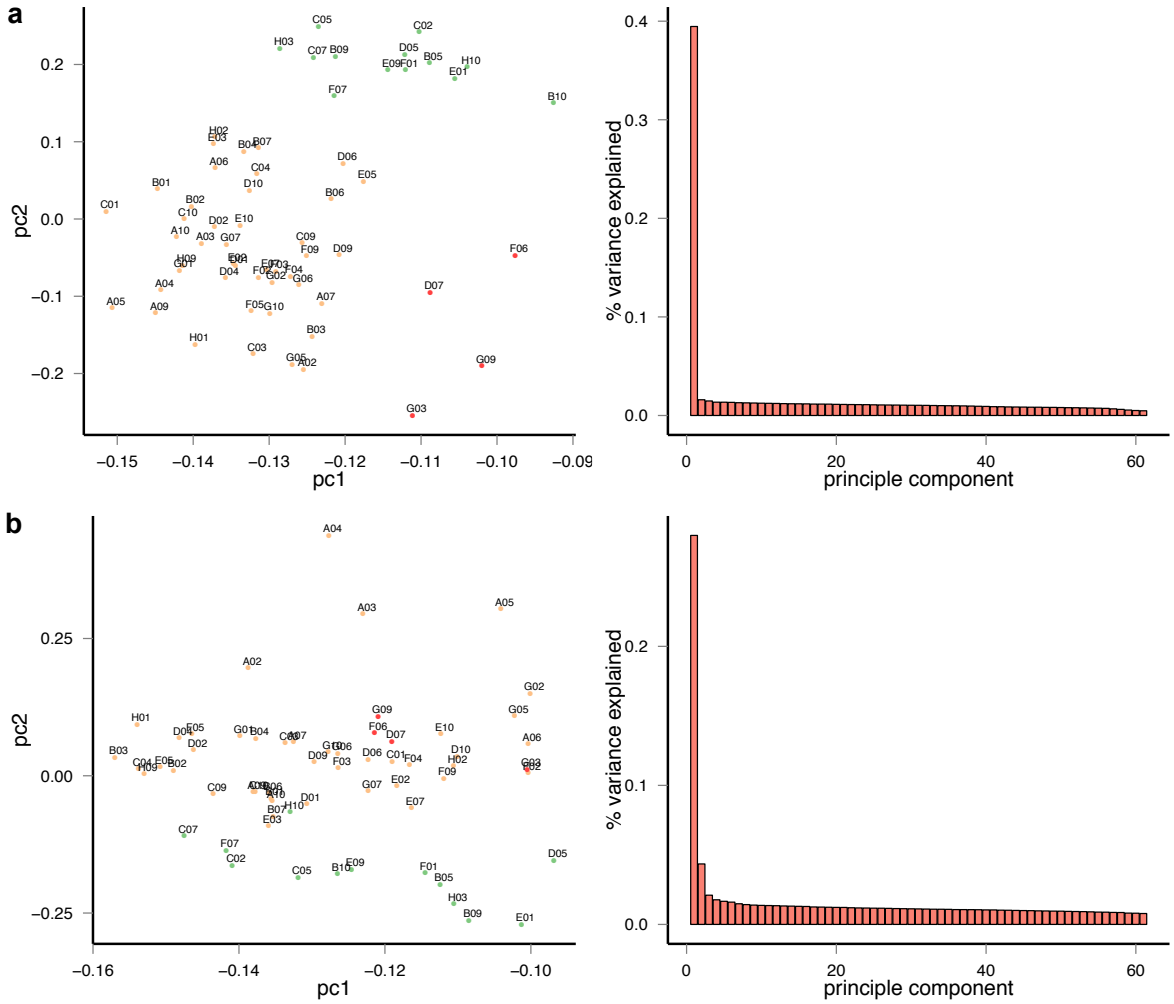


Figure A.6 Shown are projections onto first two principle components (left) alongside with percentage of variance explained by individual components (right) for both gene expression levels (a) and gene body methylation (b). Cells are color-coded based on clustering obtained using gene expression values, showing that that the methylation principal components partially recapitulate the structure in the expression data.

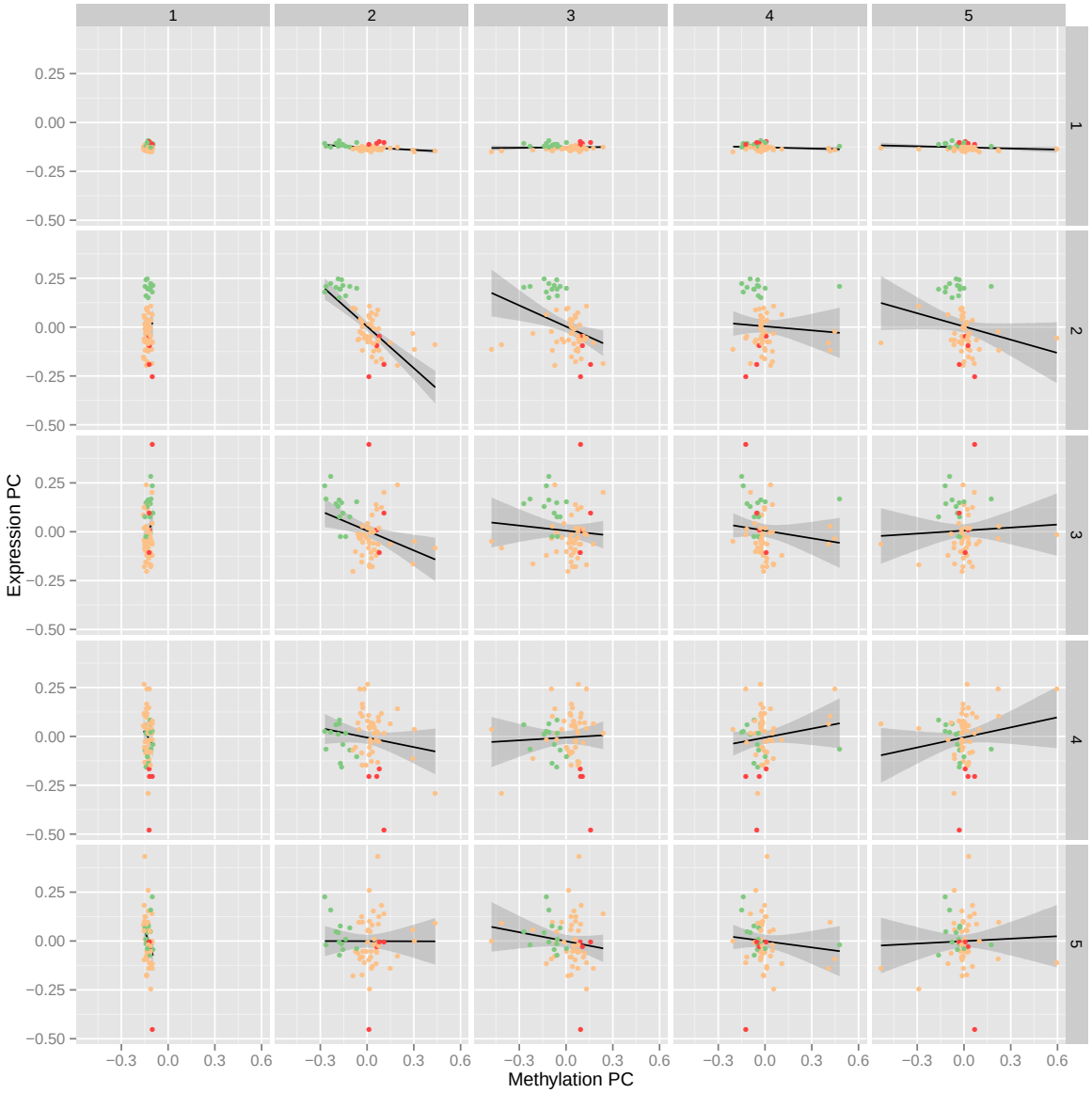


Figure A.7 Shown are scatter plots between individual principal components of gene expression levels (y-axis) and corresponding gene body methylation (x-axis), using 61 serum cells profiled using scM&T-seq. There is a strong correlation between the second principal component of DNA methylation and the corresponding component from gene expression, suggesting shared axes of variation between transcriptome and methylome profiles.

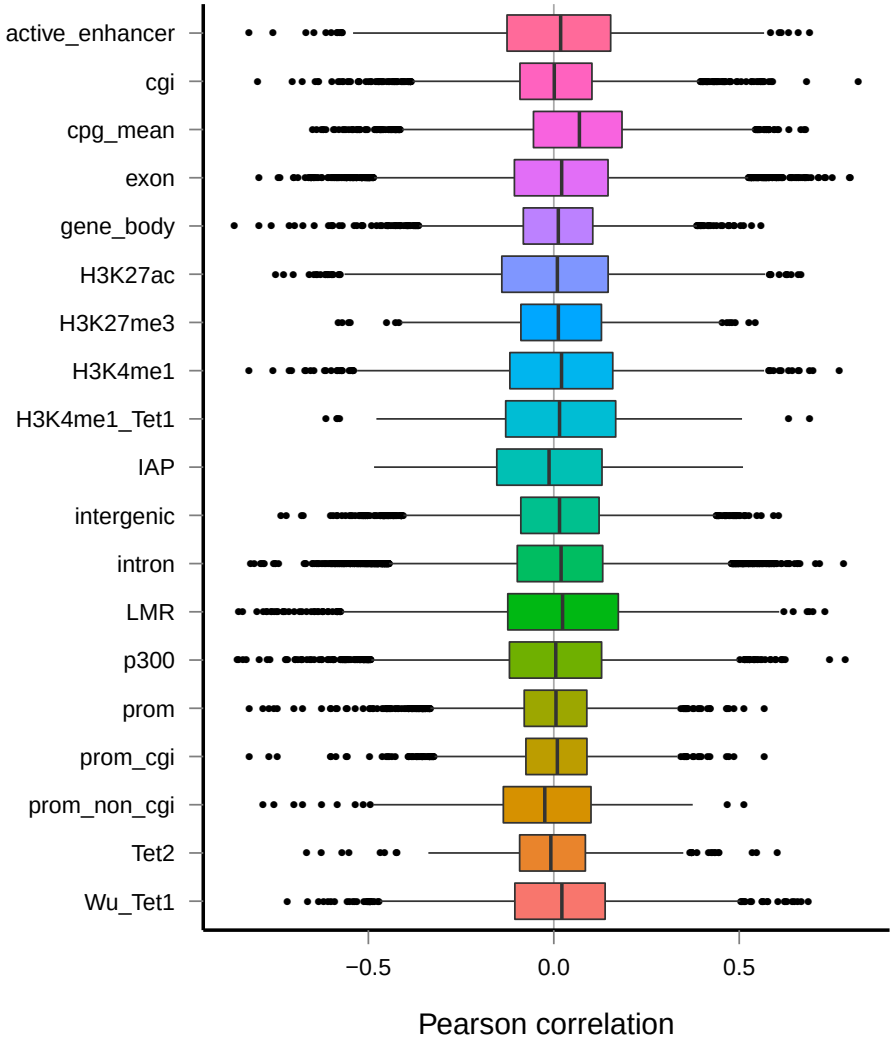


Figure A.8 Correlation coefficients for associations between DNA-methylation profiles in alternative genomic contexts and gene expression levels. Shown are boxplots of the correlation coefficient (Pearson's r) between DNA methylation in different genomic contexts and corresponding gene expression levels.

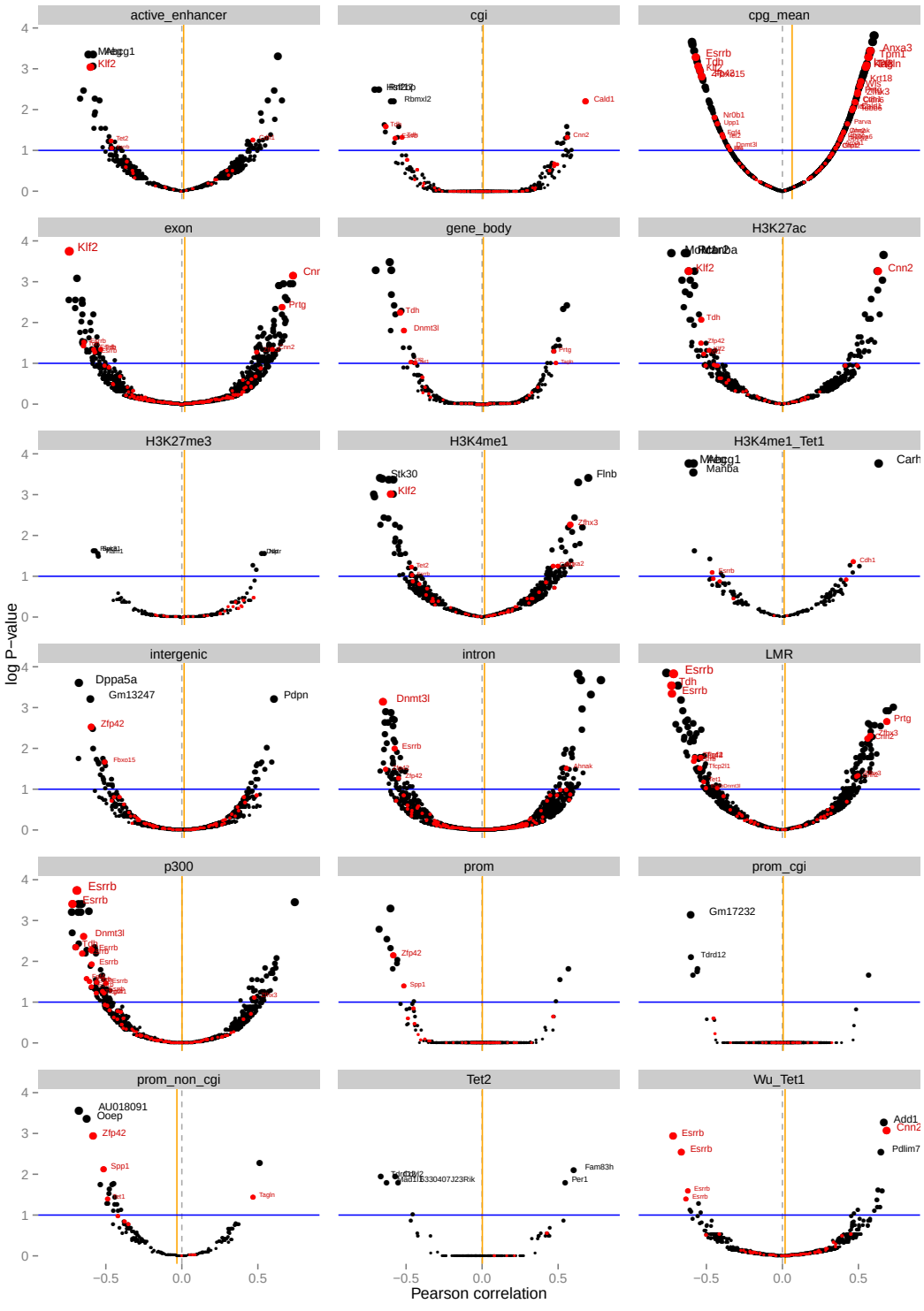


Figure A.9 Volcano plots for associations tests between DNA methylation profiles in alternative genomic context and gene expression levels. For each context, shown is the correlation coefficient (Pearson's r, x-axis) versus the adjusted p-value (Benjamini Hochberg adjustment; y-axis). The blue horizontal line corresponds to the FDR = 0.1 significance level. Each dot corresponds to a gene and the size for the adjusted p-value of the association test. Genes colored in red correspond to known pluripotency genes. The vertical orange line denotes the average correlation coefficient across all genes for a given annotation.

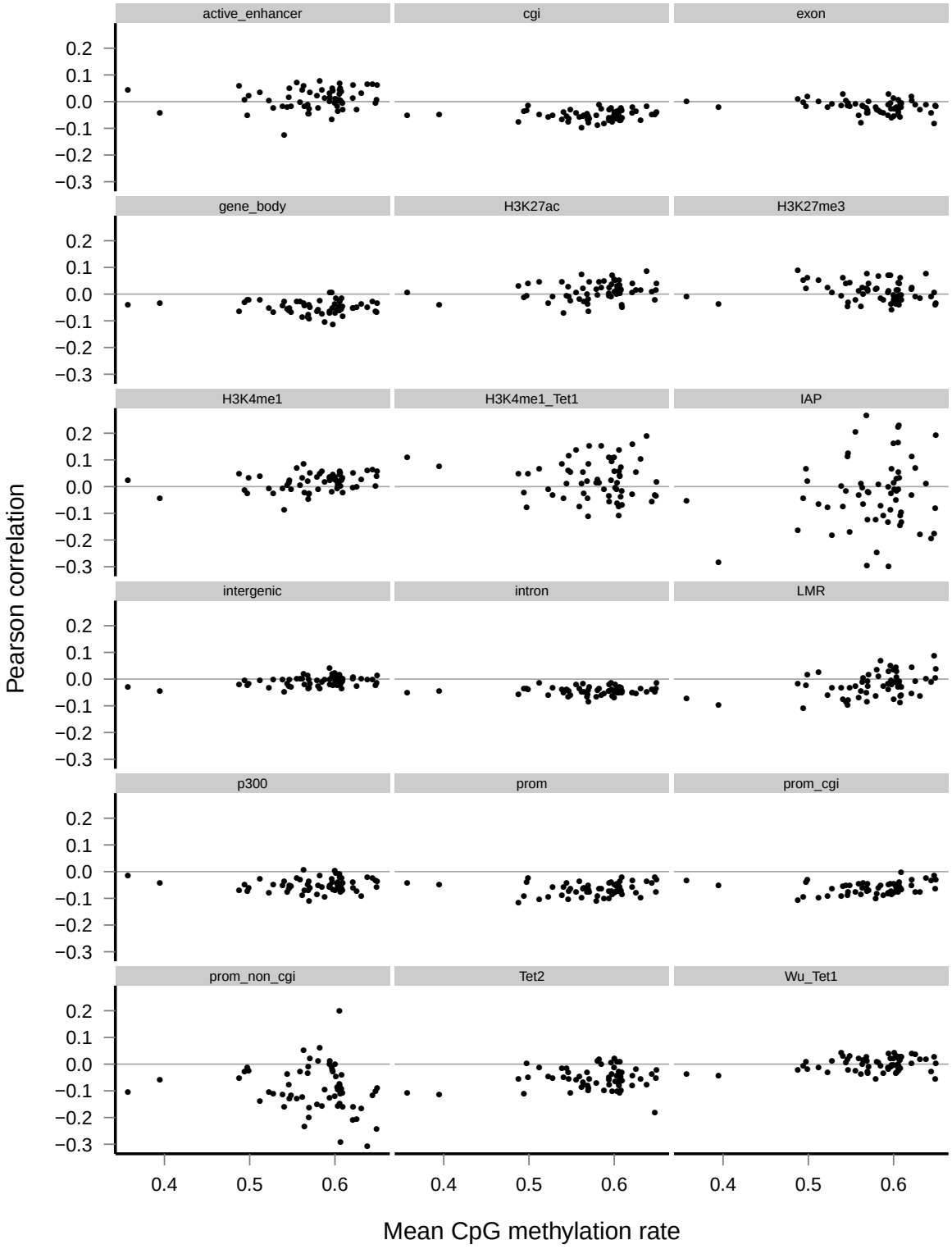


Figure A.10 Comparison of results of cell-specific correlation analysis with mean CpG methylation rate.

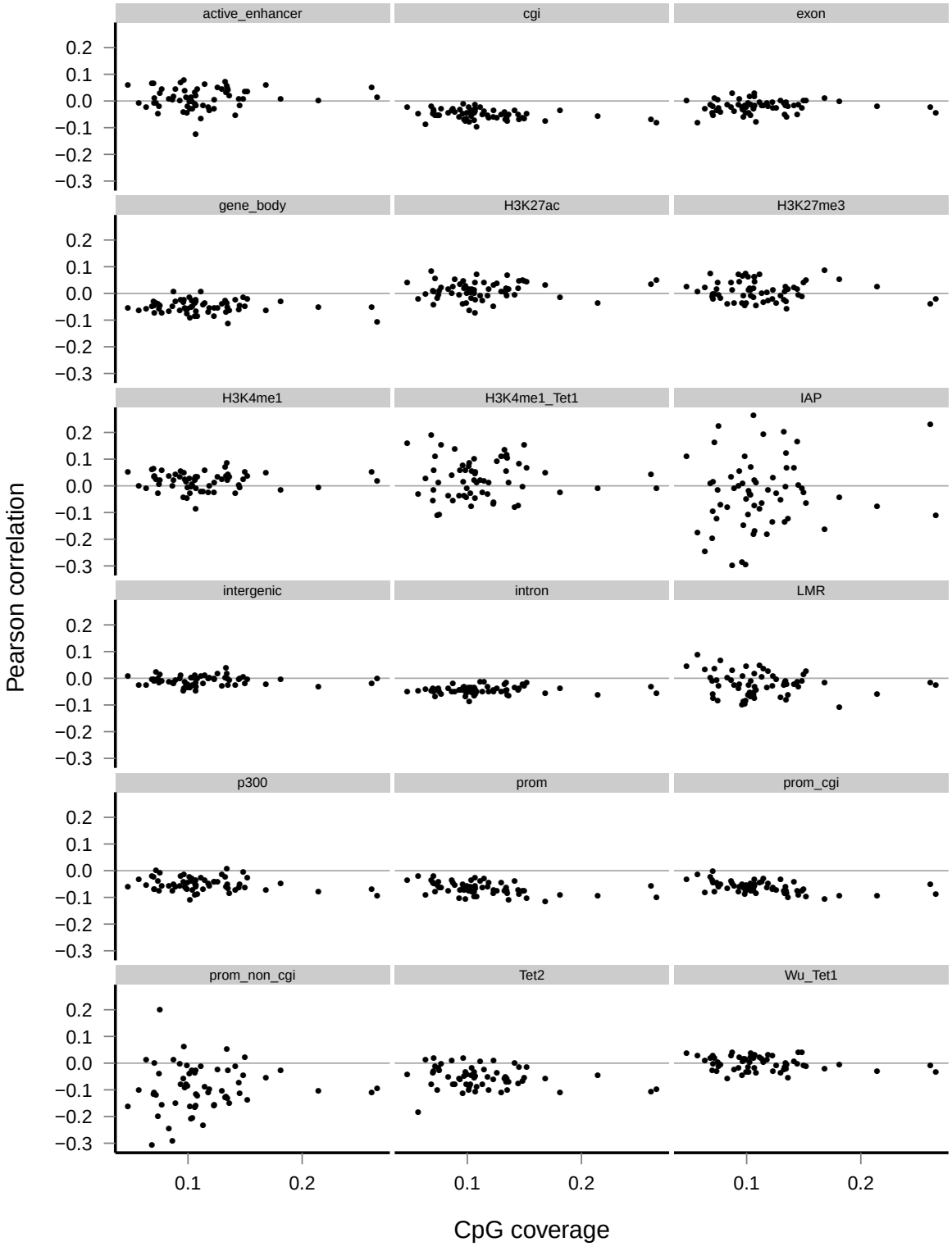


Figure A.11 Comparison of results of cell-specific correlation analysis with CpG coverage.

Appendix B

Deep neural networks for predicting DNA methylation

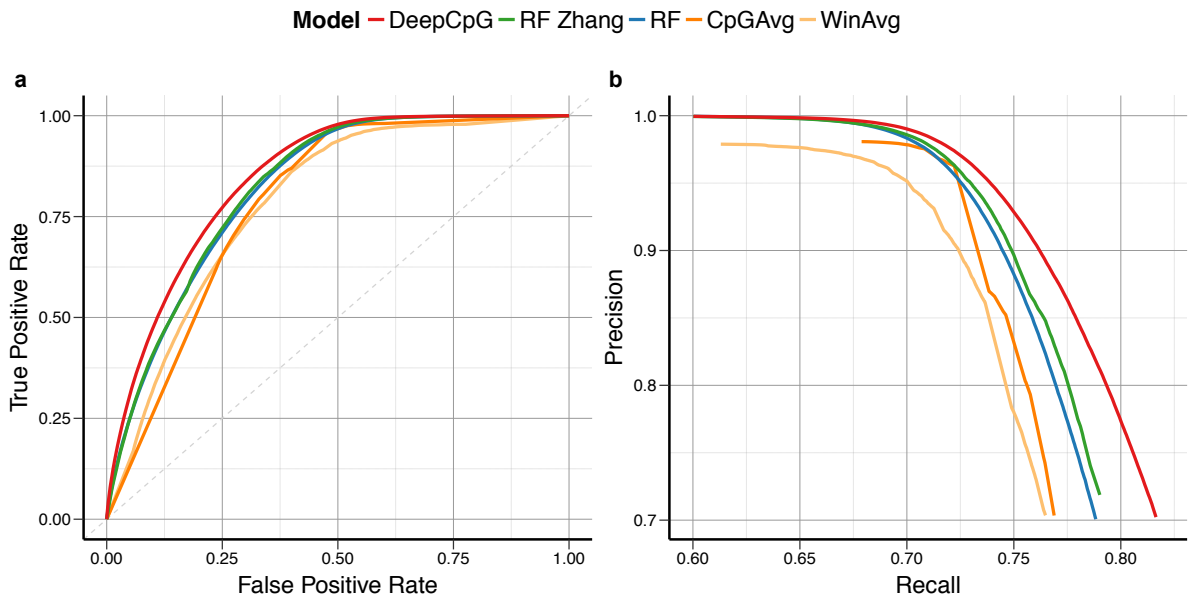


Figure B.1 Receiver operating characteristic and precision recall curves for predicting DNA methylation states using alternative methods. Receiver operating characterisic curve (a) and precision recall curve (b) for predicting methylation states in serum ESCs, analogous to results shown in main Figure 2. Considered were DeepCpG and random forest classifiers, either trained using similar features as DeepCpG (RF) or using additional DNA annotations (RF Zhang). Two baseline methods were considered, which estimate methylation states by averaging observed methylation states, either across consecutive 3 kb regions within individual cells (WinAvg), or across cells at a single CpG site (CpGAvg). Curves represent the average performance across cells.

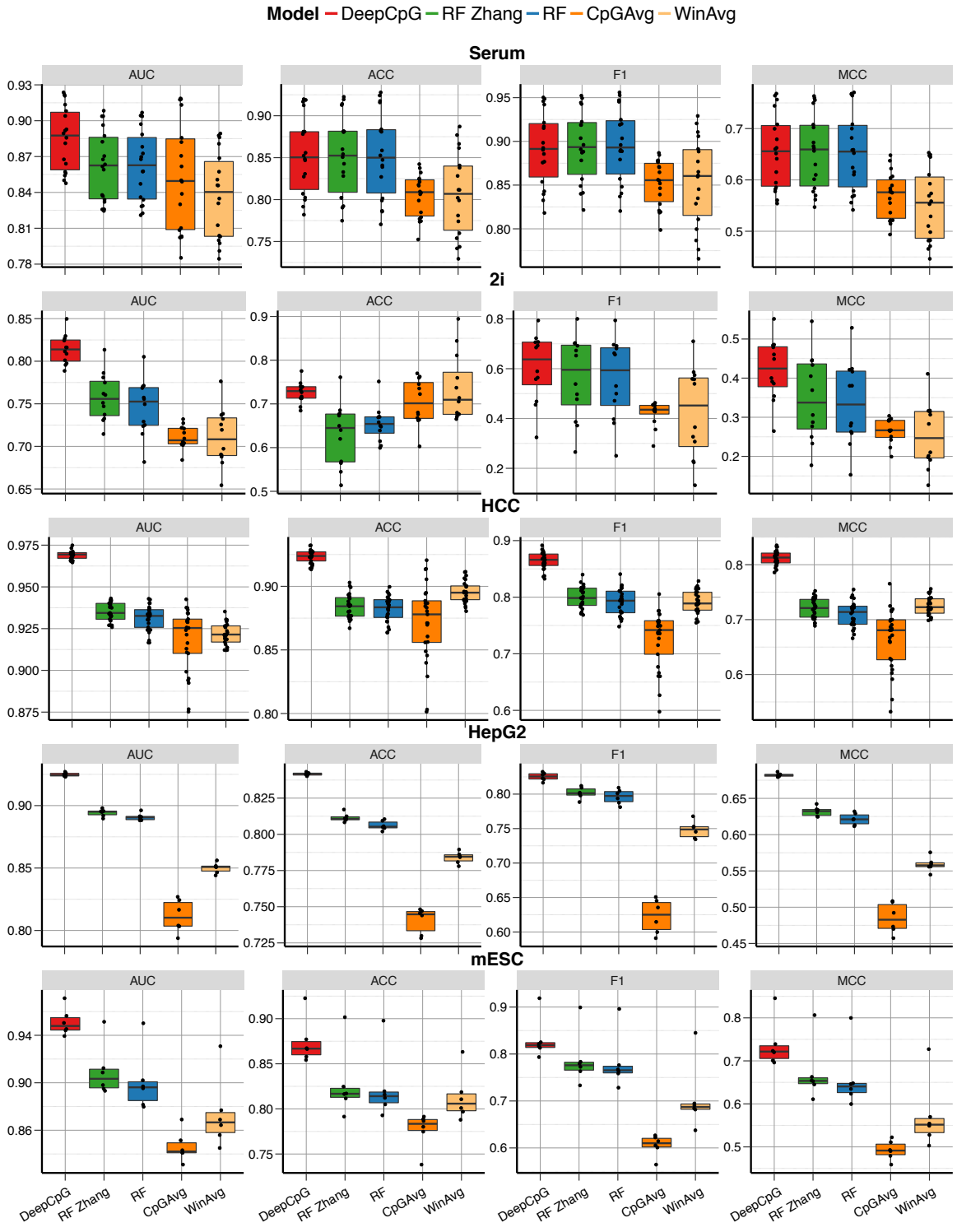


Figure B.2 Prediction performance of alternative methods and metrics across five datasets. Test prediction performance metrics for 18 serum and 12 2i mouse ESCs profiled using scBS-seq, as well as for cells profiled using scRRBS-seq, including 25 human HCC cells, 6 HepG2 cells, and 6 additional mouse ESCs. Shown is the prediction performance for alternative methods, considering the area under receiver operating characteristic curve (AUC), accuracy (ACC), F1 score (F1), or Matthews correlation coefficient (MCC).

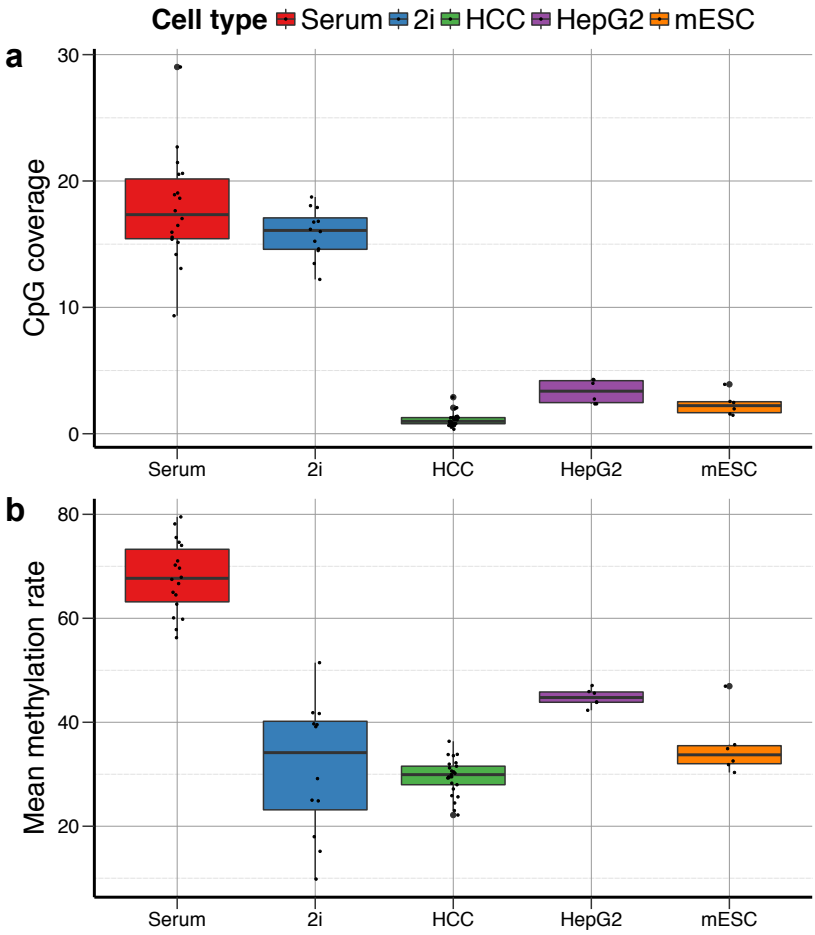


Figure B.3 Quality metrics of cells profiled using scBS-seq and scRRBS-seq. Genome wide CpG coverage (a) and mean methylation rate (b) of cells profiled using scBS-seq (Serum, 2i), and scRRBS-seq (HCC, HepG2, mESC).

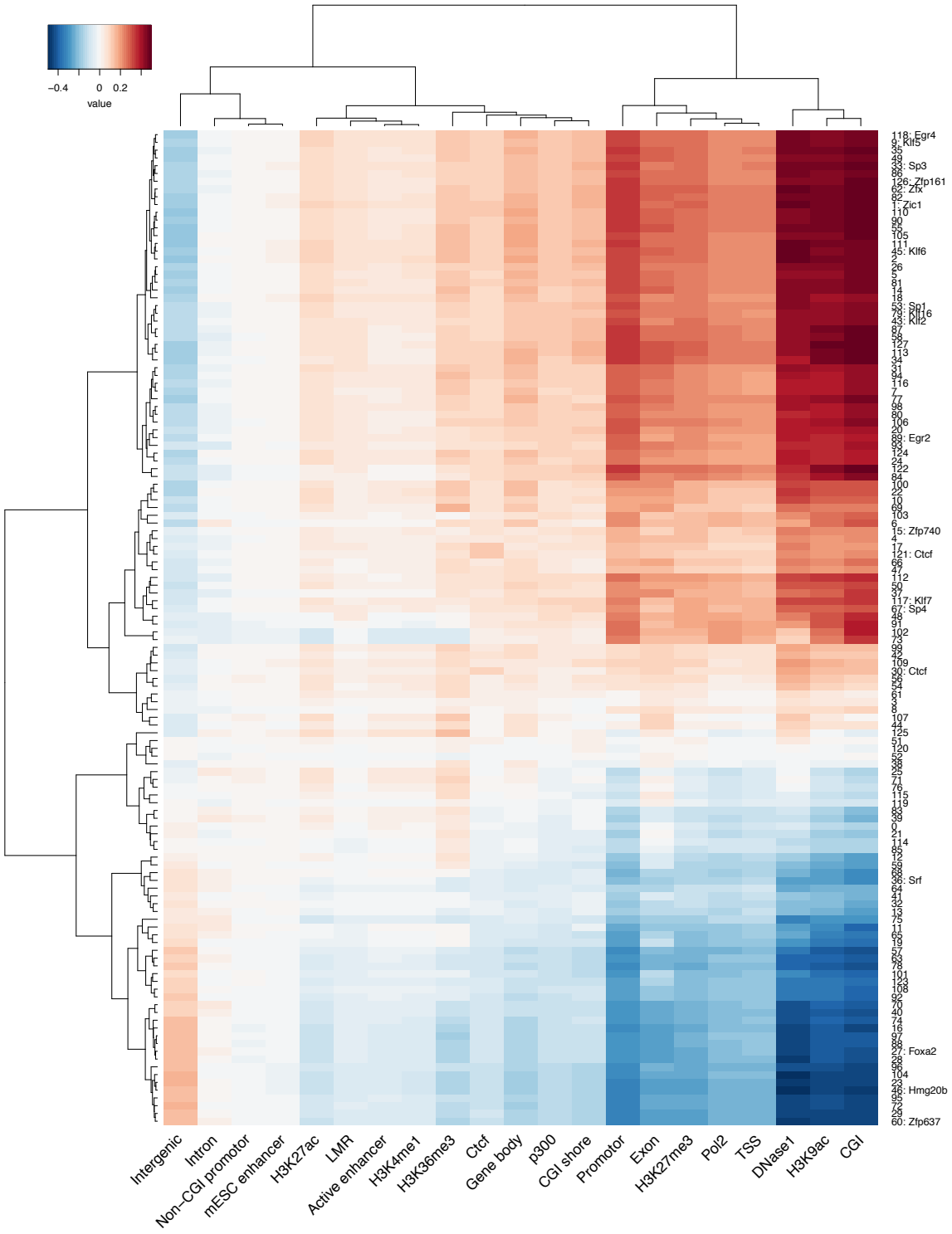


Figure B.4 Correlation between filter activities and higher-level sequence features. Spearman correlation between the activity of filters of the first convolutional layer of the DNA module (motifs) and higher-level sequence features considered in Zhang et al. Activities were strongly correlated with DNase1 hypersensitive sites, histone modification marks, and CpG dense genomic context. This indicates that DeepCpG is able to automatically learn higher-level features from the raw DNA sequence.

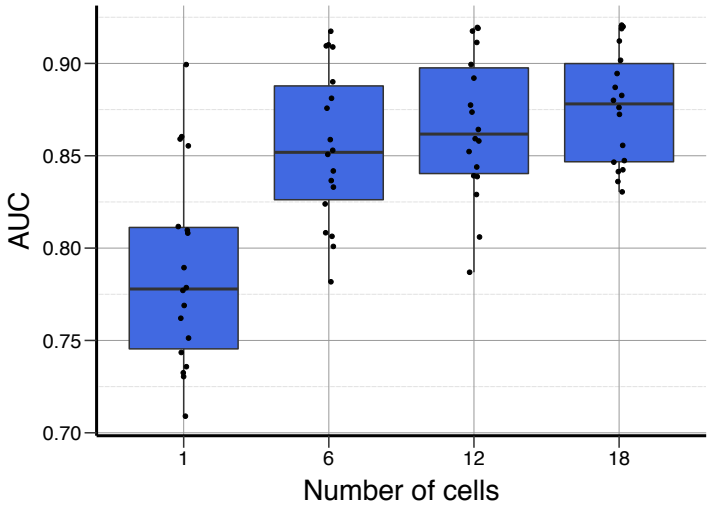


Figure B.5 Prediction performance of the DeepCpG CpG module depending on the number of cells. Test AUC for serum mouse ESCs, using an increasing number of cells as input for the DeepCpG CpG module.

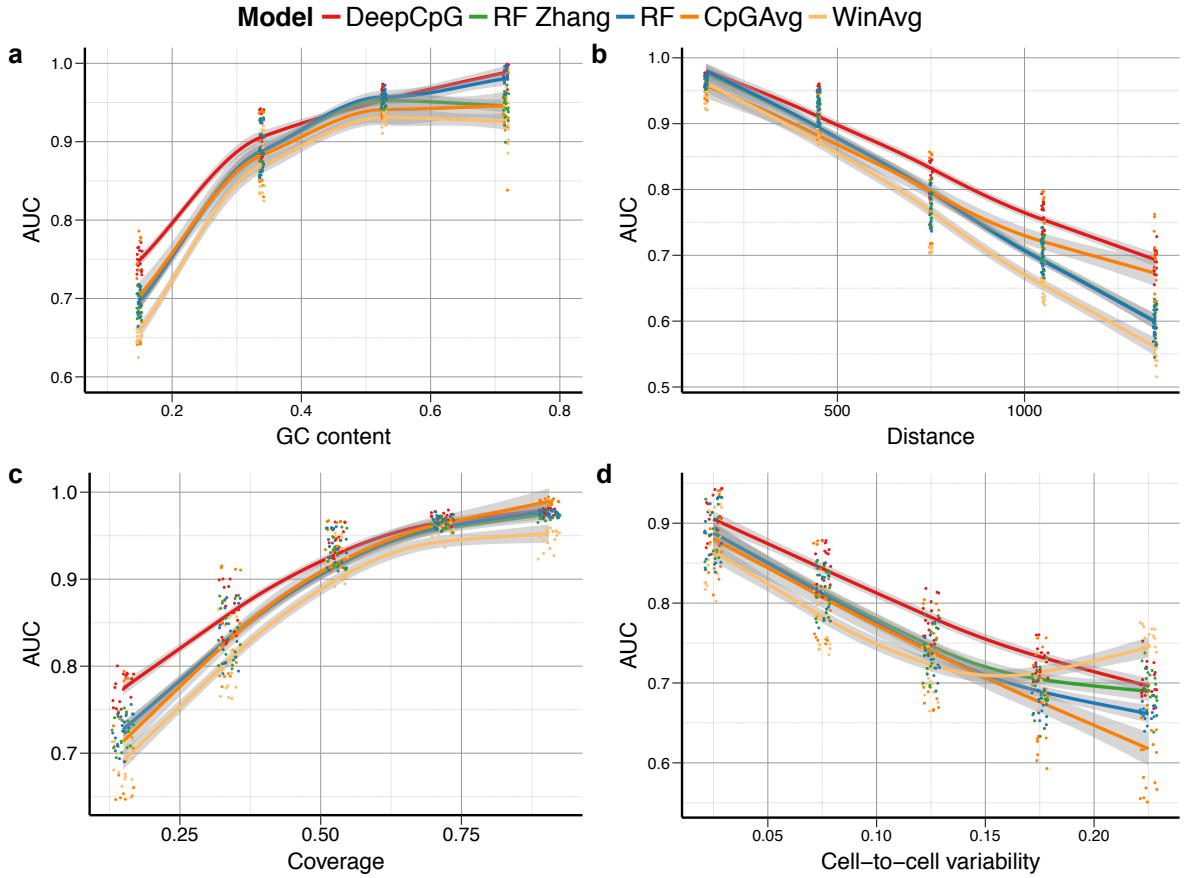


Figure B.6 Prediction performances stratified by different metrics. Test AUC for serum mouse ESCs, considering alternative methods, stratified by (a) GC content, (b) distance to neighboring CpG sites, (c) fraction of cells by which the target CpG site is covered, (d) cell-to-cell variability within 3 kb windows centered on the target CpG site. Trend lines were fit to observed coverage levels using local polynomial regression (LOESS), with shaded areas corresponding to 95% confidence intervals.

Appendix C

Analysis of deep neural networks for predicting DNA methylation

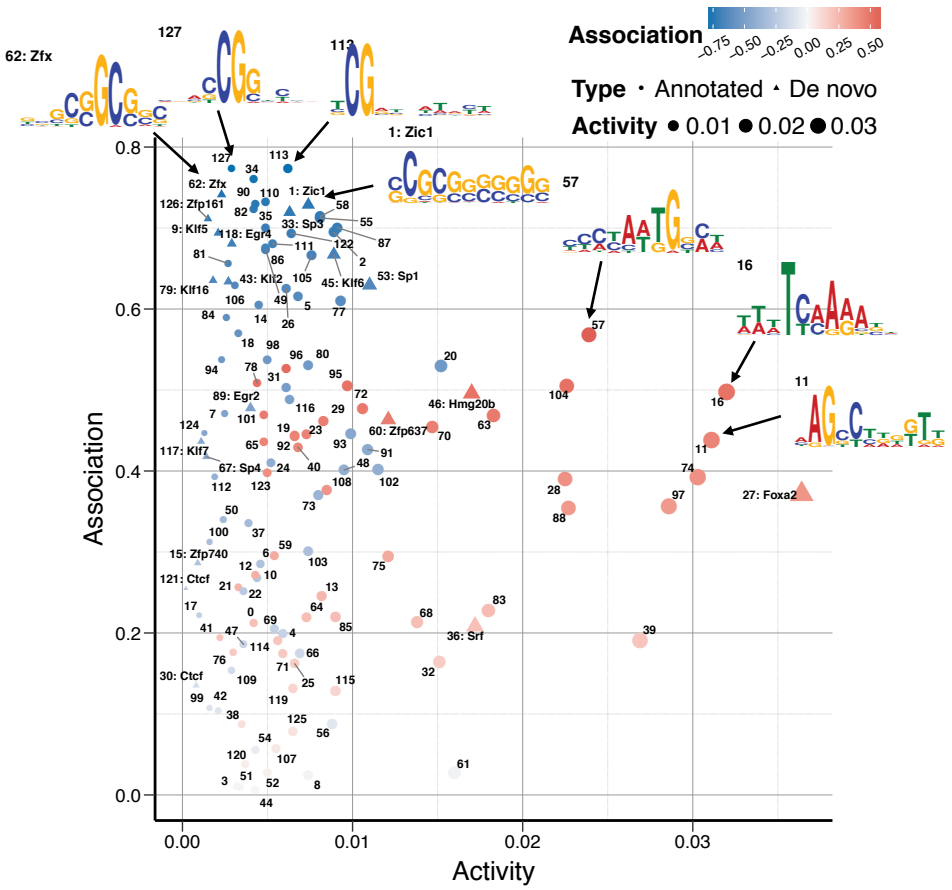


Figure C.1 Importance of the DNA sequence motifs. Average motif activity on the x-axis vs. the absolute estimated motif effect on methylation (association) on the y-axis. Marker size and colour correspond to the activity of motifs, and their association with methylation states, respectively. Triangles denote annotated motifs (matched to motif in CIS-BP or UniPROPE database; FDR < 0.05); circles denote de novo motifs.

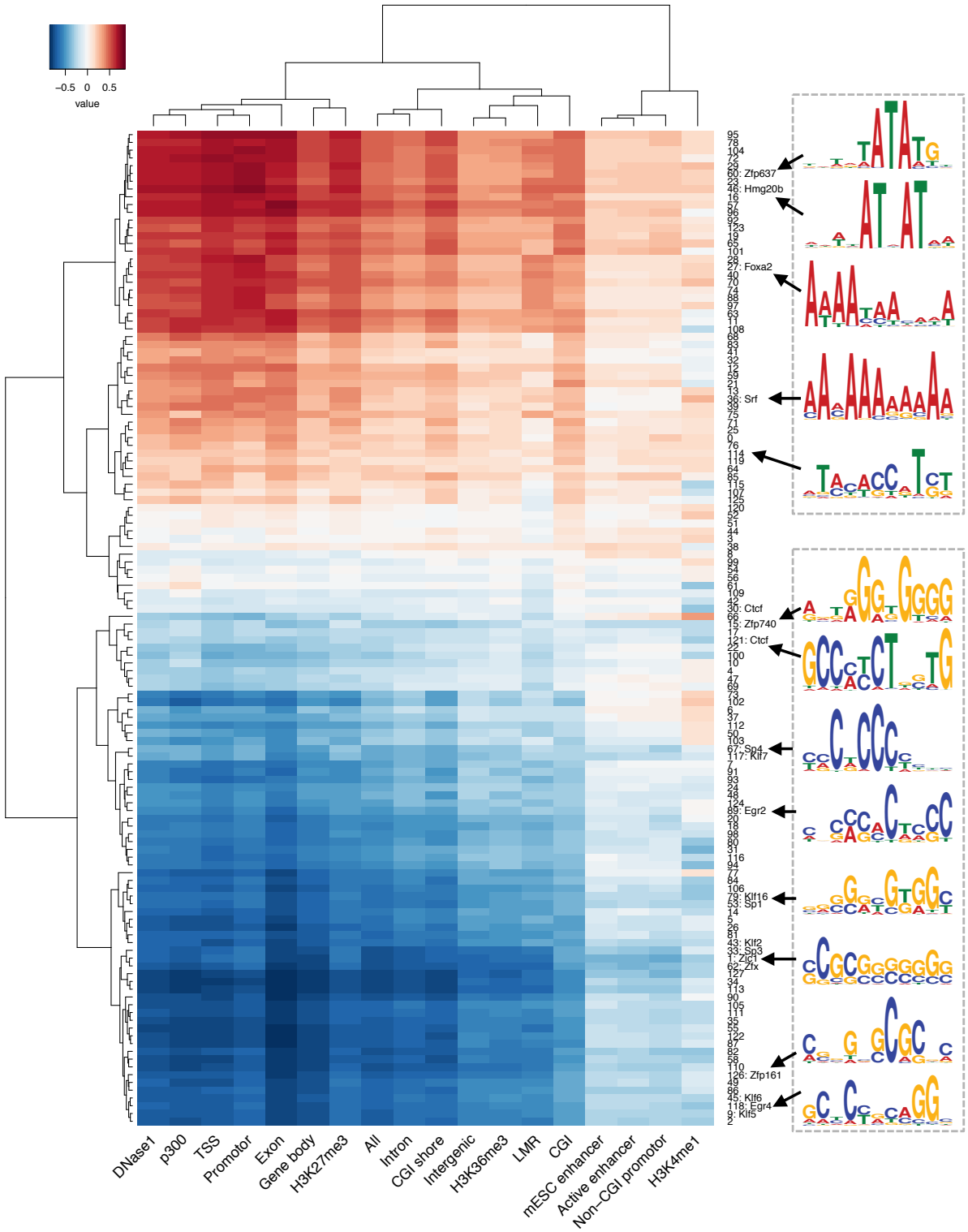


Figure C.2 Effect of DNA sequence motifs on methylation. Effect of discovered motifs on CpG methylation in different genomic contexts on test chromosomes, quantified by Pearson correlation between motif activities and predicted methylation level. Motifs cluster into CG-rich, methylation-decreasing motifs, and AT-rich methylation-increasing motifs.

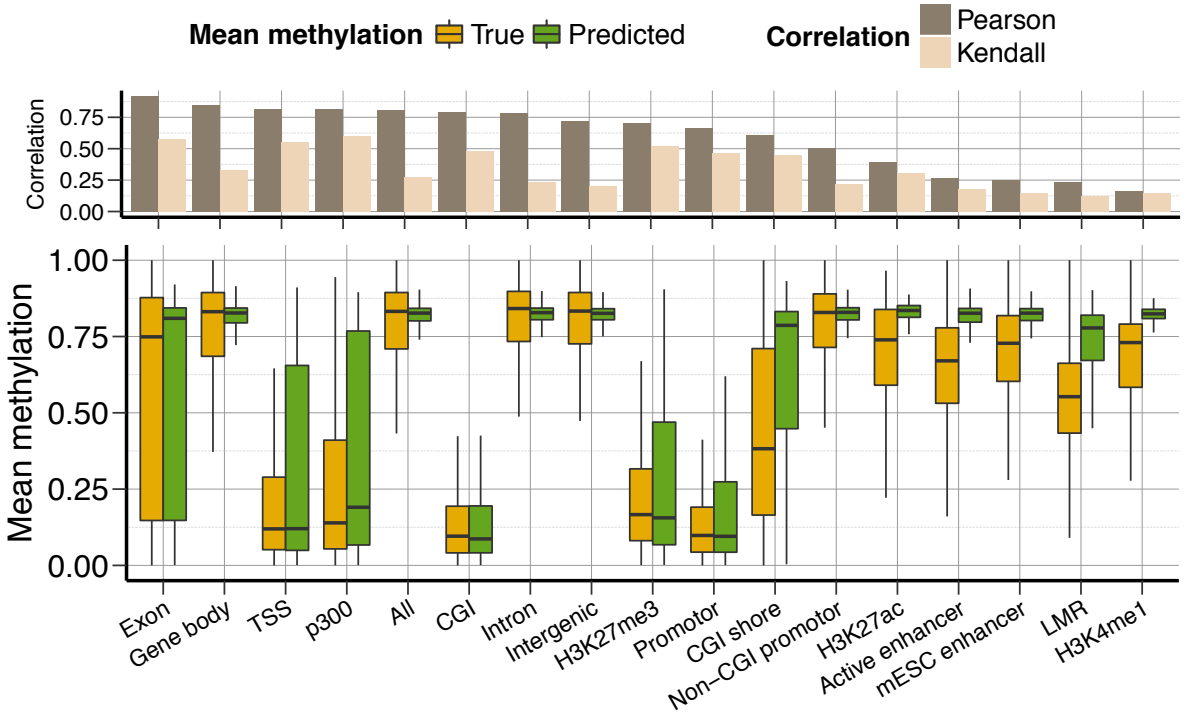


Figure C.3 Performance of DeepCpG DNA module to predict mean methylation levels. Boxplot represent predicted (green) and the observed (orange) mean methylation levels in 3 kbp windows centred on individual CpG sites for different genomic contexts on held out test chromosomes. Barplot represent Pearson and Kendall correlation coefficients.

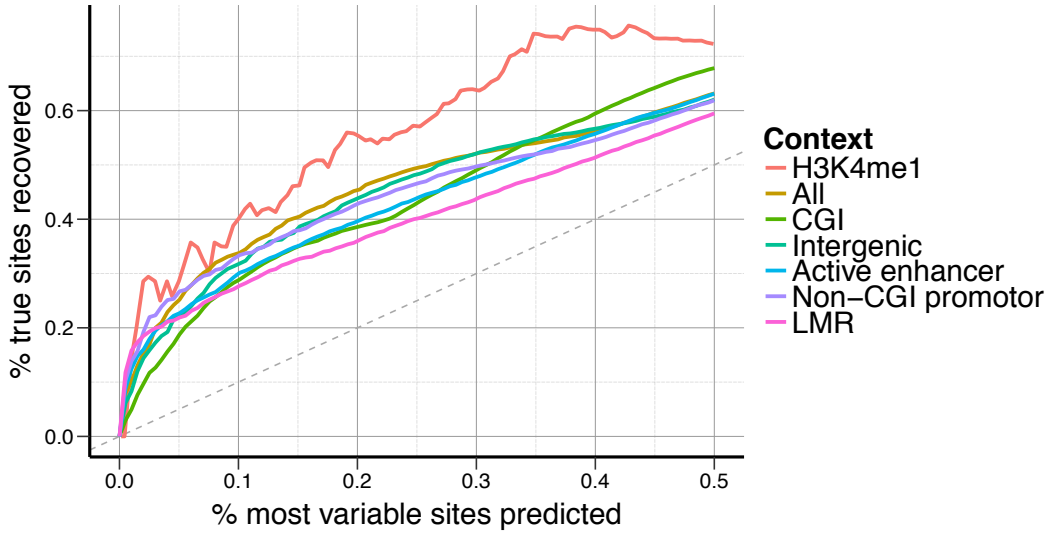


Figure C.4 Sensitivity of discovering highly-variable CpG sites. Sensitivity of discovering most variable sites for different thresholds and genomic contexts on test chromosomes. Individual CpG sites were ranked by the empirical variance estimated in 3 kbp windows centred on the CpG sites, or the variance predicted by DeepCpG, and the overlap computed for the fraction of most variables sites shown on the x-axis. The dashed line indicates the performance of a random ranking.

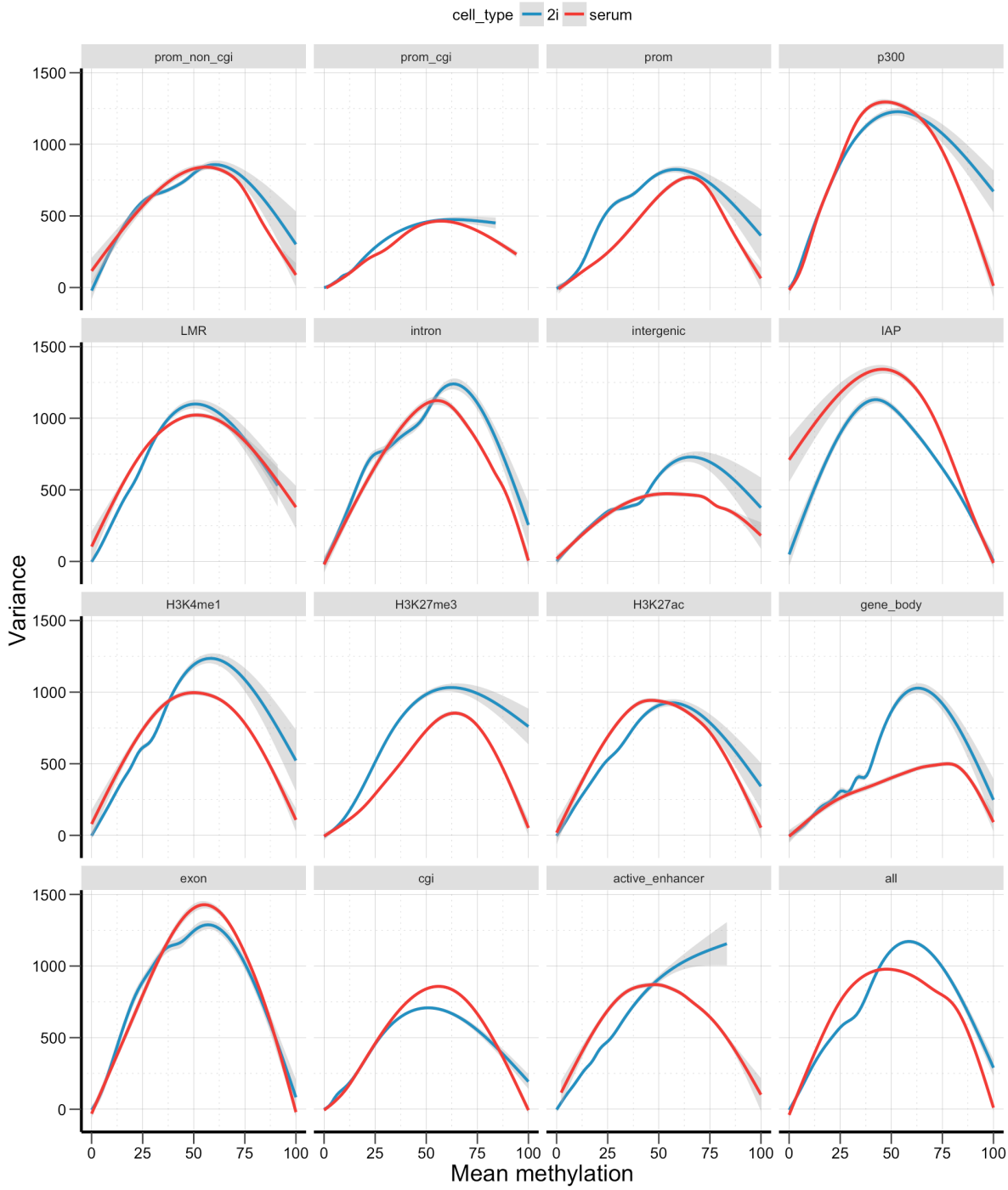


Figure C.5 Dependency between mean methylation levels and cell-to-cell variance. Smoothed regression fit between mean methylation levels (x-axis) verus cell-to-cell variance (y-axis), for 2i and serum cells in different genomic contexts. Cell-to-cell variance is linked to the mean methylation level, and highest for an intermediate methylation level of 50%.

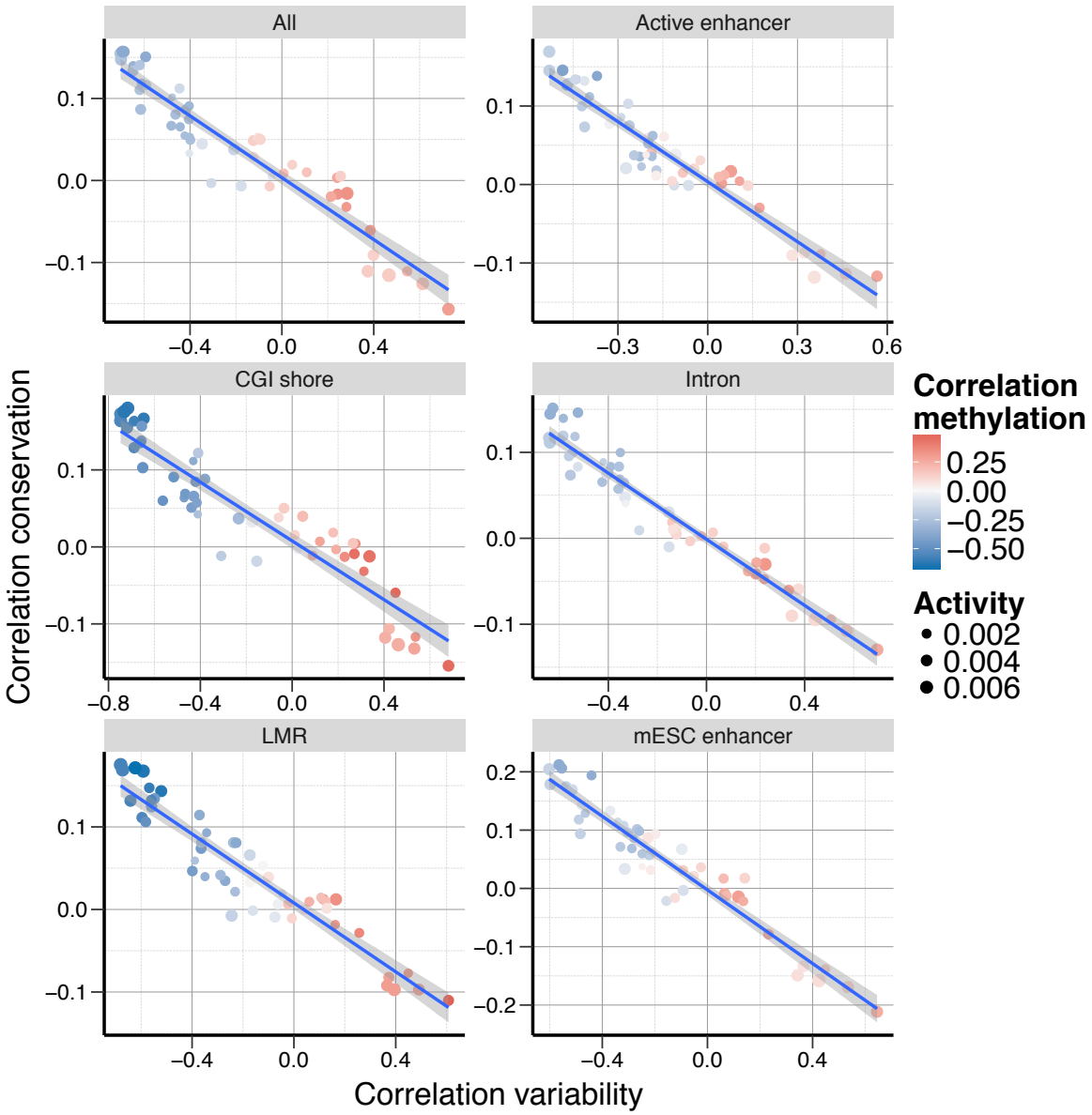


Figure C.6 Linkage between motif correlation with sequence conservation and cell-to-cell variability. Correlation of motif activities with cell-to-cell variability (x-axis), and DNA conservation (y-axis). Variability increasing motifs are most active in non-conserved regions. Individual dots correspond to motifs with their mean activity represented by the size, and influence on CpG methylation by colour.

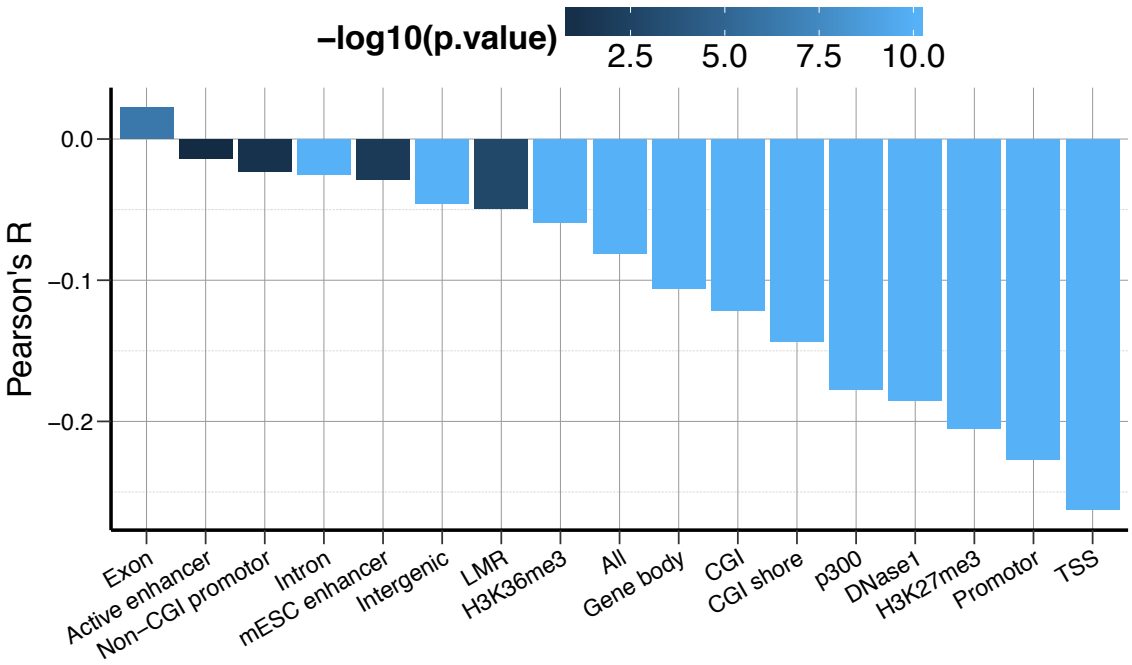


Figure C.7 Correlation between estimated mutation effects and DNA sequence conservation. Correlation between the estimated effect of single nucleotide mutations and PhastCons conservation score for alternative contexts on test chromosomes. Estimated effects are significantly anti-correlated overall ('All', $P < 1.0 \times 10^{-15}$) in CpG dense regions.

References

[1] Martın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, and Matthieu Devin. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *arXiv*, 2016.

[2] Michalis Agathocleous, Georgia Christodoulou, Vasilis Promponas, Chris Christodoulou, Vassilis Vassiliades, and Antonis Antoniou. Protein secondary structure prediction with bidirectional recurrent neural nets: Can weight updating for each residue enhance performance? In *Artificial Intelligence Applications and Innovations*, pages 128–137. Springer, 2010. ISBN 3-642-16238-X.

[3] Guillaume Alain, Yoshua Bengio, and Salah Rifai. Regularized auto-encoders estimate local statistics. pages 1–17, 2012.

[4] F. W. Albert, S. Treusch, A. H. Shockley, J. S. Bloom, and L. Kruglyak. Genetics of single-cell protein abundance variation in large yeast populations. *Nature*, 506(7489): 494–7, February 2014. doi: 10.1038/nature12904.

[5] Babak Alipanahi, Andrew Delong, Matthew T. Weirauch, and Brendan J. Frey. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nature Biotechnology*, 33:831–838, August 2015. doi: 10.1038/nbt.3300.

[6] Galen Andrew, Raman Arora, Jeff A. Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *ICML*, pages 1247–1255, 2013.

[7] Christof Angermueller, Stephen J. Clark, Heather J. Lee, Iain C. Macaulay, Mabel J. Teng, Tim Xiaoming Hu, Felix Krueger, Sébastien A. Smallwood, Chris P. Ponting, Thierry Voet, Gavin Kelsey, Oliver Stegle, and Wolf Reik. Parallel single-cell sequencing links transcriptional and epigenetic heterogeneity. *Nature Methods*, 13(3):229–232, January 2016. doi: 10.1038/nmeth.3728.

[8] Christof Angermueller, Tanel Pärnamaa, Leopold Parts, and Oliver Stegle. Deep learning for computational biology. *Molecular Systems Biology*, 12(7):878, July 2016. doi: 10.15252/msb.20156651.

[9] Christof Angermueller, Heather Lee, Wolf Reik, and Oliver Stegle. Accurate prediction of single-cell DNA methylation states using deep learning. *bioRxiv*, February 2017. doi: 10.1101/055715.

[10] S. Arsenian, B. Weinhold, M. Oelgeschläger, U. Rüther, and A. Nordheim. Serum response factor is essential for mesoderm formation during mouse embryogenesis. *The EMBO Journal*, 17(21):6289–6299, November 1998. doi: 10.1093/emboj/17.21.6289.

[11] Ehsaneddin Asgari and Mohammad R. K. Mofrad. ProtVec: A Continuous Distributed Representation of Biological Sequences. *PLOS ONE*, 10, November 2015. doi: 10.1371/journal.pone.0141287.

[12] S. Bach, A. Binder, G. Montavon, F. Klauschen, K. R. Muller, and W. Samek. On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PloS one*, 10(7), 2015. doi: 10.1371/journal.pone.0130140.

[13] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv*, 2014.

[14] Timothy L. Bailey, Mikael Boden, Fabian A. Buske, Martin Frith, Charles E. Grant, Luca Clementi, Jingyuan Ren, Wilfred W. Li, and William S. Noble. MEME Suite: Tools for motif discovery and searching. *Nucleic Acids Research*, 37(suppl 2), January 2009. doi: 10.1093/nar/gkp335.

[15] Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, James Bergstra, Ian Goodfellow, Arnaud Bergeron, Nicolas Bouchard, David Warde-Farley, and Yoshua Bengio. Theano: New features and speed improvements. *arXiv*, 2012.

[16] A. Battle, Z. Khan, S. H. Wang, A. Mitrano, M. J. Ford, J. K. Pritchard, and Y. Gilad. Genomic variation. Impact of regulatory variation from RNA to protein. *Science*, 347 (6222):664–7, February 2015. doi: 10.1126/science.1260793.

[17] Justin Bayer and Christian Osendorfer. Learning Stochastic Recurrent Networks. *arXiv*, November 2014.

[18] Jordana T Bell, Athma A Pai, Joseph K Pickrell, Daniel J Gaffney, Roger Pique-Regi, Jacob F Degner, Yoav Gilad, and Jonathan K Pritchard. DNA methylation patterns associate with genetic and gene expression variation in HapMap cell lines. *Genome Biol*, 12(1), 2011.

[19] Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. In *Neural Networks: Tricks of the Trade*, pages 437–478. Springer, 2012. ISBN 3-642-35288-X.

[20] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks. volume 19, page 153, 2007. ISBN 1049-5258.

[21] Yoshua Bengio, Aaron Courville, and Pierre Vincent. Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(8):1798–1828, 2013.

[22] Adrian Benton, Huda Khayrallah, Biman Gujral, Drew Reisinger, Sheng Zhang, and Raman Arora. Deep Generalized Canonical Correlation Analysis. *arXiv*, February 2017.

[23] James Bergstra and David D. Cox. Hyperparameter Optimization and Boosting for Classifying Facial Expressions: How good can a "Null" Model be? *arXiv*, 2013.

[24] Timothy H. Bestor, John R. Edwards, and Mathieu Boulard. Notes on the role of dynamic DNA methylation in mammalian development. *Proceedings of the National Academy of Sciences*, 112(22):6796–6799, February 2015. doi: 10.1073/pnas.1415301111.

[25] Manoj Bhasin, Hong Zhang, Ellis L. Reinherz, and Pedro A. Reche. Prediction of methylated CpGs in DNA sequences using a support vector machine. *FEBS Letters*, 579 (20):4302–4308, August 2005. doi: 10.1016/j.febslet.2005.07.002.

[26] Marina Bibikova, Bret Barnes, Chan Tsan, Vincent Ho, Brandy Klotzle, Jennie M. Le, David Delano, Lu Zhang, Gary P. Schroth, Kevin L. Gunderson, Jian-Bing Fan, and Richard Shen. High density DNA methylation array with single CpG site resolution. *Genomics*, 98(4):288–295, October 2011. doi: 10.1016/j.ygeno.2011.07.007.

[27] Adrian Bird. DNA methylation patterns and epigenetic memory. *Genes & Development*, 16(1):6–21, January 2002. doi: 10.1101/gad.947102.

[28] Christopher M Bishop. Mixture density networks, 1994.

[29] E. M. Blackwood and J. T. Kadonaga. Going the distance: A current view of enhancer action. *Science (New York, N.Y.)*, 281(5373):60–63, July 1998.

[30] Christoph Bock, Martina Paulsen, Sascha Tierling, Thomas Mikeska, Thomas Lengauer, and Jörn Walter. CpG Island Methylation in Human Lymphocytes Is Highly Correlated with DNA Sequence, Repeats, and Predicted DNA Structure. *PLoS Genetics*, 2(3), March 2006. doi: 10.1371/journal.pgen.0020026.

[31] Christoph Bock, Isabel Beerman, Wen-Hui Lien, Zachary D. Smith, Hongcang Gu, Patrick Boyle, Andreas Gnirke, Elaine Fuchs, Derrick J. Rossi, and Alexander Meissner. DNA Methylation Dynamics during In Vivo Differentiation of Blood and Skin Stem Cells. *Molecular Cell*, 47(4):633–647, August 2012. doi: 10.1016/j.molcel.2012.06.019.

[32] Guillaume Bouchard, Dawei Yin, and Shengbo Guo. Convex collective matrix factorization. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, pages 144–152, 2013.

[33] Nicolas Boulanger-Lewandowski, Yoshua Bengio, and Pascal Vincent. Modeling Temporal Dependencies in High-Dimensional Sequences: Application to Polyphonic Music Generation and Transcription. *arXiv*, June 2012.

[34] Nicolas Boulanger-Lewandowski, Yoshua Bengio, and Pierre Vincent. High-dimensional sequence transduction. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference On*, pages 3178–3182. IEEE, 2013.

[35] R. Bourgon, R. Gentleman, and W. Huber. Independent filtering increases detection power for high-throughput experiments. *Proceedings of the National Academy of Sciences*, 107(21):9546–9551, May 2010. doi: 10.1073/pnas.0914005107.

[36] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[37] Arie B. Brinkman, Femke Simmer, Kelong Ma, Anita Kaan, Jingde Zhu, and Hendrik G. Stunnenberg. Whole-genome DNA methylation profiling using MethylCap-seq. *Methods*, 52(3):232–236, November 2010. doi: 10.1016/j.ymeth.2010.06.012.

[38] B. Burgess-Beusse, C. Farrell, M. Gaszner, M. Litt, V. Mutskov, F. Recillas-Targa, M. Simpson, A. West, and G. Felsenfeld. The insulation of genes from external enhancers and silencing chromatin. *Proceedings of the National Academy of Sciences*, 99:16433–16437, December 2002. doi: 10.1073/pnas.162342499.

[39] Bokai Cao, Hucheng Zhou, and Philip S. Yu. Multi-view Machines. *arXiv*, June 2015.

[40] Sarath Chandar, Mitesh M. Khapra, Hugo Larochelle, and Balaraman Ravindran. Correlational Neural Networks. *arXiv*, April 2015.

[41] Zhengping Che, Sanjay Purushotham, Robinder Khemani, and Yan Liu. Distilling Knowledge from Deep Networks with Applications to Healthcare Domain. *arXiv*, 2015.

[42] C. Cheng, K. K. Yan, K. Y. Yip, J. Rozowsky, R. Alexander, C. Shou, and M. Gerstein. A statistical framework for modeling gene expression using chromatin features and application to modENCODE datasets. *Genome Biol*, 12(2), 2011. doi: 10.1186/gb-2011-12-2-r15.

[43] Kyunghyun Cho, Aaron Courville, and Yoshua Bengio. Describing Multimedia Content using Attention-based Encoder–Decoder Networks. *arXiv*, July 2015.

[44] François Chollet. Keras: Theano-based deep learning library.

[45] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv*, December 2014.

[46] Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron Courville, and Yoshua Bengio. A Recurrent Latent Variable Model for Sequential Data. *arXiv*, June 2015.

[47] Dan Ciresan, Alessandro Giusti, Luca M Gambardella, and Jürgen Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. pages 2843–2851, 2012.

[48] Dan C Cireşan, Alessandro Giusti, Luca M Gambardella, and Jürgen Schmidhuber. Mitosis detection in breast cancer histology images with deep neural networks. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013*, pages 411–418. Springer, 2013. ISBN 3-642-40762-5.

[49] Ronan Collobert, Koray Kavukcuoglu, and Clément Farabet. Torch7: A matlab-like environment for machine learning. 2011. EPFL-CONF-192376.

[50] G. E. Crooks. WebLogo: A Sequence Logo Generator. *Genome Research*, 14(6): 1188–1190, May 2004. doi: 10.1101/gr.849004.

[51] George E. Dahl, Navdeep Jaitly, and Ruslan Salakhutdinov. Multi-task Neural Networks for QSAR Predictions. *arXiv*, June 2014.

[52] Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. pages 2933–2941, 2014.

[53] E. de Wit and W. de Laat. A decade of 3C technologies: Insights into nuclear organization. *Genes & Development*, 26(1):11–24, January 2012. doi: 10.1101/gad.179804.111.

[54] Li Deng and Roberto Togneri. Deep dynamic models for learning hidden representations of speech features. In *Speech and Audio Processing for Coding, Enhancement and Recognition*, pages 153–195. Springer, 2015.

[55] Q. Deng, D. Ramskold, B. Reinius, and R. Sandberg. Single-Cell RNA-Seq Reveals Dynamic, Random Monoallelic Gene Expression in Mammalian Cells. *Science*, 343 (6167):193–196, January 2014. doi: 10.1126/science.1245316.

[56] Siddharth S. Dey, Lennart Kester, Bastiaan Spanjaard, Magda Bienko, and Alexander van Oudenaarden. Integrated genome and transcriptome sequencing of the same cell. *Nature Biotechnology*, January 2015. doi: 10.1038/nbt.3129.

[57] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv*, 2013.

[58] John Duchi, Elad Hazan, and Yoram Singer. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 12: 2121–2159, 2011.

[59] F. Eduati, L. M. Mangravite, T. Wang, H. Tang, J. C. Bare, R. Huang, T. Norman, M. Kellen, M. P. Menden, J. Yang, X. Zhan, R. Zhong, G. Xiao, M. Xia, N. Abdo, O. Kosyk, Niehs-Ncats-Unc Dream Toxicogenetics Collaboration, S. Friend, A. Dearry, A. Simeonov, R. R. Tice, I. Rusyn, F. A. Wright, G. Stolovitzky, Y. Xie, and J. Saez-Rodriguez. Prediction of human population responses to toxic compounds by a collaborative competition. *Nat Biotechnol*, 33(9), September 2015. doi: 10.1038/nbt.3299.

[60] Jesse Eickholt and Jianlin Cheng. Predicting protein residue-residue contacts using deep networks and boosting. *Bioinformatics*, 28:3066–3072, December 2012. doi: 10.1093/bioinformatics/bts598.

[61] Jesse Eickholt and Jianlin Cheng. DNdisorder: Predicting protein disorder using boosting and deep networks. *BMC Bioinformatics*, 14:88, March 2013. doi: 10.1186/1471-2105-14-88.

[62] Jason Ernst and Manolis Kellis. Large-scale imputation of epigenomic datasets for systematic annotation of diverse human tissues. *Nature Biotechnology*, 33(4):364–376, April 2015. doi: 10.1038/nbt.3157.

[63] Rasool Fakoor, Faisal Ladhak, Azade Nazi, and Manfred Huber. Using deep learning to enhance cancer diagnosis and classification. 2013.

[64] BWAC Farley and W Clark. Simulation of self-organizing systems by digital computer. *Transactions of the IRE Professional Group on Information Theory*, 4(4):76–84, 1954.

[65] Matthias Farlik, Nathan C. Sheffield, Angelo Nuzzo, Paul Datlinger, Andreas Schönegger, Johanna Klughammer, and Christoph Bock. Single-Cell DNA Methylome Sequencing and Bioinformatic Inference of Epigenomic Cell-State Dynamics. *Cell Reports*, 10 (8):1386–1397, March 2015. doi: 10.1016/j.celrep.2015.02.001.

[66] Martin E. Fernandez-Zapico, Gwen A. Lomberk, Shoichiro Tsuji, Cathrine J. DeMars, Michael R. Bardsley, Yi-Hui Lin, Luciana L. Almada, Jing-Jing Han, Debabrata Mukhopadhyay, Tamas Ordog, Navtej S. Buttar, and Raul Urrutia. A functional family-wide screening of SP/KLF proteins identifies a subset of suppressors of *KRAS* -mediated cell growth. *Biochemical Journal*, 435(2):529–537, April 2011. doi: 10.1042/BJ20100773.

[67] Alessandro Ferrari, Stefano Lombardi, and Alberto Signoroni. Bacterial colony counting by Convolutional Neural Networks. pages 7458–7461. IEEE, 2015.

[68] Gabriella Ficz, Timothy A. Hore, Fátima Santos, Heather J. Lee, Wendy Dean, Julia Arand, Felix Krueger, David Oxley, Yu-Lee Paul, Jörn Walter, Simon J. Cook, Simon Andrews, Miguel R. Branco, and Wolf Reik. FGF Signaling Inhibition in ESCs Drives Rapid Genome-wide Demethylation to the Epigenetic Ground State of Pluripotency. *Cell Stem Cell*, 13(3):351–359, September 2013. doi: 10.1016/j.stem.2013.06.004.

[69] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. *arXiv*, June 2015.

[70] Erik Gawehn, Jan A. Hiss, and Gisbert Schneider. Deep Learning in Drug Discovery. *Molecular Informatics*, 35(1):3–14, January 2016. doi: 10.1002/minf.201501008.

[71] J Raphael Gibbs, Marcel P van der Brug, Dena G Hernandez, Bryan J Traynor, Michael A Nalls, Shiao-Lin Lai, Sampath Arepalli, Allissa Dillman, Ian P Rafferty, and Juan Troncoso. Abundant quantitative trait loci exist for DNA methylation and gene expression in human brain. *PLoS Genet*, 6(5), 2010.

[72] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. pages 580–587, 2014.

[73] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. pages 249–256, 2010.

[74] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. pages 315–323, 2011.

[75] A. Graves, A.-R. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6645–6649, May 2013. doi: 10.1109/ICASSP.2013.6638947.

[76] Alex Graves. Generating Sequences With Recurrent Neural Networks. *arXiv*, August 2013.

[77] F. Grubert, J. B. Zaugg, M. Kasowski, O. Ursu, D. V. Spacek, A. R. Martin, P. Greenside, R. Srivas, D. H. Phanstiel, A. Pekowska, N. Heidari, G. Euskirchen, W. Huber, J. K. Pritchard, C. D. Bustamante, L. M. Steinmetz, A. Kundaje, and M. Snyder. Genetic Control of Chromatin States in Humans Involves Local and Distal Chromosomal Interactions. *Cell*, 162(5), August 2015. doi: 10.1016/j.cell.2015.07.048.

[78] Hongshan Guo, Ping Zhu, Xinglong Wu, Xianlong Li, Lu Wen, and Fuchou Tang. Single-cell methylome landscapes of mouse embryonic stem cells and early embryos analyzed using reduced representation bisulfite sequencing. *Genome Research*, 23(12): 2126–2135, January 2013. doi: 10.1101/gr.161679.113.

[79] Hongshan Guo, Ping Zhu, Fan Guo, Xianlong Li, Xinglong Wu, Xiaoying Fan, Lu Wen, and Fuchou Tang. Profiling DNA methylome landscapes of mammalian cells with single-cell reduced-representation bisulfite sequencing. *Nature Protocols*, 10(5):645–659, May 2015. doi: 10.1038/nprot.2015.039.

[80] Ehsan Habibi, Arie B. Brinkman, Julia Arand, Leonie I. Kroeze, Hindrik H.D. Kerstens, Filomena Matarese, Konstantin Lepikhov, Marta Gut, Isabelle Brun-Heath, Nina C. Hubner, Rosaria Benedetti, Lucia Altucci, Joop H. Jansen, Jörn Walter, Ivo G. Gut, Hendrik Marks, and Hendrik G. Stunnenberg. Whole-Genome Bisulfite Sequencing of Two Distinct Interconvertible DNA Methylomes of Mouse Embryonic Stem Cells. *Cell Stem Cell*, 13(3):360–369, September 2013. doi: 10.1016/j.stem.2013.06.002.

[81] D Hardoon, S Szedmak, and J Shawe-Taylor. Canonical Correlation Analysis: An Overview with Application to Learning Methods. *Neural Computation*, 16(12):2639–2664, December 2004. doi: 10.1162/0899766042321814.

[82] Trevor Hastie, Robert Tibshirani, Jerome Friedman, and James Franklin. The elements of statistical learning: Data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2):83–85, 2005.

[83] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *arXiv*, 2015.

[84] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. pages 1026–1034, 2015.

[85] Eric Hervouet, François M. Vallette, and Pierre-François Cartron. Dnmt3/transcription factor interactions as crucial players in targeted DNA methylation. *Epigenetics*, 4(7): 487–499, October 2009. doi: 10.4161/epi.4.7.9883.

[86] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, and Tara N Sainath. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97, 2012.

[87] Geoffrey E Hinton. A practical guide to training restricted boltzmann machines. In *Neural Networks: Tricks of the Trade*, pages 599–619. Springer, 2012. ISBN 3-642-35288-X.

[88] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[89] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780, November 1997. doi: 10.1162/neco.1997.9.8.1735.

[90] Robin Holliday and John E. Pugh. DNA modification mechanisms and gene activity during development. *COLD SPRING HARBOR MONOGRAPH SERIES*, 32:639–645, 1996.

[91] Gary C Hon, Nisha Rajagopal, Yin Shen, David F McCleary, Feng Yue, My D Dang, and Bing Ren. Epigenetic memory at embryonic enhancers identified in DNA methylation maps from adult mouse tissues. *Nature Genetics*, 45(10):1198–1206, September 2013. doi: 10.1038/ng.2746.

[92] Yu Hou, Huahu Guo, Chen Cao, Xianlong Li, Boqiang Hu, Ping Zhu, Xinglong Wu, Lu Wen, Fuchou Tang, Yanyi Huang, and Jirun Peng. Single-cell triple omics sequencing reveals genetic, epigenetic, and transcriptomic heterogeneity in hepatocellular carcinomas. *Cell Research*, 26(3):304–319, March 2016. doi: 10.1038/cr.2016.23.

[93] Guizhen Huang, Miao Yuan, Jie Zhang, Jun Li, Di Gong, Yanyan Li, Jie Zhang, Ping Lin, and Lugang Huang. IL-6 mediates differentiation disorder during spermatogenesis in obesity-associated inflammation by affecting the expression of Zfp637 through the SOCS3/STAT3 pathway. *Scientific Reports*, 6, June 2016. doi: 10.1038/srep28012.

[94] Yi-Wen Huang, Tim H.-M. Huang, and Li-Shu Wang. Profiling DNA Methylomes from Microarray to Genome-Scale Sequencing. *Technology in cancer research & treatment*, 9(2):139–147, April 2010.

[95] David H Hubel and Torsten N Wiesel. The period of susceptibility to the physiological effects of unilateral eye closure in kittens. *The Journal of physiology*, 206(2):419, 1970.

[96] DH Hubel and TN Wiesel. Shape and arrangement of columns in cat's striate cortex. *The Journal of physiology*, 165(3):559, 1963.

[97] Leah K. Hutnick, Xinhua Huang, Tao-Chuan Loo, Zhicheng Ma, and Guoping Fan. Repression of Retrotransposal Elements in Mouse Embryonic Stem Cells Is Primarily Mediated by a DNA Methylation-independent Mechanism. *Journal of Biological Chemistry*, 285(27):21082–21091, July 2010. doi: 10.1074/jbc.M110.125674.

[98] Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *Learning and Intelligent Optimization*, pages 507–523. Springer, 2011. ISBN 3-642-25565-5.

[99] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv*, February 2015.

[100] Viren Jain, Joseph F Murray, Fabian Roth, Srinivas Turaga, Valentin Zhigulin, Kevin L Briggman, Moritz N Helmstaedter, Winfried Denk, and H Sebastian Seung. Supervised learning of image restoration with convolutional networks. pages 1–8. IEEE, 2007. ISBN 1-4244-1630-2.

[101] D. A. Jaitin, E. Kenigsberg, H. Keren-Shaul, N. Elefant, F. Paul, I. Zaretsky, A. Mildner, N. Cohen, S. Jung, A. Tanay, and I. Amit. Massively Parallel Single-Cell RNA-Seq for Marker-Free Decomposition of Tissues into Cell Types. *Science*, 343(6172):776–779, February 2014. doi: 10.1126/science.1247651.

[102] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun. What is the best multi-stage architecture for object recognition? In *2009 IEEE 12th International Conference on Computer Vision*, pages 2146–2153, September 2009. doi: 10.1109/ICCV.2009.5459469.

[103] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. pages 675–678. ACM, 2014. ISBN 1-4503-3063-0.

[104] Peter A. Jones. Functions of DNA methylation: Islands, start sites, gene bodies and beyond. *Nature Reviews Genetics*, 13(7):484–492, July 2012. doi: 10.1038/nrg3230.

[105] Peter A. Jones and Gangning Liang. Rethinking how DNA methylation patterns are maintained. *Nature Reviews Genetics*, 10(11):805–811, November 2009. doi: 10.1038/nrg2651.

[106] Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. Neural Machine Translation in Linear Time. *arXiv*, October 2016.

[107] H. M. Kang, C. Ye, and E. Eskin. Accurate discovery of expression quantitative trait loci under confounding from spurious and genuine regulatory hotspots. *Genetics*, 180 (4):1909–25, December 2008. doi: 10.1534/genetics.108.094201.

[108] Irene M. Kaplow, Julia L. MacIsaac, Sarah M. Mah, Lisa M. McEwen, Michael S. Kobor, and Hunter B. Fraser. A pooling-based approach to mapping genetic variants associated with DNA methylation. *Genome Research*, April 2015. doi: 10.1101/gr.183749.114.

[109] R. Karlic, H. R. Chung, J. Lasserre, K. Vlahovicek, and M. Vingron. Histone modification levels are predictive for gene expression. *Proceedings of the National Academy of Sciences*, 107(7):2926–31, February 2010. doi: 10.1073/pnas.0909344107.

[110] Andrej Karpathy and Justin Johnson. CS231n Convolutional Neural Networks for Visual Recognition. http://cs231n.github.io/, 2016.

[111] D. B. Kell. Metabolomics, machine learning and modelling: Towards an understanding of the language of cells. *Biochem Soc Trans*, 33(Pt 3), June 2005. doi: 10.1042/BST0330520.

[112] D. R. Kelley, J. Snoek, and J. Rinn. Basset: Learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome research*, Advance online, May 2016. doi: 10.1101/gr.200535.115.

[113] Tae Hoon Kim, Ziedulla K. Abdullaev, Andrew D. Smith, Keith A. Ching, Dmitri I. Loukinov, Roland D. Green, Michael Q. Zhang, Victor V. Lobanenkov, and Bing Ren. Analysis of the Vertebrate Insulator Protein CTCF-Binding Sites in the Human Genome. *Cell*, 128(6):1231–1245, March 2007. doi: 10.1016/j.cell.2006.12.048.

[114] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv*, 2014.

[115] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv*, December 2013.

[116] Pang Wei Koh, Emma Pierson, and Anshul Kundaje. Denoising genome-wide histone ChIP-seq with convolutional neural networks. *bioRxiv*, January 2017. doi: 10.1101/052118.

[117] Aleksandra A. Kolodziejczyk, Jong Kyoung Kim, Jason C.H. Tsang, Tomislav Ilicic, Johan Henriksson, Kedar N. Natarajan, Alex C. Tuck, Xuefei Gao, Marc Bühler, Pentao Liu, John C. Marioni, and Sarah A. Teichmann. Single Cell RNA-Sequencing of Pluripotent States Unlocks Modular Transcriptional Variation. *Cell Stem Cell*, 17(4): 471–485, October 2015. doi: 10.1016/j.stem.2015.09.011.

[118] Oren Z Kraus, Lei Jimmy Ba, and Brendan Frey. Classifying and Segmenting Microscopy Images Using Convolutional Multiple Instance Learning. *arXiv*, 2015.

[119] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. pages 1097–1105, 2012.

[120] Peter W. Laird. Principles and challenges of genome-wide DNA methylation analysis. *Nature Reviews Genetics*, 11(3):191, March 2010. doi: 10.1038/nrg2732.

[121] Jack Lanchantin, Ritambhara Singh, Zeming Lin, and Yanjun Qi. Deep Motif: Visualizing Genomic Sequence Classifications. *arXiv*, May 2016.

[122] Jack Lanchantin, Ritambhara Singh, Beilun Wang, and Yanjun Qi. Deep Motif Dashboard: Visualizing and Understanding Genomic Sequences Using Deep Neural Networks. *arXiv*, 2016.

[123] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[124] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521:436–444, May 2015. doi: 10.1038/nature14539.

[125] Byunghan Lee, Taehoon Lee, Byunggook Na, and Sungroh Yoon. DNA-Level Splice Junction Prediction using Deep Recurrent Neural Networks. *arXiv*, 2015.

[126] Catherine S. Lee, Newman J. Sund, Rüdiger Behr, Pedro L. Herrera, and Klaus H. Kaestner. Foxa2 is required for the differentiation of pancreatic alpha-cells. *Developmental Biology*, 278(2):484–495, February 2005. doi: 10.1016/j.ydbio.2004.10.012.

[127] Heather J. Lee, Timothy A. Hore, and Wolf Reik. Reprogramming the Methylome: Erasing Memory and Creating Diversity. *Cell Stem Cell*, 14(6):710–719, June 2014. doi: 10.1016/j.stem.2014.05.008.

[128] Michael K. K. Leung, Hui Yuan Xiong, Leo J. Lee, and Brendan J. Frey. Deep learning of the tissue-regulated splicing code. *Bioinformatics*, 30, June 2014. doi: 10.1093/bioinformatics/btu277.

[129] J. Li, T. Ching, S. Huang, and L. X. Garmire. Using epigenomics data to predict gene expression in lung cancer. *BMC Bioinformatics*, 16 Suppl 5, 2015. doi: 10.1186/1471-2105-16-S5-S10.

[130] Stan Z Li. *Markov Random Field Modeling in Image Analysis*. Springer Science & Business Media, 2009. ISBN 1-84800-279-3.

[131] Yingrui Li, Jingde Zhu, Geng Tian, Ning Li, Qibin Li, Mingzhi Ye, Hancheng Zheng, Jian Yu, Honglong Wu, Jihua Sun, Hongyu Zhang, Quan Chen, Ruibang Luo, Minfeng Chen, Yinghua He, Xin Jin, Qinghui Zhang, Chang Yu, Guangyu Zhou, Jinfeng Sun, Yebo Huang, Huisong Zheng, Hongzhi Cao, Xiaoyu Zhou, Shicheng Guo, Xueda Hu, Xin Li, Karsten Kristiansen, Lars Bolund, Jiujin Xu, Wen Wang, Huanming Yang, Jian Wang, Ruiqiang Li, Stephan Beck, Jun Wang, and Xiuqing Zhang. The DNA Methylome of Human Peripheral Blood Mononuclear Cells. *PLoS Biology*, 8(11), November 2010. doi: 10.1371/journal.pbio.1000533.

[132] Zhanchao Li, Lili Chen, Yanhua Lai, Zong Dai, and Xiaoyong Zou. The prediction of methylation states in human DNA sequences based on hexanucleotide composition and feature selection. *Analytical Methods*, 6(6):1897, 2014. doi: 10.1039/c3ay41962b.

[133] Zhen Li and Yizhou Yu. Protein Secondary Structure Prediction Using Cascaded Convolutional and Recurrent Neural Networks. *arXiv*, April 2016.

[134] M. W. Libbrecht and W. S. Noble. Machine learning applications in genetics and genomics. *Nat Rev Genet*, 16(6):321–32, June 2015. doi: 10.1038/nrg3920.

[135] Zachary C. Lipton. A Critical Review of Recurrent Neural Networks for Sequence Learning. *arXiv*, May 2015.

[136] Zi Liu, Xuan Xiao, Wang-Ren Qiu, and Kuo-Chen Chou. iDNA-Methyl: Identifying DNA methylation sites via pseudo trinucleotide composition. *Analytical Biochemistry*, 474:69–77, April 2015. doi: 10.1016/j.ab.2014.12.009.

[137] Lingyi Lu. Predicting DNA methylation status using word composition. *Journal of Biomedical Science and Engineering*, 03(07):672–676, 2010. doi: 10.4236/jbise.2010.37091.

[138] P.-L. Luu, H. R. Scholer, and M. J. Arauzo-Bravo. Disclosing the crosstalk among DNA methylation, transcription factors, and histone marks in human pluripotent cells through discovery of DNA methylation motifs. *Genome Research*, 23(12):2013–2029, December 2013. doi: 10.1101/gr.155960.113.

[139] James Lyons, Abdollah Dehzangi, Rhys Heffernan, Alok Sharma, Kuldip Paliwal, Abdul Sattar, Yaoqi Zhou, and Yuedong Yang. Predicting backbone C-alpha angles and dihedrals from protein sequences by stacked sparse auto-encoder deep neural network. *Journal of Computational Chemistry*, 35(28):2040–2046, October 2014. doi: 10.1002/jcc.23718.

[140] Iain C. Macaulay and Thierry Voet. Single Cell Genomics: Advances and Future Perspectives. *PLoS Genetics*, 10(1), January 2014. doi: 10.1371/journal.pgen.1004126.

[141] Iain C. Macaulay, Wilfried Haerty, Parveen Kumar, Yang I. Li, Tim Xiaoming Hu, Mabel J. Teng, Mubeen Goolam, Nathalie Saurat, Paul Coupland, Lesley M. Shirley, Miriam Smith, Niels Van der Aa, Ruby Banerjee, Peter D. Ellis, Michael A. Quail, Harold P. Swerdlow, Magdalena Zernicka-Goetz, Frederick J. Livesey, Chris P. Ponting, and

Thierry Voet. G&T-seq: Parallel sequencing of single-cell genomes and transcriptomes. *Nature Methods*, 12(6):519–522, June 2015. doi: 10.1038/nmeth.3370.

[142] David J. C. MacKay. Bayesian Interpolation. *Neural Computation*, 4(3):415–447, May 1992. doi: 10.1162/neco.1992.4.3.415.

[143] Aravindh Mahendran and Andrea Vedaldi. Visualizing Deep Convolutional Neural Networks Using Natural Pre-Images. *arXiv*, December 2015.

[144] Andigoni Malousi, Ioanna Chouvarda, Sofia Kouidou, and Nicos Maglaveras. A Predictive Model for Genomic Methylation Targets in Humans. *Journal of Bioinformatics*, January 2014.

[145] Richard Marais, Judy Wynne, and Richard Treisman. The SRF accessory protein Elk-1 contains a growth factor-regulated transcriptional activation domain. *Cell*, 73(2): 381–393, April 1993. doi: 10.1016/0092-8674(93)90237-K.

[146] K. Märtens, J. Hallin, J. Warringer, G. Liti, and L. Parts. Predicting quantitative traits from genome and phenome with near perfect accuracy. *Nature communications*, 7: 11512, 2016. doi: 10.1038/ncomms11512.

[147] B.W. Matthews. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure*, 405(2): 442–451, October 1975. doi: 10.1016/0005-2795(75)90109-9.

[148] Alika K. Maunakea, Raman P. Nagarajan, Mikhail Bilenky, Tracy J. Ballinger, Cletus D'Souza, Shaun D. Fouse, Brett E. Johnson, Chibo Hong, Cydney Nielsen, Yongjun Zhao, Gustavo Turecki, Allen Delaney, Richard Varhol, Nina Thiessen, Ksenya Shchors, Vivi M. Heine, David H. Rowitch, Xiaoyun Xing, Chris Fiore, Maximiliaan Schillebeeckx, Steven J. M. Jones, David Haussler, Marco A. Marra, Martin Hirst, Ting Wang, and Joseph F. Costello. Conserved role of intragenic DNA methylation in regulating alternative promoters. *Nature*, 466(7303):253–257, July 2010. doi: 10.1038/nature09165.

[149] Wolfgang Mayer, Alain Niveleau, Jörn Walter, Reinald Fundele, and Thomas Haaf. Embryogenesis: Demethylation of the zygotic paternal genome. *Nature*, 403(6769): 501–502, February 2000. doi: 10.1038/35000656.

[150] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[151] Alexander Meissner, Tarjei S. Mikkelsen, Hongcang Gu, Marius Wernig, Jacob Hanna, Andrey Sivachenko, Xiaolan Zhang, Bradley E. Bernstein, Chad Nusbaum, David B. Jaffe, Andreas Gnirke, Rudolf Jaenisch, and Eric S. Lander. Genome-scale DNA methylation maps of pluripotent and differentiated cells. *Nature*, 454(7205):766–770, August 2008. doi: 10.1038/nature07107.

[152] M. P. Menden, F. Iorio, M. Garnett, U. McDermott, C. H. Benes, P. J. Ballester, and J. Saez-Rodriguez. Machine learning prediction of cancer cell sensitivity to drugs based on genomic and chemical properties. *PLoS One*, 8(4), 2013. doi: 10.1371/journal.pone.0061318.

[153] Eric M. Mendenhall, Richard P. Koche, Thanh Truong, Vicky W. Zhou, Biju Issac, Andrew S. Chi, Manching Ku, and Bradley E. Bernstein. GC-Rich Sequence Elements Recruit PRC2 in Mammalian ES Cells. *PLoS Genetics*, 6(12), December 2010. doi: 10.1371/journal.pgen.1001244.

[154] Ryszard S Michalski, Jaime G Carbonell, and Tom M Mitchell. *Machine Learning: An Artificial Intelligence Approach*. Springer Science & Business Media, 2013. ISBN 3-662-12405-X.

[155] Fumihito Miura, Yusuke Enomoto, Ryo Dairiki, and Takashi Ito. Amplification-free whole-genome bisulfite sequencing by post-bisulfite adaptor tagging. *Nucleic Acids Research*, 40(17), January 2012. doi: 10.1093/nar/gks454.

[156] S. B. Montgomery, M. Sammeth, M. Gutierrez-Arcelus, R. P. Lach, C. Ingle, J. Nisbett, R. Guigo, and E. T. Dermitzakis. Transcriptome genetics using second generation sequencing in a Caucasian population. *Nature*, 464(7289), April 2010. doi: 10.1038/nature08903.

[157] Lisa D Moore, Thuc Le, and Guoping Fan. DNA Methylation and Its Basic Function. *Neuropsychopharmacology*, 38(1):23–38, January 2013. doi: 10.1038/npp.2012.112.

[158] Kevin P Murphy. *Machine Learning: A Probabilistic Perspective*. MIT press, 2012. ISBN 0-262-01802-0.

[159] Radford M Neal. *Bayesian Learning for Neural Networks*, volume 118. Springer Science & Business Media, 2012.

[160] Yurii Nesterov. A method of solving a convex programming problem with convergence rate O (1/k2). volume 27, pages 372–376, 1983. 2.

[161] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*, volume 87. Springer Science & Business Media, 2013. ISBN 1-4419-8853-X.

[162] D. E. Newburger and M. L. Bulyk. UniPROBE: An online database of protein binding microarray data on protein-DNA interactions. *Nucleic Acids Research*, 37(Database), January 2009. doi: 10.1093/nar/gkn660.

[163] Feng Ning, Damien Delhomme, Yann LeCun, Fabio Piano, Léon Bottou, and Paolo Emilio Barbano. Toward automatic phenotyping of developing embryos from videos. *Image Processing, IEEE Transactions on*, 14(9):1360–1371, 2005.

[164] Gaurav Pandey and Ambedkar Dukkipati. Variational methods for Conditional Multimodal Deep Learning. *arXiv*, March 2016.

[165] Bernadett Papp and Kathrin Plath. Pluripotency re-centered around Esrrb: Pluripotency re-centered around Esrrb. *The EMBO Journal*, 31(22):4255–4257, November 2012. doi: 10.1038/emboj.2012.285.

[166] T. Pärnamaa and L. Parts. Accurate classification of protein subcellular localization from high throughput microscopy images using deep learning. *bioRxiv*, April 2016. doi: 10.1101/050757.

[167] L. Parts, O. Stegle, J. Winn, and R. Durbin. Joint genetic analysis of gene expression data with inferred cellular phenotypes. *PLoS Genet*, 7(1):e1001276, 2011. doi: 10.1371/journal.pgen.1001276.

[168] L. Parts, Y. C. Liu, M. M. Tekkedil, L. M. Steinmetz, A. A. Caudy, A. G. Fraser, C. Boone, B. J. Andrews, and A. P. Rosebrock. Heritability and genetic basis of protein level variation in an outbred population. *Genome Res*, 24(8), August 2014. doi: 10.1101/gr.170506.113.

[169] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. *ICML*, 28:1310–1318, 2013.

[170] Len A. Pennacchio, Wendy Bickmore, Ann Dean, Marcelo A. Nobrega, and Gill Bejerano. Enhancers: Five essential questions. *Nature Reviews Genetics*, 14(4):288–295, March 2013. doi: 10.1038/nrg3458.

[171] J. K. Pickrell, J. C. Marioni, A. A. Pai, J. F. Degner, B. E. Engelhardt, E. Nkadori, J. B. Veyrieras, M. Stephens, Y. Gilad, and J. K. Pritchard. Understanding mechanisms underlying human gene expression variation with RNA sequencing. *Nature*, 464(7289), April 2010. doi: 10.1038/nature08872.

[172] Nongluk Plongthongkum, Dinh H. Diep, and Kun Zhang. Advances in the profiling of DNA modifications: Cytosine methylation and beyond. *Nature Reviews Genetics*, 15 (10):647–661, October 2014. doi: 10.1038/nrg3772.

[173] David Martin Powers. Evaluation: From Precision, Recall and F-measure to ROC, Informedness, Markedness and Correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, December 2011.

[174] Daniel Quang and Xiaohui Xie. DanQ: A hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic Acids Research*, 44 (11), June 2016. doi: 10.1093/nar/gkw226.

[175] Simon Quenneville, Gaetano Verde, Andrea Corsinotti, Adamandia Kapopoulou, Johan Jakobsson, Sandra Offner, Ilaria Baglivo, Paolo V. Pedone, Giovanna Grimaldi, Andrea Riccio, and Didier Trono. In embryonic stem cells, ZFP57/KAP1 recognize a methylated hexanucleotide to affect chromatin and DNA methylation of imprinting control regions. *Molecular Cell*, 44(3):361–372, November 2011. doi: 10.1016/j.molcel.2011.08.032.

[176] Janarthanan Rajendran, Mitesh M. Khapra, Sarath Chandar, and Balaraman Ravindran. Bridge Correlational Neural Networks for Multilingual Multimodal Representation Learning. *arXiv*, October 2015.

[177] B. Rakitsch and O. Stegle. Modelling local gene networks increases power to detect trans-acting genetic effects on gene expression. *Genome Biol*, 17(1):33, 2016. doi: 10.1186/s13059-016-0895-2.

[178] B. H. Ramsahoye, D. Biniszkiewicz, F. Lyko, V. Clark, A. P. Bird, and R. Jaenisch. Non-CpG methylation is prevalent in embryonic stem cells and may be mediated by DNA methyltransferase 3a. *Proceedings of the National Academy of Sciences*, 97(10): 5237–5242, May 2000. doi: 10.1073/pnas.97.10.5237.

[179] Keith D. Robertson. DNA methylation and human disease. *Nature Reviews Genetics*, 6 (8):597–610, August 2005. doi: 10.1038/nrg1655.

[180] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015*, pages 234–241. Springer, 2015. ISBN 3-319-24573-2.

[181] Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[182] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1, 1988.

[183] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, and Michael Bernstein. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115 (3):211–252, 2015.

[184] Ruslan Salakhutdinov and Geoffrey Hinton. An efficient learning procedure for deep Boltzmann machines. *Neural computation*, 24(8):1967–2006, 2012.

[185] Ruslan Salakhutdinov and Hugo Larochelle. Efficient learning of deep Boltzmann machines. pages 693–700, 2010.

[186] Juergen Schmidhuber. Deep Learning in Neural Networks: An Overview. *Neural Networks*, 61:85–117, January 2015. doi: 10.1016/j.neunet.2014.09.003.

[187] Omer Schwartzman and Amos Tanay. Single-cell epigenomics: Techniques and emerging applications. *Nature Reviews Genetics*, 16(12):716–726, December 2015. doi: 10.1038/nrg3980.

[188] Iulian V. Serban, Alexander G. Ororbia II, Joelle Pineau, and Aaron Courville. Multi-modal Variational Encoder-Decoders. *arXiv*, December 2016.

[189] Rita Shaknovich, Maria E. Figueroa, and Ari Melnick. HELP (HpaII Tiny Fragment Enrichment by Ligation-Mediated PCR) Assay for DNA Methylation Profiling of Primary Normal and Malignant B Lymphocytes. In Sridar V. Chittur, editor, *Microarray Methods for Drug Discovery*, volume 632, pages 191–201. Humana Press, Totowa, NJ, 2010. ISBN 978-1-60761-662-7 978-1-60761-663-4. doi: 10.1007/978-1-60761-663-4_12.

[190] Shikhar Sharma, Ryan Kiros, and Ruslan Salakhutdinov. Action Recognition using Visual Attention. November 2015.

[191] Kenjiro Shirane, Hidehiro Toh, Hisato Kobayashi, Fumihito Miura, Hatsune Chiba, Takashi Ito, Tomohiro Kono, and Hiroyuki Sasaki. Mouse Oocyte Methylomes at Base Resolution Reveal Genome-Wide Accumulation of Non-CpG Methylation and Role of DNA Methyltransferases. *PLoS Genetics*, 9(4), April 2013. doi: 10.1371/journal.pgen.1003439.

[192] Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not Just a Black Box: Learning Important Features Through Propagating Activation Differences. *arXiv*, May 2016.

[193] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv*, 2014.

[194] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv*, December 2013.

[195] Ajit P. Singh and Geoffrey J. Gordon. Relational Learning via Collective Matrix Factorization. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 650–658, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-193-4. doi: 10.1145/1401890.1401969.

[196] Sébastien A. Smallwood, Heather J. Lee, Christof Angermueller, Felix Krueger, Heba Saadeh, Julian Peat, Simon R. Andrews, Oliver Stegle, Wolf Reik, and Gavin Kelsey. Single-cell genome-wide bisulfite sequencing for assessing epigenetic heterogeneity. *Nature Methods*, 11(8):817–820, August 2014. doi: 10.1038/nmeth.3035.

[197] Zachary D. Smith, Hongcang Gu, Christoph Bock, Andreas Gnirke, and Alexander Meissner. High-throughput bisulfite sequencing in mammalian genomes. *Methods*, 48 (3):226–232, July 2009. doi: 10.1016/j.ymeth.2009.05.003.

[198] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. pages 2951–2959, 2012.

[199] Søren Kaae Sønderby and Ole Winther. Protein Secondary Structure Prediction with Long Short Term Memory Networks. *arXiv*, December 2014.

[200] Matt Spencer, Jesse Eickholt, and Jianlin Cheng. A deep learning network approach to ab initio protein secondary structure prediction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 12(1):103–112, 2015.

[201] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for Simplicity: The All Convolutional Net. *arXiv*, December 2014.

[202] Nitish Srivastava and Ruslan R. Salakhutdinov. Multimodal learning with deep boltzmann machines. In *Advances in Neural Information Processing Systems*, pages 2222–2230, 2012.

[203] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15:1929–1958, 2014.

[204] Michael B. Stadler, Rabih Murr, Lukas Burger, Robert Ivanek, Florian Lienert, Anne Schöler, Christiane Wirbelauer, Edward J. Oakeley, Dimos Gaidatzis, Vijay K. Tiwari, and Dirk Schübeler. DNA-binding factors shape the mouse methylome at distal regulatory regions. *Nature*, December 2011. doi: 10.1038/nature10716.

[205] O. Stegle, L. Parts, R. Durbin, and J. Winn. A Bayesian framework to account for complex non-genetic factors in gene expression levels greatly increases power in eQTL studies. *PLoS Comput Biol*, 6(5), May 2010. doi: 10.1371/journal.pcbi.1000770.

[206] Charles Stein and Abraham Wald. Sequential Confidence Intervals for the Mean of a Normal Distribution with Known Variance. *The Annals of Mathematical Statistics*, 18 (3):427–433, September 1947. doi: 10.1214/aoms/1177730389.

[207] Michael Stevens, Jeffrey B. Cheng, Daofeng Li, Mingchao Xie, Chibo Hong, Cécile L. Maire, Keith L. Ligon, Martin Hirst, Marco A. Marra, Joseph F. Costello, and Ting Wang. Estimating absolute methylation levels at single-CpG resolution from methylation enrichment and restriction enzyme sequencing methods. *Genome Research*, 23(9): 1541–1553, January 2013. doi: 10.1101/gr.152231.112.

[208] Gary D Stormo, Thomas D Schneider, Larry Gold, and Andrzej Ehrenfeucht. Use of the 'Perceptron'algorithm to distinguish translational initiation sites in E. coli. *Nucleic Acids Research*, 10(9):2997–3011, 1982.

[209] Stefan H. Stricker, Anna Köferle, and Stephan Beck. From profiles to function in epigenomics. *Nature Reviews Genetics*, November 2016. doi: 10.1038/nrg.2016.138.

[210] L. Sumoy, L. Carim, M. Escarceller, M. Nadal, M. Gratacòs, M.A. Pujana, X. Estivill, and B. Peral. HMG20A and HMG20B map to human chromosomes 15q24 and 19p13.3 and constitute a distinct class of HMG-box genes with ubiquitous expression. *Cytogenetic and Genome Research*, 88(1-2):62–67, March 2000. doi: 10.1159/000015486.

[211] Ilya Sutskever. *Training Recurrent Neural Networks*. PhD thesis, University of Toronto, 2013.

[212] Ilya Sutskever, Geoffrey E. Hinton, and Graham W. Taylor. The Recurrent Temporal Restricted Boltzmann Machine. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 1601–1608. Curran Associates, Inc., 2009.

[213] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. pages 3104–3112, 2014.

[214] Masahiro Suzuki, Kotaro Nakayama, and Yutaka Matsuo. Joint Multimodal Learning with Deep Generative Models. *arXiv*, November 2016.

[215] A. L. Swan, A. Mobasheri, D. Allaway, S. Liddell, and J. Bacardit. Application of machine learning to proteomics data: Classification and biomarker identification in postgenomics biology. *OMICS*, 17(12):595–610, December 2013. doi: 10.1089/omi.2013.0017.

[216] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. *arXiv*, 2015.

[217] Bruce Thompson. Canonical Correlation Analysis. In Brian S. Everitt and David C. Howell, editors, *Encyclopedia of Statistics in Behavioral Science*. John Wiley & Sons, Ltd, Chichester, UK, October 2005. ISBN 978-0-470-86080-9 978-0-470-01319-9. doi: 10.1002/0470013192.bsa068.

[218] John P. Thomson, Peter J. Skene, Jim Selfridge, Thomas Clouaire, Jacky Guy, Shaun Webb, Alastair R. W. Kerr, Aimée Deaton, Rob Andrews, Keith D. James, Daniel J. Turner, Robert Illingworth, and Adrian Bird. CpG islands influence chromatin structure via the CpG-binding protein Cfp1. *Nature*, 464(7291):1082–1086, April 2010. doi: 10.1038/nature08924.

[219] Shih-Yin Tsai, Rene Opavsky, Nidhi Sharma, Lizhao Wu, Shan Naidu, Eric Nolan, Enrique Feria-Arias, Cynthia Timmers, Jana Opavska, Alain de Bruin, Jean-Leon Chong, Prashant Trikha, Soledad A. Fernandez, Paul Stromberg, Thomas J. Rosol, and Gustavo Leone. Mouse development with a single E2F activator. *Nature*, 454(7208):1137–1141, August 2008. doi: 10.1038/nature07066.

[220] Mark A Urich, Joseph R Nery, Ryan Lister, Robert J Schmitz, and Joseph R Ecker. MethylC-seq library preparation for base-resolution whole-genome bisulfite sequencing. *Nature Protocols*, 10(3):475–483, February 2015. doi: 10.1038/nprot.2014.114.

[221] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. WaveNet: A Generative Model for Raw Audio. *arXiv*, September 2016.

[222] Aaron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel Recurrent Neural Networks. *arXiv*, January 2016.

[223] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *The Journal of Machine Learning Research*, 11:3371–3408, 2010.

[224] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. pages 3156–3164, 2015.

[225] Colum P. Walsh and Timothy H. Bestor. Cytosine methylation and mammalian development. *Genes & Development*, 13(1):26–34, January 1999.

[226] H. Wan, S. Dingle, Y. Xu, V. Besnard, K. H. Kaestner, S.-L. Ang, S. Wert, M. T. Stahlman, and J. A. Whitsett. Compensatory Roles of Foxa1 and Foxa2 during Lung Morphogenesis. *Journal of Biological Chemistry*, 280(14):13809–13816, April 2005. doi: 10.1074/jbc.M414122200.

[227] Kun Wang, Kan Cao, and Sridhar Hannenhalli. Chromatin and genomic determinants of alternative splicing. pages 345–354. ACM, 2015. ISBN 1-4503-3853-4.

[228] S. M. Waszak, O. Delaneau, A. R. Gschwind, H. Kilpinen, S. K. Raghav, R. M. Witwicki, A. Orioli, M. Wiederkehr, N. I. Panousis, A. Yurovsky, L. Romano-Palumbo, A. Planchon, D. Bielser, I. Padioleau, G. Udin, S. Thurnheer, D. Hacker, N. Hernandez, A. Reymond, B. Deplancke, and E. T. Dermitzakis. Population Variation and Genetic Control of Modular Chromatin Architecture in Humans. *Cell*, 162(5), August 2015. doi: 10.1016/j.cell.2015.08.001.

[229] Matthew T. Weirauch, Ally Yang, Mihai Albu, Atina G. Cote, Alejandro Montenegro-Montero, Philipp Drewe, Hamed S. Najafabadi, Samuel A. Lambert, Ishminder Mann, Kate Cook, Hong Zheng, Alejandra Goity, Harm van Bakel, Jean-Claude Lozano, Mary Galli, Mathew G. Lewsey, Eryong Huang, Tuhin Mukherjee, Xiaoting Chen, John S. Reece-Hoyes, Sridhar Govindarajan, Gad Shaulsky, Albertha J.M. Walhout, François-Yves Bouget, Gunnar Ratsch, Luis F. Larrondo, Joseph R. Ecker, and Timothy R. Hughes. Determination and Inference of Eukaryotic Transcription Factor Sequence Specificity. *Cell*, 158(6):1431–1443, September 2014. doi: 10.1016/j.cell.2014.08.009.

[230] Yu-I Weng, Tim H.-M. Huang, and Pearlly S. Yan. Methylated DNA Immunoprecipitation and Microarray-Based Analysis: Detection of DNA Methylation in Breast Cancer Cell Lines. In Ok-Kyong Park-Sarge and Thomas E. Curry, editors, *Molecular Endocrinology*, volume 590, pages 165–176. Humana Press, Totowa, NJ, 2009. ISBN 978-1-60327-377-0 978-1-60327-378-7. doi: 10.1007/978-1-60327-378-7_10.

[231] John W. Whitaker, Zhao Chen, and Wei Wang. Predicting the human epigenome from DNA motifs. *Nature Methods*, 12(3):265–272, March 2015. doi: 10.1038/nmeth.3065.

[232] Warren A. Whyte, David A. Orlando, Denes Hnisz, Brian J. Abraham, Charles Y. Lin, Michael H. Kagey, Peter B. Rahl, Tong Ihn Lee, and Richard A. Young. Master Transcription Factors and Mediator Establish Super-Enhancers at Key Cell Identity Genes. *Cell*, 153(2):307–319, April 2013. doi: 10.1016/j.cell.2013.03.035.

[233] Yuanpu Xie, Fuyong Xing, Xiangfei Kong, Hai Su, and Lin Yang. Beyond Classification: Structured Regression for Robust Cell Detection Using Convolutional Neural Network. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015*, pages 358–365. Springer, 2015. ISBN 3-319-24573-2.

[234] Caiming Xiong, Stephen Merity, and Richard Socher. Dynamic Memory Networks for Visual and Textual Question Answering. *arXiv*, March 2016.

[235] Hui Y. Xiong, Babak Alipanahi, Leo J. Lee, Hannes Bretschneider, Daniele Merico, Ryan K. C. Yuen, Yimin Hua, Serge Gueroussov, Hamed S. Najafabadi, Timothy R. Hughes, Quaid Morris, Yoseph Barash, Adrian R. Krainer, Nebojsa Jojic, Stephen W. Scherer, Benjamin J. Blencowe, and Brendan J. Frey. The human splicing code reveals new insights into the genetic determinants of disease. *Science*, 347:1254806, September 2015. doi: 10.1126/science.1254806.

[236] Huijuan Xu and Kate Saenko. Ask, Attend and Answer: Exploring Question-Guided Spatial Attention for Visual Question Answering. *arXiv*, November 2015.

[237] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. *arXiv*, February 2015.

[238] Yan Xu, Tao Mo, Qiwei Feng, Peilin Zhong, Maode Lai, and Eric I Chang. Deep learning of feature representation with multiple instance learning for medical image analysis. pages 1626–1630. IEEE, 2014.

[239] Wai-Shin Yong, Fei-Man Hsu, and Pao-Yang Chen. Profiling genome-wide DNA methylation. *Epigenetics & Chromatin*, 9:26, 2016. doi: 10.1186/s13072-016-0075-3.

[240] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3320–3328. Curran Associates, Inc., 2014.

[241] Matthew D. Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks. In *Computer Vision – ECCV 2014*, pages 818–833. Springer, Cham, September 2014. doi: 10.1007/978-3-319-10590-1_53.

[242] F. Zhang, J. H. Pomerantz, G. Sen, A. T. Palermo, and H. M. Blau. Active tissue-specific DNA demethylation conferred by somatic cell nuclei in stable heterokaryons. *Proceedings of the National Academy of Sciences*, 104(11):4395–4400, March 2007. doi: 10.1073/pnas.0700181104.

[243] Weiwei Zhang, Tim D. Spector, Panos Deloukas, Jordana T. Bell, and Barbara E. Engelhardt. Predicting genome-wide DNA methylation using methylation marks, genomic position, and DNA regulatory elements. *Genome Biology*, 16(1):14, January 2015. doi: 10.1186/s13059-015-0581-9.

[244] Wenlu Zhang, Rongjian Li, Tao Zeng, Qian Sun, Sudhir Kumar, Jieping Ye, and Shuiwang Ji. Deep model based transfer and multi-task learning for biological image analysis. pages 1475–1484. ACM, 2015. ISBN 1-4503-3664-7.

[245] Hao Zheng, Shi-Wen Jiang, and Hongwei Wu. Enhancement on the predictive power of the prediction model for human genomic DNA methylation. In *International Conference on Bioinformatics and Computational Biology (BIOCOMP)*, 2011.

[246] Hao Zheng, Hongwei Wu, Jinping Li, and Shi-Wen Jiang. CpGIMethPred: Computational model for predicting methylation status of CpG islands in human genome. *BMC Medical Genomics*, 6, January 2013. doi: 10.1186/1755-8794-6-S1-S13.

[247] J. Zhou and O. G. Troyanskaya. Predicting effects of noncoding variants with deep learning-based sequence model. *Nat Methods*, 12(10):931–4, October 2015. doi: 10.1038/nmeth.3547.

[248] Xuan Zhou, Zhanchao Li, Zong Dai, and Xiaoyong Zou. Prediction of methylation CpGs and their methylation degrees in human DNA sequences. *Computers in Biology and Medicine*, 42(4):408–413, April 2012. doi: 10.1016/j.compbiomed.2011.12.008.

[249] Michael J. Ziller, Fabian Müller, Jing Liao, Yingying Zhang, Hongcang Gu, Christoph Bock, Patrick Boyle, Charles B. Epstein, Bradley E. Bernstein, Thomas Lengauer, Andreas Gnirke, and Alexander Meissner. Genomic Distribution and Inter-Sample Variation of Non-CpG Methylation across Human Cell Types. *PLoS Genetics*, 7(12), December 2011. doi: 10.1371/journal.pgen.1002389.

[250] Michael J. Ziller, Hongcang Gu, Fabian Müller, Julie Donaghey, Linus T.-Y. Tsai, Oliver Kohlbacher, Philip L. De Jager, Evan D. Rosen, David A. Bennett, Bradley E. Bernstein, Andreas Gnirke, and Alexander Meissner. Charting a dynamic DNA methylation landscape of the human genome. *Nature*, 500(7463):477–481, August 2013. doi: 10.1038/nature12433.