

Epigenetics of complex traits and diseases

Charles E. Breeze

UCL Cancer Institute

University College London

This thesis is submitted for the degree of Doctor of Philosophy

DECLARATION

I, Charles Edmund Breeze, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Charles E. Breeze

29 March 2017

PUBLICATIONS

The following publications have resulted from the work presented in this thesis and conducted through collaboration on other projects. The corresponding abstracts and title pages can be found in the *Appendices*.

1. **Breeze, C.E.**, Paul, D.S., van Dongen, J., Butcher, L.M., Ambrose, J.C., Barrett, J.E., Lowe, R., Rakan, V.K., Iotchkova, V., Frontini, M., et al. (2016). eFORGE: A Tool for Identifying Cell Type-Specific Signal in Epigenomic Data. *Cell Rep.* 17, 2137–215.
2. Lewis, J.* , **Breeze, C.E.***, Charlesworth, J., Maclaren, O.J., and Cooper, J. (2016). Where next for the reproducibility agenda in computational biology? *BMC Syst. Biol.* 10, 52
3. Pan, S., Lai, H., Shen, Y., **Breeze, C.**, Beck, S., Hong, T., Wang, C., and Teschendorff, A.E. (2017). DNA methylome analysis reveals distinct epigenetic patterns of ascending aortic dissection and bicuspid aortic valve. *Cardiovasc. Res.* 113, 692–704.
4. Stunnenberg, H.G., Abrignani, S., Adams, D., Almeida, M. de, Altucci, L., Amin, V., Amit, I., Antonarakis, S.E., Aparicio, S., Arima, T., Arrigoni, L., Arts, R., Asnafi, V., Badosa, M.E., Bae, J.-B., Bassler, K., Beck, S., Berkman, B., Bernstein, B.E., Bilenky, M., Bird, A., Bock, C., Boehm, B., Bourque, G., **Breeze, C.E.**, et al. (2016). The International Human Epigenome Consortium: A Blueprint for Scientific Collaboration and Discovery. *Cell* 167, 1145–1149.
5. Teschendorff, A.E., **Breeze, C.E.**, Zheng, S.C., and Beck, S. (2017). A comparison of reference-based algorithms for correcting cell-type heterogeneity in Epigenome-Wide Association Studies. *BMC Bioinformatics* 18, 105.
6. Bartlett, T.E., Chindera, K., McDermott, J., **Breeze, C.E.**, Cooke, W.R., Jones, A., Reisel, D., Karegodar, S.T., Arora, R., Beck, S., et al. (2016). Epigenetic reprogramming of fallopian tube fimbriae in BRCA mutation carriers defines early ovarian cancer evolution. *Nat. Commun.* 7, 11620.
7. van Dongen, J., Nivard, M.G., Willemsen, G., Hottenga, J.-J., Helmer, Q., Dolan, C.V., Ehli, E.A., Davies, G.E., van Ijzerson, M., **Breeze, C.E.**, et al. (2016). Genetic and environmental influences interact with age and sex in shaping the human methylome. *Nat. Commun.* 7, 11115.
8. Chen, Y., **Breeze, C.E.**, Zhen, S., Beck, S., and Teschendorff, A.E. (2016). Tissue-independent and tissue-specific patterns of DNA methylation alteration in cancer. *Epigenetics Chromatin* 9, 10.

ABSTRACT

Thousands of genetic and epigenetic variants have been identified for many common diseases including cancer through genome-wide association studies (GWAS) and epigenome-wide association studies (EWAS). To advance the complex interpretation of both GWAS and EWAS results, I developed new software tools (FORGE2 and eFORGE) for the analysis and interpretation of GWAS and EWAS data, respectively. Both tools determine the cell type-specific regulatory component of a set of target regions (either GWAS-identified genetic variants or EWAS-identified differentially methylated positions). This is achieved by detecting enrichment of overlap with histone mark peaks or DNase I hypersensitive sites across hundreds of tissues, primary cell types, and cell lines from the ENCODE, Roadmap Epigenomics, and BLUEPRINT projects. Application of both tools to publicly available datasets identified novel disease-relevant cell types for many common diseases, a stem cell-like signature in cancer EWAS, and also demonstrated the ability to detect cell-composition effects for EWAS performed on heterogeneous tissues. To complement these bioinformatics efforts and validate selected variants predicted by FORGE2, eFORGE and additional analyses, I performed conformation capture using 4C-seq to fine-map the 3D context of the genomic regions involved, uncovering novel interactions for autoimmunity-associated variants and IKZF3.

ACKNOWLEDGEMENTS

The majority of the results presented in chapter 3, relating to the eFORGE software tool, have been published in Cell Reports¹⁴⁶. The corresponding title page, journal cover and abstract can be found in the *Appendices*. I developed eFORGE during a secondment at the European Bioinformatics Institute (EBI), in collaboration with Dr Ian Dunham, Dr Valentina Iotchkova and Prof. Ewan Birney, and also in collaboration with QMUL researcher Dr Robert Lowe and UCL researchers Dr James Barrett, Dr Javier Herrero, Dr Dirk Paul, Prof. Andrew Teschendorff and Prof. Stephan Beck. Notably, the web interface for eFORGE was developed by Dr Javier Herrero at the Bill Lyons Informatics Centre at the UCL Cancer Institute.

I developed the FORGE2 software tool with advice from Dr Ian Dunham (EBI) and incorporating some eFORGE code improvements introduced by Dr Javier Herrero (UCL Cancer Institute).

I performed the 4C-seq experiments on disease-associated loci following advice from UCL researchers including Dr Dimitra Georgopolou, Dr José Ariza, Dr Suzana Hadjur, Dr Anna Köferle, Dr Andy Feber, Dr Miljana Tanic and Prof. Stephan Beck. In addition, Dr Pawan Dhami and Alex McLatchie aided with high-throughput sequencing. Dr Chris Barrington provided guidelines for 4C-seq computational analysis. Prof. Javier Novo (University of Navarra) aided in 4C-seq target selection.

CONTENTS

1	Introduction.....	16
1.1	Epigenetics.....	17
1.1.1	DNA methylation	17
1.2	Tools for analysing DNA methylation	18
1.2.1	Experimental tools.....	18
1.2.2	Bioinformatics tools	22
1.3	Open chromatin and DNase I hypersensitive sites.....	40
1.4	Histone marks	45
1.5	3D genomics.....	51
1.6	Genomics of disease.....	56
1.7	Epigenomics of disease	60
1.8	Thesis aims	61
1.8.1	Design and implementation of FORGE2	62
1.8.2	Design and implementation of eFORGE	62
1.8.3	4C-seq of neurological and autoimmune disorder-associated regions. 63	
2	Materials and Methods.....	64
2.1	4C-seq of neurological and autoimmune disorder-associated regions....	65
2.1.1	Samples.....	65
2.1.2	4C-seq.....	66
2.2	eFORGE methods	77
2.2.1	Introduction.....	77
2.2.2	Sample processing.....	78
2.2.3	Preparation of sample overlaps.....	80
2.2.4	Analysis strategy	80
2.2.5	Annotation categories	82
2.2.6	Proximity filtering.....	84
2.2.7	eFORGE code structure.....	85
2.2.8	eFORGE input.....	87
2.2.9	eFORGE outputs.....	88
2.2.10	Multiple testing correction and false positive rates	89
2.2.11	Database extension and generation	91
2.2.12	tDMP and cDMP analysis.....	91
2.2.13	Source code	92
2.3	FORGE2 methods	93
2.3.1	Introduction.....	93
2.3.2	Sample processing.....	94
2.3.3	Preparation of sample overlaps.....	95
2.3.4	Analysis strategy	95
2.3.5	Annotation categories	97
2.3.6	LD filtering	97
2.3.7	FORGE2 input.....	98
2.3.8	FORGE2 outputs.....	99
2.3.9	Multiple testing correction and false positive rates.....	101
2.3.10	GWAS analysis.....	102
2.3.11	Source code	102

3	eFORGE Results.....	103
3.1	tDMP analysis	104
3.2	EWAS findings.....	105
4	FORGE2 results.....	118
4.1	FORGE2 analysis of the GWAS catalogue	119
4.2	Analysis of psoriasis GWAS data	128
4.3	Conclusions on GWAS analyses	131
5	4C-seq results.....	132
5.1	Introduction	133
5.2	4C-seq of regions homologous to neural disease loci in mNSCs	133
5.3	4C-seq of immune disease loci in sorted blood cells.....	147
5.3.1	4C-seq analysis of the LRRC8B locus.....	148
5.3.2	4C-seq analysis of the IKZF3 locus	152
5.3.3	4C-seq analysis of the NCF4 locus.....	156
5.3.4	4C-seq analysis of the CSF2 locus	158
5.3.5	4C-seq analysis of the CCR6 locus.....	161
5.3.6	4C-seq analysis of the TRAF1/C5 locus.....	164
6	Discussion.....	169
6.1	eFORGE.....	170
6.1.1	eFORGE aims and context.....	170
6.1.2	eFORGE findings and applications.....	174
6.1.3	Considerations and limitations of eFORGE	180
6.1.4	Outlook on eFORGE	181
6.2	FORGE2	183
6.2.1	FORGE2 aims and context	183
6.2.2	FORGE2 findings.....	184
6.2.3	Considerations on FORGE and FORGE2	187
6.2.4	Outlook on FORGE2.....	187
6.3	4C-seq analysis.....	188
6.3.1	4C-seq analysis aims and context.....	188
6.3.2	4C-seq analysis findings	192
6.3.3	Outlook on 4C-seq analysis.....	194
7	References	198

LIST OF TABLES

Table 1.1: 450k packages.....	24
Table 2.1: BLUEPRINT samples	65
Table 2.2: Mouse target loci.....	71
Table 2.3: Human target loci	73
Table 2.4: Human-mouse homologous regions	75
Table 2.5: eFORGE sample processing.....	79
Table 2.6: eFORGE background probe matching	83
Table 2.7: eFORGE background bin subcategories.....	84
Table 2.8: eFORGE ENCODE false positives.....	90
Table 2.9: eFORGE Epigenomics Roadmap false positives	91
Table 3.1: list of EWAS in the eFORGE catalogue	117
Table 4.1: Common findings between FORGE2 and FORGE.....	125
Table 4.2: FORGE2 findings not detected in FORGE analyses.	125
Table 4.3: FORGE2 findings that show a different tissue from FORGE.....	125
Table 4.4: Additional tissue FORGE2 findings.....	126

LIST OF FIGURES

Figure 1.1: 450k probe types	20
Figure 1.2: 450k analysis pipeline	23
Figure 1.3: Multidimensional scaling	27
Figure 1.4: DNA methylation density plot	28
Figure 1.5: 450k sample clustering	29
Figure 1.6: Singular value decomposition heatmap	32
Figure 1.7: FEM analysis	37
Figure 1.8: Cell composition effects	39
Figure 1.9: DNase-seq protocol	41
Figure 1.10: FAIRE-seq protocol	42
Figure 1.11: ATAC-seq protocol	43
Figure 1.12: Developmental landscape and open chromatin	44
Figure 1.13: Epigenetic Dashboard	46
Figure 1.14: Overview of 3C-related technologies	53
Figure 2.1: Process for selecting neurological disorder-associated variants	68
Figure 2.2: Process for selecting immunity-associated loci	69
Figure 2.3: FORGE2 H3K4me1 analysis of human target loci	74
Figure 2.4: eFORGE analysis strategy	81
Figure 2.5: eFORGE directory schematic	86
Figure 2.6: eFORGE code schematic	87
Figure 2.7: FORGE2 analysis schematic	96
Figure 3.1: eFORGE tDMP analysis	104
Figure 3.2: eFORGE analysis of autoimmune disease EWAS	106
Figure 3.3: eFORGE analysis of an EWAS performed on a surrogate tissue	108
Figure 3.4: eFORGE analysis of 5 cancer EWAS	111
Figure 3.5: Aggregated eFORGE results for analysis across 20 EWAS	113
Figure 3.6: Genomic distribution and tissue origin from studies in the eFORGE catalogue	115
Figure 4.1: Types of FORGE2 findings when compared to FORGE	120
Figure 4.2: FORGE and FORGE2 PR interval GWAS analysis	126
Figure 4.3: FORGE and FORGE2 Alzheimer's disease GWAS analysis	127
Figure 4.4: FORGE2 psoriasis GWAS analysis	129
Figure 5.1: Schematic of 4C-seq mNSC computational and experimental analysis	135
Figure 5.2: 4C-seq results for the mouse region homologous to the human locus that contains Alzheimer's disease EWAS DMP cg02672452	139
Figure 5.3: 4C-seq results for the mouse region homologous to the human locus that contains schizophrenia-associated GWAS SNP rs548181	140
Figure 5.4: 4C-seq results for the Gcdh locus in mouse NSCs	143
Figure 5.5: 4C-seq results for the Synj2 locus in mouse NSCs	144
Figure 5.6: 4C-seq results for the Ppm1m locus in mouse NSCs	145
Figure 5.7: 4C-seq positive control	146
Figure 5.8: Schematic of computational and experimental analysis for human immune GWAS regions	148
Figure 5.9: 4C-seq results for the LRRC8B locus	151
Figure 5.10: 4C-seq results for the IKZF3 locus	155
Figure 5.11: 4C-seq results for the NCF4 locus	157

Figure 5.12: 4C-seq results for the CSF2 locus.....	160
Figure 5.13: 4C-seq results for the CCR6 locus.....	163
Figure 5.14: 4C-seq results for the TRAF1/C5 locus.....	166

LIST OF ABBREVIATIONS AND ACRONYMS

ATAC-seq: Assay for Transposase Accessible Chromatin with high-throughput sequencing

BED: Browser extensible data

BH: Benjamini–Hochberg

BMI: Body mass index

BMIQ: Beta MIxture Quantile dilation

BSA: Bovine serum albumin

BY: Benjamini–Yekutieli

CHD: Congenital Heart Defect

CHiCP: Capture HiC Plotter

CLL: Chronic Lymphocytic Leukaemia

CNV: Copy number variation

COMET: Block of comethylation

CRISPR: Clustered regularly interspaced short palindromic repeats

ChIA-PET: Chromatin Interaction Analysis by Paired-End Tag Sequencing

ChIP-seq: Chromatin immunoprecipitation with high-throughput sequencing

CpG: Cytosine-guanine dinucleotide

DHS: DNase I hypersensitive site

DMC: Differentially methylated COMET

DMP: Differentially methylated position

DMR: Differentially methylated region

DNAm: DNA methylation

DNaseI: Deoxyribonuclease I

DVP: Differentially variable position

EWAS: Epigenome-wide association study

FACS: Fluorescence-activated cell sorting

FAIRE: Formaldehyde-Assisted Isolation of Regulatory Elements

FDR: False discovery rate

FISH: Fluorescence in situ hybridisation

FORGE2: Functional element Overlap analysis of Regions from GWAS

Experiments 2

FPR: False positive rate

GEE: Generalised estimating equation

GO: Gene ontology

GSEA: Gene set enrichment analysis

GWAS: Genome-wide association study

IBD: Inflammatory bowel disease

IGR: Intergenic region

IHEC: International human epigenome consortium

IPA: Ingenuity pathway analysis

KEGG: Kyoto encyclopedia of genes and genomes

LD: Linkage disequilibrium

LMA: Ligation-mediated amplification

LMM: Linear mixed model

LPS: Lipopolysaccharide

MAF: Minor allele frequency

MALT: Mucosa-associated lymphoid tissue

MDS: Multidimensional scaling

MEGDEL: 3-methylglutaconic aciduria, deafness, encephalopathy, and Leigh-like disease

MS: Multiple sclerosis

MVP: Methylation variable position

MeDIP: Methylated DNA immunoprecipitation

PCA: Principal component analysis

PCHi-C: Promoter Capture-HiC

PCR: Polymerase chain reaction

RA: Rheumatoid arthritis

RNase: Ribonuclease

RRBS: Reduced representation bisulphite sequencing

SLE: Systemic lupus erythematosus

SNP: Single nucleotide polymorphism

SSc: Systemic sclerosis

SVD: Singular value decomposition

SWAN: Subset-quantile within array normalisation

T1D: Type 1 diabetes

TE: Tris and ethylenediaminetetraacetic acid

TFBS: Transcription factor binding site

TSH: Thyroid-stimulating hormone

TSS: Transcription start site

UC: Ulcerative Colitis

UTR: Untranslated region

WGBS: Whole-genome bisulphite sequencing

cDMP: Cell type-specific differentially methylated position

ccRCC: Clear cell renal cell carcinoma

eFORGE: Experimentally-derived Functional element Overlap analysis of
ReGions from EWAS

eQTL: Expression quantitative trait locus

eQTM: Expression quantitative trait methylation

iPSC: Induced pluripotent stem cell

mNSC: Mouse neural stem cell

meQTL: Methylation quantitative trait locus

qPCR: Quantitative polymerase chain reaction

tDMP: Tissue-specific differentially methylated position

tDMR: Tissue-specific differentially methylated region

LIST OF APPENDICES

SUPPLEMENTARY TABLES

Table S1: sequences of PCR primers targeting mouse regions homologous to human GWAS/EWAS loci.....	231
Table S2: BLUEPRINT 4C-seq primers	235
Table S3: tDMP sample IDs	250

SUPPLEMENTARY FIGURES

Figure S1: Gcdh locus homology.....	228
Figure S2: Ppm1m locus homology	230

PUBLICATIONS	251
--------------	-----

eFORGE ANALYSIS CODE	261
----------------------	-----

1 Introduction

1.1 Epigenetics

Epigenetics studies the “stably heritable phenotype resulting from changes in a chromosome without alterations in the DNA sequence”¹. Histone marks and DNA methylation (DNAm) rank among the main epigenetic factors. In addition, there is an increasing understanding of the importance of three-dimensional genomic context in regulating gene expression. Particularly, the role of enhancers, sequences of DNA that loop to gene promoters to stimulate transcription, is essential to achieve physiological transcription rates of target genes². Here I present work that focuses on all of these aspects and their role in disease.

1.1.1 DNA methylation

5-methylcytosine is formed when a methyl (CH₃) group is added to the carbon-5 position of cytosine. This is currently the most commonly studied form of nucleotide methylation. Methylated genomic cytosines are predominantly found forming part of cytosine-guanine dinucleotides (CpG). Non-CpG DNAm, though less frequent, is also becoming increasingly relevant in research, thanks to its recent detection in adult tissues³. Humans are diploid organisms, that is, they have two full sets of chromosomes (23 chromosomes from the father and 23 chromosomes from the mother). The sequence of each set of chromosomes is a haploid genotype (or haplotype). Most of the approximately 28 million CpGs in the haploid human genome are present in a methylated state. Functionally, DNAm is generally considered a silencing mark, although the specific role of DNAm can vary with context. DNAm at transcription start sites can prevent

initiation of transcription, while DNAm at gene bodies may stimulate transcription elongation⁴.

1.2 Tools for analysing DNA methylation

1.2.1 Experimental tools

Among the different technologies used to measure DNA methylation are sequence-based technologies and array-based technologies. Both of these technologies apply bisulphite to convert unmethylated cytosines to uracil, thus distinguishing methylated cytosines from unmethylated cytosines. Subsequent PCR reactions will amplify uracil as thymine, while 5-methylcytosines will be amplified as cytosines.

Sequence-based technologies include whole-genome bisulphite sequencing, reduced presentation bisulphite sequencing and pyrosequencing-related technologies. Whole-genome bisulphite sequencing is the most costly of the sequencing-based technologies, as it covers the entire genome and requires a read depth of at least 30x and, ideally, duplicates⁵. Because of the prohibitive cost of whole-genome bisulphite sequencing, reduced representation bisulphite sequencing was developed^{6,7}. This technology enriches sequencing libraries for certain base content of the genome. Depending on the restriction enzymes used, regions with a high CpG, such as CpG islands, can be targeted.

Array-based technologies include those developed by Illumina and Affymetrix. Illumina arrays are the most common DNAm arrays used in the field of epigenetics research⁸. These arrays target a subset of genomic CpGs and present a cost-effective and highly reproducible way of assaying DNA methylation. Initially, Illumina developed the Golden Gate assay, focusing on 1505 cytosines of interest in disease and development⁹. Given the success of the GoldenGate assay, Illumina scaled up the number of regions in its successive arrays, which include the 27k array, the 450k array and the 850k (or EPIC array). The name of each of these arrays indicates the number of cytosines analysed. It is important to note that Illumina array development was not reduced to simply increasing the number of regions but also includes important modifications in the chemical assays employed to detect DNAm. Among these modifications one of the most important is the presence of type I and type II probes on Illumina arrays, which has been the focus of much statistical study in the development of analysis pipelines (including normalisation methods such as BMIQ or SWAN, in pipelines such as ChAMP¹⁰, minfi¹¹, RnBeads¹², methylumi or waterRmelon¹³).

The more detailed description of type I and II probes in Illumina array chemistry is shown in **figure 1.1**.

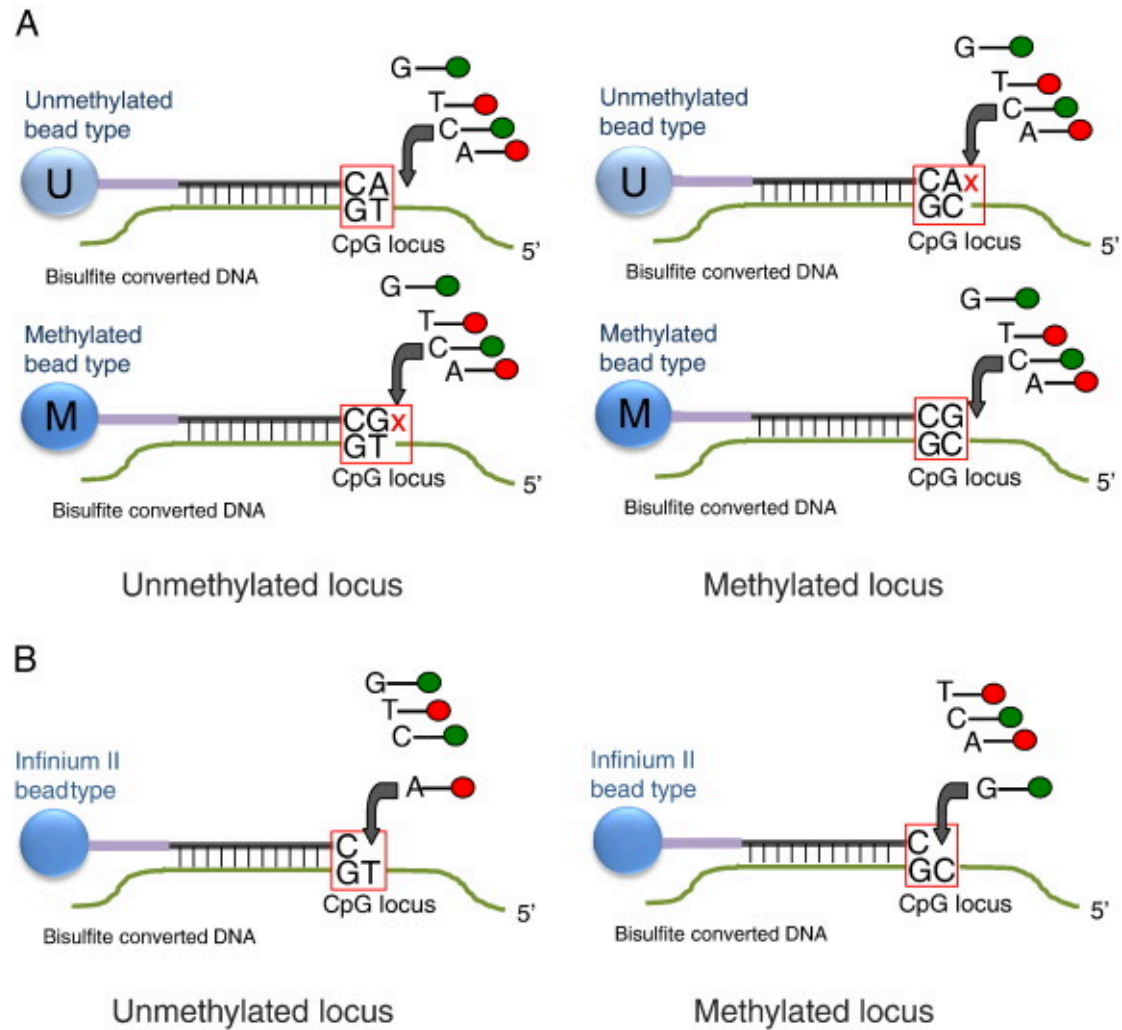


Figure 1.1: 450k probe types. Type I (A) and type II probes (B). In type I design, each genomic locus is assayed by two bead types, one for unmethylated (U) and one for methylated (M), on the same colour channel. In type II design, each genomic locus is assayed by one bead type, with different colour channels for methylated and unmethylated cytosines. For both probe types DNAm is detected on bisulphite converted DNA by allele-specific single-base extension. A fluorescent label is incorporated in this extension reaction, allowing the identification of DNAm. Original figure from Bibikova et al., 2011¹⁴.

Reproducibility between Illumina arrays has been demonstrated to be very high ($R^2 > 0.95$)^{14,15}. This high reproducibility, coupled with easy implementation, a low price and the availability of multiple analysis pipelines has contributed to establishing Illumina arrays as the leaders in the field, especially as a main technology in epigenome-wide association studies (EWAS).

EWAS represent a novel approach in the study of human disease, studying changes in DNAm that may result from disease progression (as opposed to genetic predisposition), and/or environmental factors such as smoking¹⁶. EWAS study the relationship of DNA methylation with disease in large cohorts in a way analogous to GWAS. However, unlike GWAS, EWAS are subject to scrutiny on potential confounding by cell composition effects^{8,17,18}. Therefore study design is vitally important in the field of EWAS, as EWAS benefit greatly from measures that reduce confounding, such as cell sorting methods.

An extensive comparison of DNAm technologies falls outside the scope of this thesis, but I recommend consulting the DNAm technology benchmarking paper produced by the BLUEPRINT consortium¹⁹.

1.2.2 Bioinformatics tools

DNAm array analysis tools.

In epigenetics, one of the main areas of active development of bioinformatics pipelines focuses on Illumina DNAm array technologies, such as the Illumina 450k array.

After inaugural efforts from Illumina (Genome Studio software), the field saw a rapid development of a range of pipelines (e.g. ChAMP¹⁰, minfi¹¹, RnBeads¹², methylumi²⁰ and watermelon¹³). Different pipelines offer varying utilities, and depending on the focus in analysis one pipeline or another may be preferred. An extensive list of 450k analysis packages is shown in **table 1.1**. In addition, a comparative list of analysis steps included in the main pipelines is shown in **figure 1.2**.

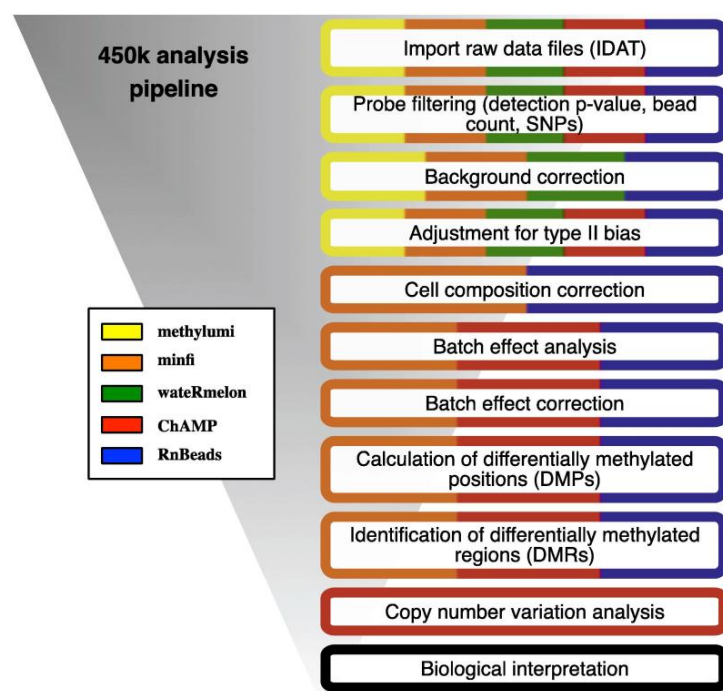


Figure 1.2: 450k analysis pipeline. List of main pipelines for 450k analysis, alongside the analysis steps performed. Colours indicate which pipeline includes each specific analysis step. Original figure from Morris and Beck, 2015²¹.

Package	Use
ChAMP	Comprehensive suite of functions; automated pipeline
COHCAP	CpG island analysis and gene expression data integration
Comb-p	DMR calling
DMRcate	DMR calling
Epigenetic clock	Predictor of sample age
EWasher	Reference-free cell composition correction
FastDMA	Quantile normalisation and DMP/DMR calling
IMA	Preprocessing including normalisation methods; Pipeline option
Lumi	Background correction, general normalisation
Marmal-aid	450k database for data integration
MethylAid	Interface for interactive sample QC
Methylumi	Comprehensive suite of functions
Minfi	Comprehensive suite of functions
NIMBL	Matlab code for QC and DMP calling
RefFreeEWAS	Reference-free cell composition correction
RnBeads	Comprehensive suite of functions
shinyMethyl	Interface for interactive sample QC
watermelon	Preprocessing including performance metrics and numerous normalisation methods

Table 1.1: 450k packages. A list of freely available packages for 450k analysis, including details on the application of each package. Original table from Morris and Beck, 2015²¹.

Most of the pipelines for 450k analysis share a series of common elements. The main processes include¹⁰:

1. Data loading
2. QC
3. Normalisation
4. SVD/PCA
5. Correction of batch effects
6. Identifying differentially methylated positions (DMPs) and regions (DMRs)
7. Downstream functional analysis (pathways, GO, FEM)

Additional elements in pipelines include¹⁰:

- Detecting copy number variants (CNVs, e.g. ChAMP)
- Correcting cell composition effects (e.g. minfi, recent versions of ChAMP).

Data loading

There are a variety of data formats in 450k analysis. One of the most common formats, the IDAT format, comprises data from the raw images of the arrays. This “raw data” format produces very large files and is hard to transfer and manage,

especially for large-scale studies. Therefore another format, the “beta matrix”, is also used as part of 450k data analysis, especially as part of meta-analyses or for replication of studies. This matrix includes a series of beta-values (DNAm values between 0 and 1) for each of the probes on the 450k array. Some pipelines, including ChAMP, support initial analysis from a beta matrix as well as from IDAT files.

In addition to DNAm probes, there are 65 SNP probes on the 450k array (control probes). The analysis pipeline must filter these control probes and also any DNAm probes with failed measurements. The 450k array includes 485,512 DNAm probes, and the EPIC array includes 867,531 DNAm probes. Of these probes, a percentage may fail. Probe filtering usually includes¹⁰:

- probes that have failed to hybridise (and thus have a detection p-value < 0.01). Samples with a high number of failed probes may be removed from analysis.
- probes represented by fewer than 3 beads on the array.
- SNP-related probes. If SNPs are shown to have a very large effect on DNAm results the relevant probes should be excluded from analysis²².
- Cross-hybridising probes. A number of probes that target multiple genomic loci have been identified, including autosomal probes that cross-hybridise to the sex chromosomes²².
- non-CpG probes. Non-CpG DNAm presents important features that distinguish it from the more common CpG DNAm, such as an enrichment

in stem cells and pluripotent cells²³. Non-CpG probes are typically excluded from analysis, though not always²⁴.

- probes located on chromosomes X and Y. These probes require separate analysis in cohorts that contain both men and women, as they could otherwise be driving spurious DNAm signals. These probes are often excluded from data processing.

Quality Control (QC): it is vital to check the quality of the data before normalisation. The strategies typically used to check data quality include Multidimensional Scaling (MDS), density plots and clustering methods¹⁰:

-MDS (Multidimensional Scaling): this method aims to represent the degree of similarity between individual samples analysed. Each datapoint in MDS plots represents a sample from the study, and datapoints are grouped by similarity to other datapoints. MDS plots are a way to display information contained in a distance matrix (**figure 1.3**).

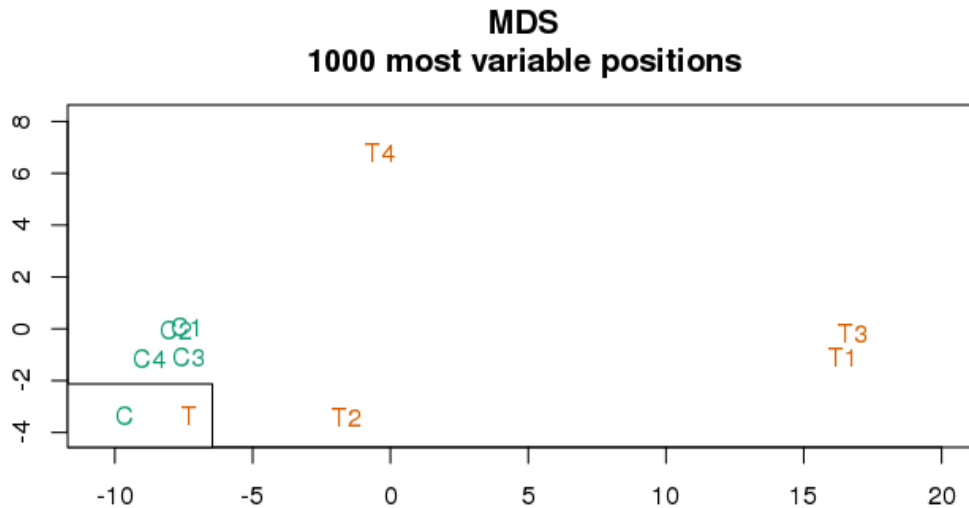


Figure 1.3: Multidimensional scaling. This MDS plot separates the green samples from the brown samples, reflecting the distance between sample groups C and T. Original figure from ChAMP package vignette¹⁰.

-Density plots: DNAm follows a standard bimodal distribution. Alterations of this bimodal curve can be detected by observing the beta distribution line across samples (**figure 1.4**). Additional peaks between the two expected peaks from the bimodal distribution may signal a lack of sample quality.

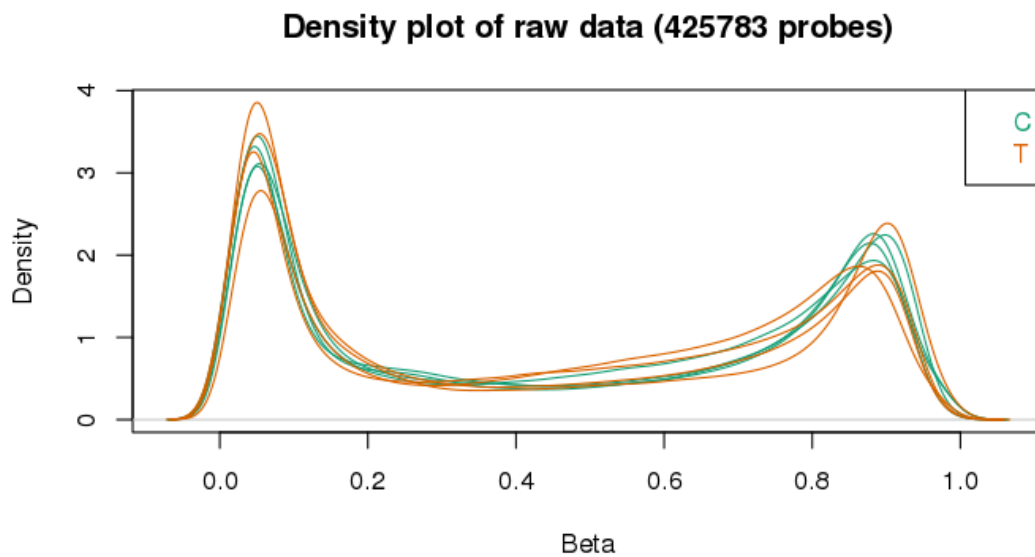


Figure 1.4: DNA methylation density plot. Density plot with the classic bimodal distribution of DNAm beta values. The peak close to zero corresponds to unmethylated sites across the genome. The peak close to one corresponds to methylated sites, and the rest of the bimodal curve corresponds to sites that present intermediate methylation values. Original figure from ChAMP package vignette¹⁰.

-Clustering methods: a typical hierarchical clustering plot (or dendrogram) shows the ordered similarity relationships between samples based on the data (**figure 1.5**). If a particular group of samples do not cluster well or if individual samples cluster in an unexpected group this may point to either low sample quality or labelling errors.

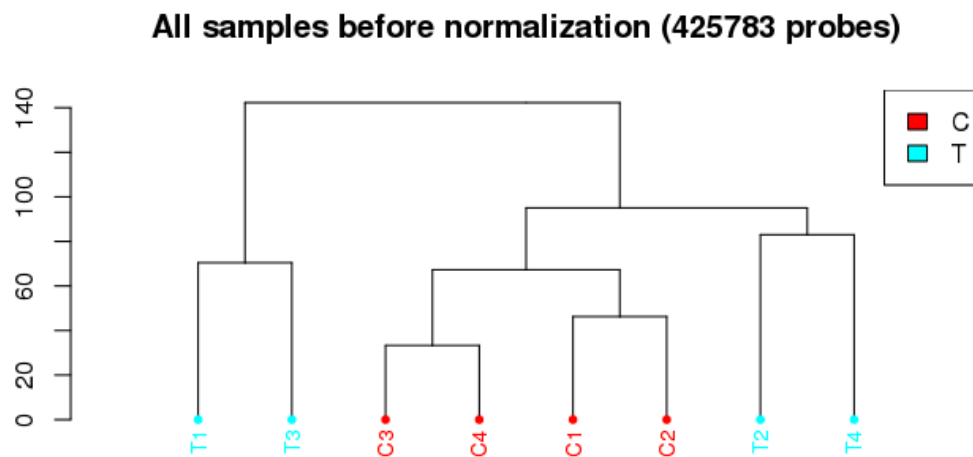


Figure 1.5: 450k sample clustering. Clustering dendrogram showing adequate clustering separating sample groups C (red) and T (blue). Original figure from ChAMP package vignette¹⁰.

Normalisation:

I have previously explained the difference between type I and type II probes (**figure 1.1**). Due to this difference in design, type I and II probes have different beta distributions. This difference in distribution, if uncorrected, could potentially lead to spurious results in downstream analysis. To minimise the effect of probe type in a given study (which is a confounder when trying to find DNAm differences between cases and controls), it is necessary to apply a

normalisation method (several are available -e.g. SWAN²⁵, BMIQ²⁶, PBC²⁷, and Funnorm²⁸).

The characteristic feature of each of the main methods is the following:

SWAN: “subset-quantile within array normalisation”, this method makes the key assumption that the amount of CpGs within the 50 bp probe sequence reflect the underlying biology of the region in question²⁵. This information is then used to adjust type I and type II probe distribution, basing the adjustment on a subset of randomly selected type I and II probes that have one, two and three underlying CpGs.

BMIQ: this method first applies a three-state beta-mixture model to assign probes to DNAm states²⁶. Quantiles are then formed from the probabilities that result from the beta-mixture model. Finally, this method adjusts the data by performing a DNA methylation-dependent dilation transformation.

PBC: this method rescales type I and type II probe values simply by shifting the summits of the two peaks in the bimodal distribution of DNAm values (that is, the methylated and unmethylated peak).

Funnorm: this method considers any difference in the 848 control probes on the 450k array as unwanted technical variation, and adjusts the data accordingly. It is important to note that sometimes a proportion of these 848 control probes do not pass array QC. For these cases the effectiveness of this method is limited.

SVD:

SVD (singular value decomposition²⁹) analysis is a matrix factorisation process performed to identify the main components of variation. These components can be either technical or biological.

SVD analysis results can be plotted as a heatmap (**figure 1.6**) showing the top principal components correlated with the covariates information provided. SVD can highlight important confounders or batch effects in the data that may require adjustment in downstream analysis (as described in the next section).

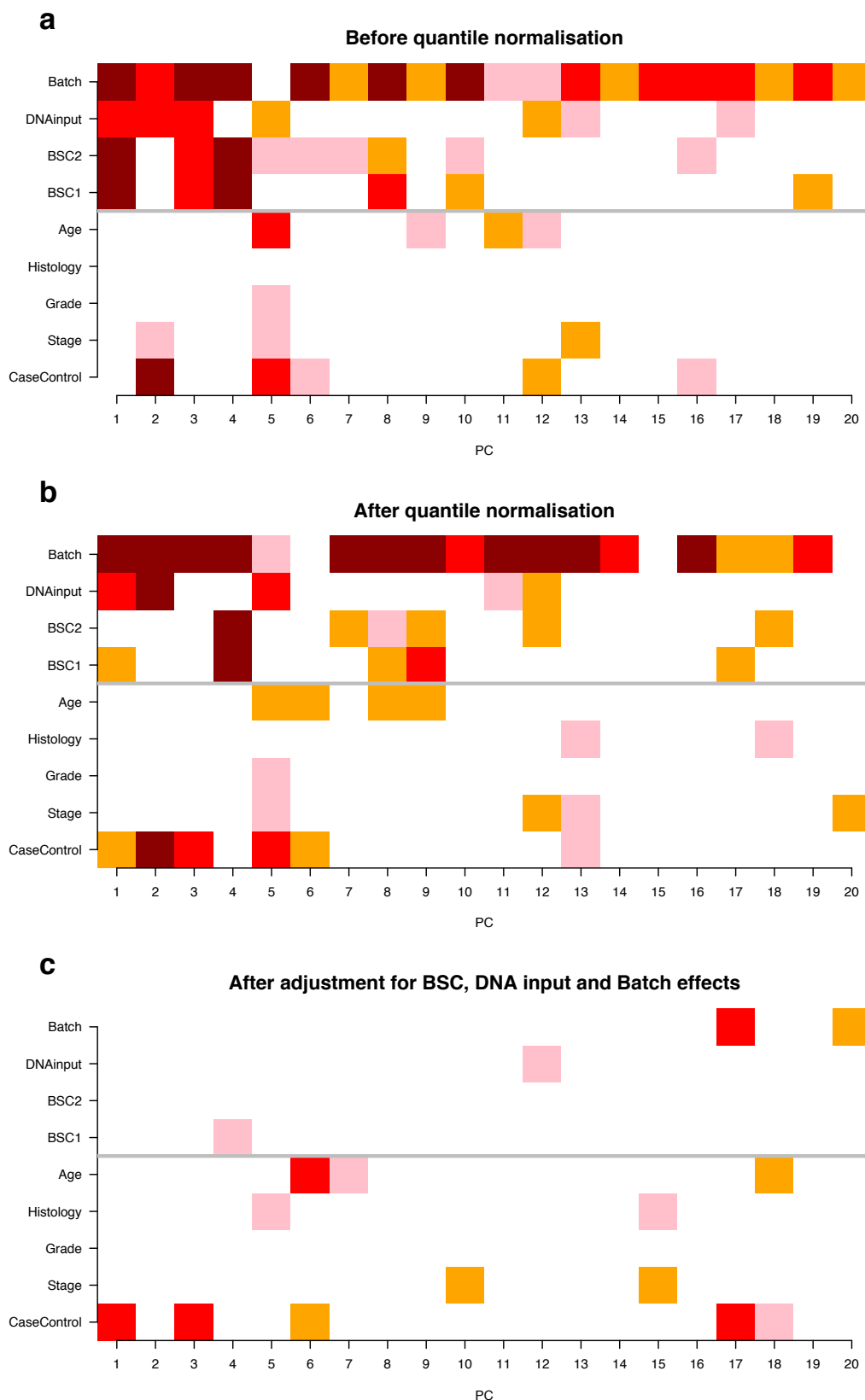


Figure 1.6: Singular value decomposition heatmap. This heatmap identifies the top 20 principal components underlying variation from an EWAS on ovarian

cancer before quantile normalisation (a), after quantile normalisation (b) and after adjustment for bisulphite conversion (BSC), DNA input and batch effects (c). The grey line separates phenotypic factors (age, case-control status, stage of cancer, cancer grade and histological subtype) from technical factors (batch number, DNA input, bisulphite conversion efficiency controls 1 and 2). Variation due to these technical factors represents a source of confounding, and is removed in (c). Original figure from Teschendorff et al., 2009²⁹.

Adjustment for batch effects:

A batch effect is a technical source of variation introduced during sample handling. For example, if one day the temperature in the lab is higher, or if the person handling the arrays is different, then the array measurements will differ slightly. Randomised sample design (in which cases and controls are assigned randomly to different array batches) helps to reduce the confounding caused by batch effects.

A computational tool named ComBat³⁰ has been implemented in packages such as ChAMP to correct for batch effects in 450k array analysis. ComBat uses an empirical Bayes framework (with options for parametric and non-parametric priors) to input previously known information to estimate sources of technical

variation. ComBat has proven to be an efficient method for reducing unwanted sources of variation, and has been reported to reduce in certain cases up to 98% of batch effects³¹.

Differentially methylated probes/regions/comets:

Differential DNAm can be expressed at different levels, from the single cytosine to large regions spanning megabases. For a brief description of the different categories employed in differential DNAm studies:

- Differentially methylated position (DMP): differential DNA methylation detected at the level of a single cytosine. As noted above, the 450k array contains 485,512 different probes, and each probe targets a specific genomic cytosine. Probes that detect significant DNA methylation changes between cases and controls are classed as Differentially Methylated Positions (DMPs).
- Differentially methylated region (DMR): a group of proximal cytosines with similar methylation change can be grouped into a differentially methylated region (DMR). Methods for identifying DMRs include Probe Lasso³², bumhunter³³, comb-p³⁴ and DMRcate³⁵, among others.

In bumhunter, DMRs are detected as peaks in a representation of smoothed DNAm values. Probe Lasso gathers proximal significant

positions using a flexible window approach, in an effort to combat CpG density bias. DMRcate tackles the same problem using a Gaussian kernel to fit repeated DNAm measurements spatially across the genome. Finally, comb-p employs spatially assigned p-values to detect regions of enrichment.

- COMETs: The study of the oscillations of genomic DNAm values has led to the definition of COMETs³⁶, or blocks of comethylation, which are segmentations of the genome into areas of continual smoothed DNAm level estimates. One of the uses of COMETs is that they allow for the recovery of methylome feature information for non-saturating levels of sequencing depth in whole-genome bisulphite sequencing. A package, named “cometvintage”, has been developed to identify differentially methylated comets (DMCs). In regards to COMET structure, it is important to consider that a significant correlation between COMETs and haplotype blocks has been reported³⁶, signalling a potential epigenetic equivalent of the haplotype map. At a local level, it had previously been known that DNAm levels correlated within 1kb^{37,38}.

Functional/pathway analysis

After uncovering differentially methylated positions, regions or COMETs, there are further steps that have to be taken to identify the biological relevance of the differential DNAm observed. One of the classical ways to do this is to perform a Gene Ontology (GO) analysis^{39,40}, which uses a hierarchical set of terms encompassing most of the known biological processes, cellular components and molecular functions. Additional pathway analysis may be done, for example using the pathway catalogue developed by the Kyoto Encyclopaedia of Genes and Genomes (KEGG)⁴¹.

Another way to extract biological meaning from observed differential DNAm is to perform a Gene Set Enrichment Analysis (GSEA)⁴². This method presents a markedly different approach to the “statistical enrichment of overlap analysis” performed by tools such as DAVID^{43,44}, Ingenuity (IPA; Qiagen; <http://www.qiagen.com/ingenuity>) or AmiGO⁴⁵. In the aforementioned tools, the study list of genes (e.g. those genes with promoters that overlap differentially methylated regions) is compared to a randomly sampled set of gene lists from the appropriate background pool (e.g. the array used for the study). The top study list is thus assigned an enrichment score through a statistical test such as a binomial test, a Fisher’s exact test, a Chi-squared test or a hypergeometric test. However, the GSEA approach does not only use the top study list of genes, it uses the whole list of genes from the study (e.g. those that are differentially methylated and those that are not). This inclusion of additional information makes for a different approach, in which random permutations of the total list of

genes are performed to calculate a normalised enrichment value for the distribution of genes of a particular group. In this approach one group of genes may have a tendency to be higher up on the list, thus showing association with the phenotype under study. The appropriate statistical tests used for this list permutation analysis are the Wilcoxon test and the Kolmogorov-Smirnov test.

Complementing the previous methods, the FEM package⁴⁶ integrates protein interaction networks with DNAm data, identifying a subnetwork of protein interactions formed by genes highlighted by the differential DNAm results. An example of FEM output is given in **figure 1.7**. FEM can be used to integrate DNAm and gene expression data, or alternatively upon DNAm data alone.

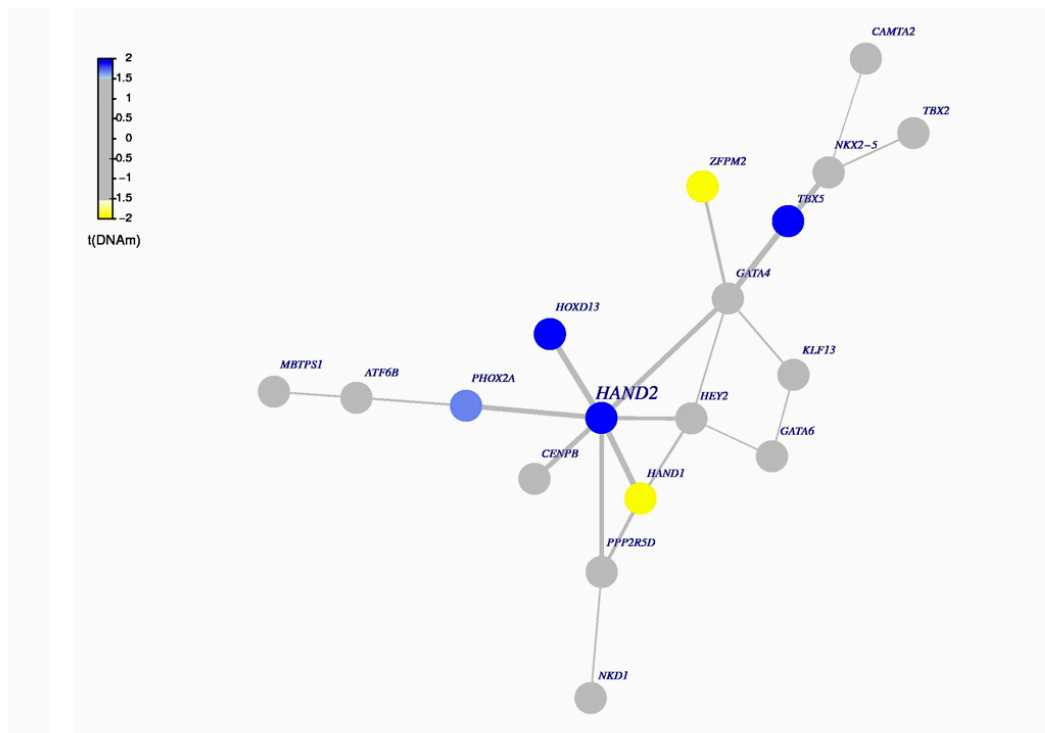


Figure 1.7: FEM analysis. Results show one protein interaction network identified using FEM analysis of differential DNAm results. Original figure from ChAMP package vignette¹⁰.

Finally, the iEVORA package⁴⁷ complements standard EWAS analysis of Differentially Methylated Positions by analysing Differentially Variable Positions. In the analysis of Differentially Methylated Positions linear mixed models are employed to detect genomic cytosines that present different methylation values in cases and controls. In iEVORA a different approach is implemented with the aim to find differentially *variable* regions, in which the *variation* of DNAm is different between cases and controls (e.g. there is a difference in the standard deviation of methylation values between cases and controls for a given genomic cytosine). In order to detect this iEVORA performs Bartlett's test, and, due to the high sensitivity of this initial test to single outliers, iEVORA also performs a t-test to rank significant Differentially Variable Positions (DVPs).

Additional elements in an EWAS pipeline:

Cell composition effects:

As mentioned before, one of the main problems in EWAS is confounding by cell composition effects^{48,49,50,51,52} [reviewed in ^{8,17,18,53,54} and described in **figure 1.8**]. To this end, in 2012 Andres Houseman developed a correction method applicable to whole blood, which applies DNAm reference data from FACS sorted blood cells⁴⁸.

Because of the need to apply cell composition correction to non whole-blood-based EWAS, a variety of reference-free correction methods have been developed^{49,51,52}. These methods assume that confounding by cell type will be one of the main sources of variance, and some of these methods try to correct the effects of the main principal components detected when analysing the data. In addition, some researchers claim that overcorrection is being performed by some of these methods⁵⁰.



Figure 1.8: Cell composition effects. Proportions of cell types may be different between cases and controls, as is shown for cell types A (blue) and B (red). This difference in proportions may drive spurious DNAm signals and is thus a strong confounder to consider in EWAS analysis. Original figure from Liang and Cookson, 2014⁵³.

Correction for cell composition effects has become so commonplace in the field of epigenetics that some of the main bioinformatics pipelines (e.g. ChAMP) have incorporated a variety of these methods into their framework¹⁰. The resulting competition from different groups will no doubt bring forward advances in the correction of cell composition effects for a variety of tissues.

1.3 Open chromatin and DNase I hypersensitive sites

There are currently three main methods to identify open chromatin sites, namely DNase-seq⁵⁵, ATAC-seq⁵⁶ and FAIRE-seq^{57,58}. One way to measure sites of open chromatin in the genome is by the study of DNase I hypersensitivity. DNase I, as the name indicates, is an enzyme that degrades DNA. In the DNase-seq protocol, open chromatin sites are degraded, leaving blunt DNA ends at either side (see **figure 1.9**). These blunt ends are ligated with a biotinylated linker, digested with MmeI and bound to Dynal beads. A second linker is then ligated and a PCR is performed prior to sequencing. After sequencing, bioinformatics approaches are applied to identify the location in the genome of the segments that were initially cleaved and degraded with DNase I.

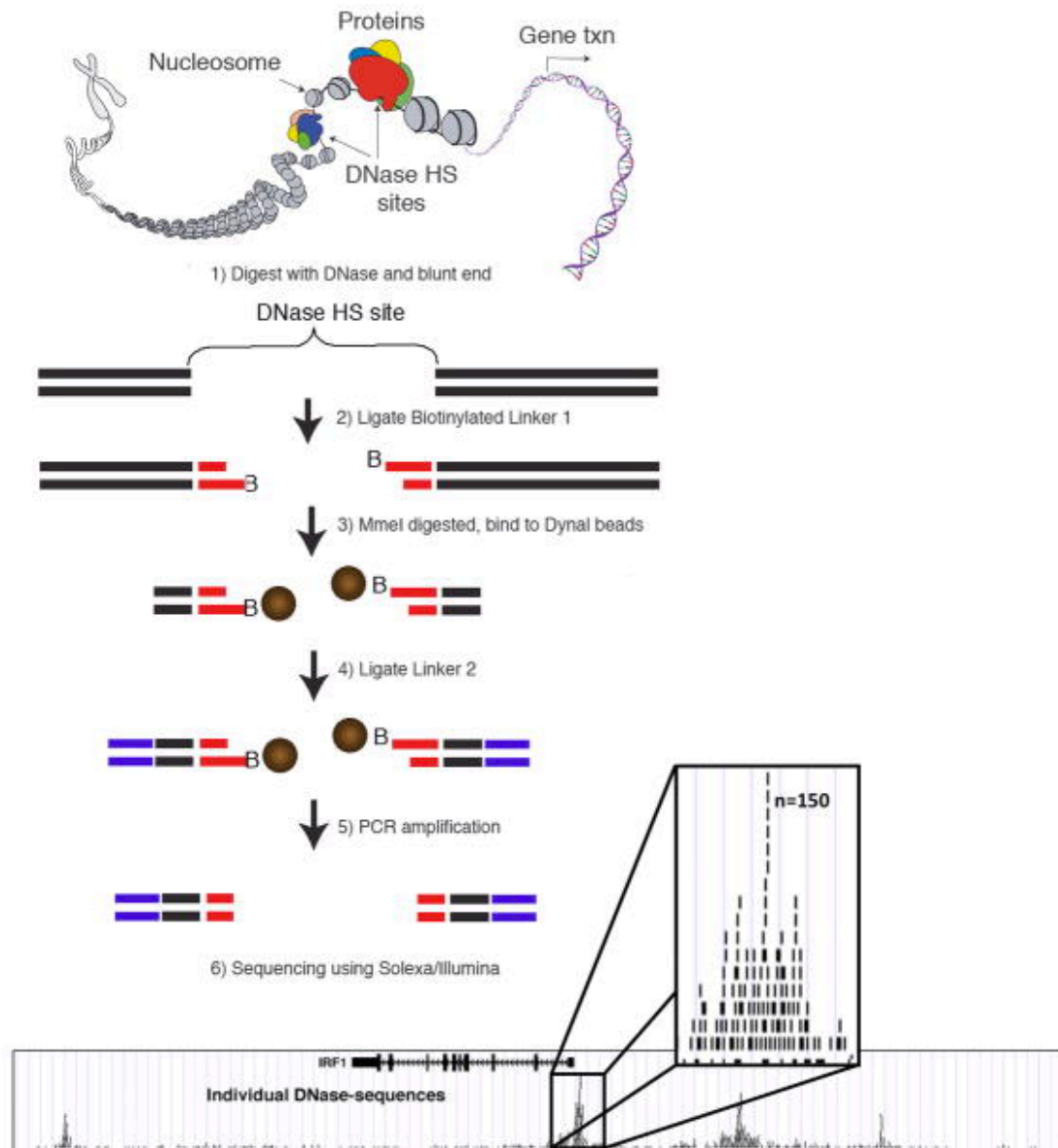


Figure 1.9: DNase-seq protocol. Peaks of DNase-seq are shown in the lower panel. Each one of these peaks corresponds to an open chromatin site (in this case also named a DNase I hypersensitive site or DHS). Original figure from Song and Crawford, 2010⁵⁵.

Another way to detect sites of open chromatin is to perform FAIRE-seq (formaldehyde-assisted isolation of regulatory elements). As the name indicates, this method employs formaldehyde to link proteins and DNA together. DNA is

then fragmented through sonication and extracted using a phenol-chloroform mixture. The phenol-chloroform mixture will retain the protein-linked DNA, and the DNA not linked to proteins (i.e. “open chromatin”) will remain in the aqueous phase (**figure 1.10**). Sequencing ensues, followed by bioinformatics analysis to pinpoint the location of the open chromatin sites.

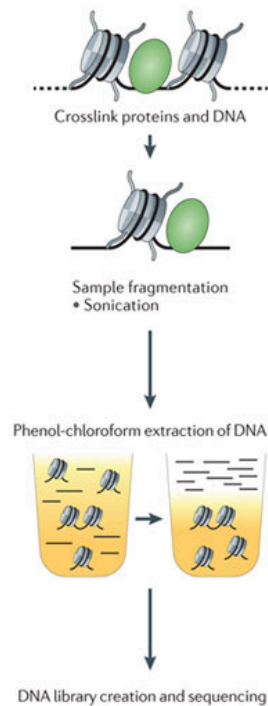


Figure 1.10: FAIRE-seq protocol. Formaldehyde crosslinking of proteins and DNA, followed by sonication, ensures that open chromatin sites will be separated from the rest of the genome through a phenol-chloroform extraction. Original figure from Furey, 2012⁵⁹.

In ATAC-seq, however, a different approach is used, based on the properties of transposons. Transposons, initially proposed by Barbara McClintock in the 1940s, are “DNA sequences that are able to move from one location to another in the genome”⁶⁰. Because of this property of moving from one genomic location to

another, transposons are involved in genomic instability, and perform a well-studied role in cancer⁶¹. However, in ATAC-seq the enzyme that catalyses the movement of transposons to other parts of the genome (transposase) is used to cut exposed DNA and ligate specific sequences named adapters to this region⁵⁶ (**figure 1.11**). Fragments of DNA are then isolated, PCR-amplified, sequenced and mapped to detect open chromatin sites. Currently, ATAC-seq mainly employs a mutated hyperactive version of transposase Tn5.

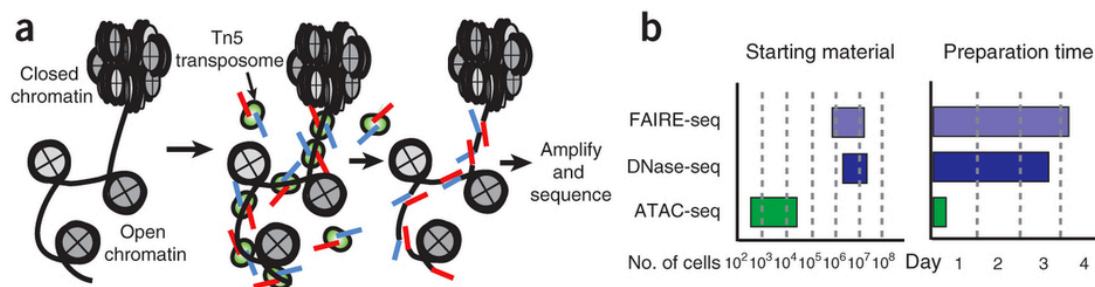


Figure 1.11: ATAC-seq protocol. A schematic of ATAC-seq is shown [A], showing the integration of Tn5 transposome in open chromatin sites, which is then detected by sequencing. In addition, ATAC-seq requires fewer cells as starting material and a shorter preparation time than competing methods DNase-seq and FAIRE-seq [B]. Original figure from Buenrostro et al., 2013⁵⁶.

As open chromatin regions are representative of enhancers and other genomic elements with a role in regulating gene expression⁶², it is natural to link the study of open chromatin to the study of genomic regulation. One specific case where this has been explored in depth is the case of development⁶³. For differentiated NK and Th2 immune cells, it has been shown that approximately

one third of DHSs are embryonic, and the other two thirds originate either from early differentiation phases or from late differentiation phases⁶³. These findings have been put in relation to the epigenetic landscape initially proposed by Conrad Waddington^{63,64} (**figure 1.12**). Other epigenetic features affected by this differentiation process include histone marks, DNAm and 3D genomics^{62,65}. These different epigenetic features are related⁶², as is shown by the success in computational approaches that use DHS and histone mark information to predict the 3D genomic landscape^{66,67}, linking up several of the main epigenetic features which are in turn linked to the regulation of gene expression⁶² and cellular phenotype⁶⁵.

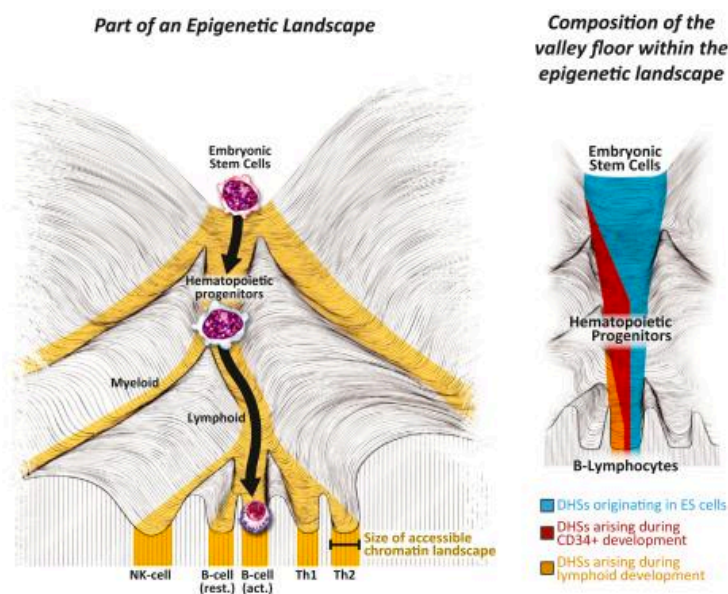


Figure 1.12: Developmental landscape and open chromatin. Waddington's epigenetic landscape and DHS origin for several immune cell types. The left panel shows the variation in size for the accessible chromatin landscape across several differentiation stages (stem cell, haematopoietic progenitors and differentiated immune cells). The right panel shows the gradual loss of stem cell DHSs (blue)

and the gain of haematopoietic DHSs (red) and lymphoid DHSs (yellow) during differentiation. Original figure from Stergachis et al., 2013⁶³.

1.4 Histone marks

In the chromatin context, DNA is wrapped around structures named nucleosomes. These nucleosomes are octamers, and each part of the octamer is a histone protein (H2A, H2B, H3 and H4). These four histone proteins are present twice in each nucleosome, forming the octamer. In addition, histone H1 is associated with the DNA that links consecutive nucleosomes. Histone H1 is known as the “linker” histone.

In addition to providing structural support for DNA, histones also provide a substrate for signalling chemicals to bind. Ongoing research continuously identifies many new types of histone modifications, and currently well over 100 histone modifications are known⁶⁸. Many of these modifications are listed in Huang et al., 2014⁶⁹. Much effort has been made to unravel a “histone code” that reflects the activity and assembly of the genome⁷⁰. It is known that certain histone marks co-localise with specific types of cis-regulatory elements⁷¹. Mechanistically, histone modifications may also act to establish regulatory elements both through *cis* (electrostatic effects that weaken the histone-DNA interaction) and *trans* effects (driven by effector protein binding and protein-protein recognition)⁷². A “dashboard” listing the regulatory states associated with the main histone marks is shown in **figure 1.13**.

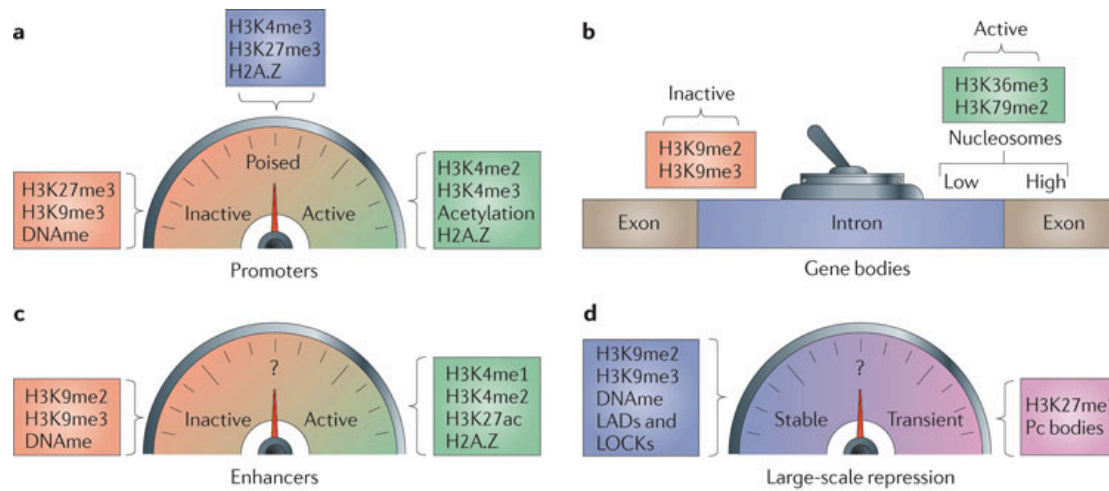


Figure 1.13: Epigenetic Dashboard. This dashboard contains the main histone marks and links to activation/repression for specific genomic contexts. Different marks separate active, inactive and poised promoters [A], including H3K4me3 (active and poised promoters) and DNAm (inactive promoters). Gene bodies can present repression-associated marks (e.g. H3K9me3) or marks linked to active transcription (e.g. H3K36me3) [B]. Enhancers typically present H3K4me1 and H3K27ac when active and can also present DNAm (inactive enhancers) [C]. Repressed regions can be separated into those that present transient repression (characterised by H3K27me3) and those that form constitutive heterochromatin (H3K9me3) [D]. Original figure from Zhou et al., 2011⁷³.

A brief description of each of the 6 main histone modifications⁷⁴ (according to IHEC guidelines, <http://www.ihec-epigenomes.org/>) is listed below:

-H3K4me3: the characteristic mark of the promoters of actively transcribed genes⁷¹. H3K4me3 is deposited by SET1A/SET1B⁷⁵, and readers of H3K4me3 include the V(D)J recombinase subunit RAG2⁶² and PHD finger proteins⁷⁶. One of

the PHD finger proteins is the RNAPII TAF3 subunit (part of TFIID). H3K4me3 is thus connected to the initiation of transcription through TFIID⁷⁷. H3K4me3 can also be found alongside H3K27me3 (a repressive mark) in the so-called bivalent domains, typical of the promoters of important developmental genes. It has been posited that these domains keep these genes inactive, but in a “poised” state for active transcription⁷⁸. Detectable levels of H3K4me3 can also be found in active enhancers⁷⁹. It is important to note that both for H3K4me3 and for subsequent histone modifications, I indicate tendencies for marks to co-localise with particular elements. These tendencies are not exclusive categorisations (e.g. H3K4me3 is typical of active promoters, but can also be found elsewhere). In addition, certain marks can co-localise with genomic elements other than the canonical/established elements. H3K4me3 in particular forms broad non-canonical genomic domains during early development⁸⁰. These findings suggest that potential new biological associations could be found for this classic promoter-associated histone mark, including an association with maternal-to-zygotic transition⁸⁰.

-H3K4me1: the most tissue specific mark among those assayed by the Epigenomics Roadmap Consortium⁸¹. H3K4me1 is predominantly enriched at enhancer regions⁷¹. In a similar way to promoters, a region may be marked as an enhancer, but it may not yet be active (it is a “poised” enhancer). H3K4me1 is a mark that covers both poised and active enhancers. H3K4me1 often precedes nucleosomal depletion, pre-marking potential enhancers prior to activation⁷⁹. H3K3me1 is deposited by MLL3/MLL4⁸². Mechanistically, H3K4me1 binds the

protein acetyltransferase TIP60⁸³, which leads to subsequent steps of enhancer activation.

-H3K27ac: this mark, along with other histone acetylations, is a mark that co-localises with active regulatory elements (i.e. active promoters and enhancers), and has been shown to separate active enhancers from those in a poised state⁸⁴. H3K27ac can be deposited both by p300 and CREB binding protein (CBP)⁸⁵. In the search for histone modifications associated with active regulatory regions, H3K27ac can be complemented by H3K9ac and H3K18ac⁸⁶. Mechanistically, one way for the activation of regulatory elements by H3K27ac is through electrostatic (or *cis*) effects: the negative charge of acetyl groups repels the negative charge of the DNA phosphate backbone, provoking a disassembly of compact chromatin. A second mechanism is formed by the so-called trans effects: protein with domains such as bromodomains (e.g. H3K27ac associated factors p300 and CBP⁸⁷) recognise H3K27ac and recruit other factors that regulate chromatin remodelling (e.g. SWI/SNF⁷²).

-H3K27me3: at the same position, but with a different chemical modification, there is a mark with completely different regulatory associations. Deposited by the polycomb repressive complex 2 (PRC2), this mark is associated with the repression of gene expression⁷¹. Mechanistically, H3K27me3 recruits another polycomb complex (PRC1) that blocks RNA polymerase and causes repression of transcription⁸⁸. H3K27me3 has been linked to the repression of alternative pathways to maintain lineage fidelity during development⁸⁹. It is important to note that, during differentiation, repression is as important as activation, and in

the same way as H3K27ac allows for the development of active regions during cellular differentiation, H3K27me3 represses alternate lineage formation, thus avoiding the activation of unnecessary processes⁸⁹.

-H3K36me3: This mark is associated with actively transcribed regions⁹⁰, and is deposited by HYPB and NSD1⁹⁰. H3K36me3 has been linked to the repression of transcription initiation in gene bodies, dosage compensation and DNA repair⁹¹. This mark is recognised by ZMYND11, which in turn is linked to the control of transcription elongation⁹². In addition, some evidence suggests a role for H3K36me3 in defining exons. Exons are enriched for nucleosomes with H3K36me3⁹³. H3K36me3 is also proposed to affect alternative splicing, with mechanistic evidence for the downstream involvement of MRG15 and PTB in the repression of the silenced exon⁹⁰.

-H3K9me3: This mark is typical of constitutive heterochromatin⁷¹, co-localising with silenced regions that contain repetitive elements, transposons and other repressed elements. One of the readers of H3K9me3 is heterochromatin protein 1 (HP1), a chromodomain protein that extends the formation of inaccessible chromatin⁷⁶. The adequate repression of constitutive heterochromatin is vital, as defects in this mechanism can result in genomic abnormalities, including chromosomal rearrangements⁹⁴.

Knockout mice experiments further highlight the relevance of the aforementioned histone modifications. Many of the related factors are required for embryonic development, including p300⁹⁵ and CBP⁹⁶ (H3K27ac), HP1⁹⁷

(H3K9me3), SET1⁹⁸ (H3K4me3), core components of PRC2, including EZH2 and SUZ12⁹⁹ (H3K27me3), HYPB¹⁰⁰ (H3K36me3) and TIP60 (H3K4me1)¹⁰¹.

1.5 3D genomics

DNA in the eukaryotic nucleus presents a remarkable capacity for folding. Each diploid human cell contains around 2 metres of DNA if the chromosomes are placed end to end, yet the nucleus is only about 6 micrometres in diameter¹⁰². The necessary packaging is achieved by specialised proteins that generate a series of coils and loops, providing increasingly higher levels of organisation.

One of the most studied mechanisms for chromatin folding is mediated through the DNA-binding factor CTCF and the cohesin complex, with CTCF binding to specific sequences in the genome and cohesin providing the structural framework for chromatin loops⁶².

At a macroscopic scale, DNA folding is also remarkably fine-tuned, as is shown by the careful assembly of chromosome structure during mitosis^{102,103}. In addition, at a local level, DNA folding exerts a remarkable influence on the functional mechanisms that control gene expression, as is exemplified by the conformational structures of enhancers, insulators and repressors^{62,104}. This relationship with gene expression is dynamic, as is shown by the dramatic changes in DNA folding during cellular differentiation⁶⁵. The disruption of regions that control 3D genomic structure through genome editing has highlighted the importance of chromosome conformation in essential biological processes⁶⁵.

Initial techniques for the study of chromosome organisation include the use of microscopy to detect specific regions of the nucleus^{105,106}. In the early 1980s, the development of fluorescence in situ hybridisation (FISH) allowed for the study of local DNA folding in finer detail^{107,108}. This technique uses fluorescent probes that only bind to genomic regions which show a high degree of sequence complementarity. Technical advances in the study of DNA folding in recent decades include the development of chromosome conformation capture (3C) techniques^{109,110}. The idea underlying these methods is that DNA contact frequencies can be determined through the quantification of ligation junctions in chromatin that has undergone a process of fixation, digestion and religation¹¹¹. The analysis of these DNA contact frequencies offers insights into chromosome topology¹¹¹. These techniques use a combination of molecular biology methods such as DNA cross-linking, restriction enzyme digestion and DNA ligation, coupled with different product quantitation techniques (array detection, sequencing) to analyse genomic organisation in a cell's natural state. Originally proposed in the year 2002, 3C has paved the way for a family of related techniques with different applications, including 4C, 5C, Hi-C, capture Hi-C, capture-C¹⁰⁴, and ChIA-PET¹¹¹ (**figure 1.14**).

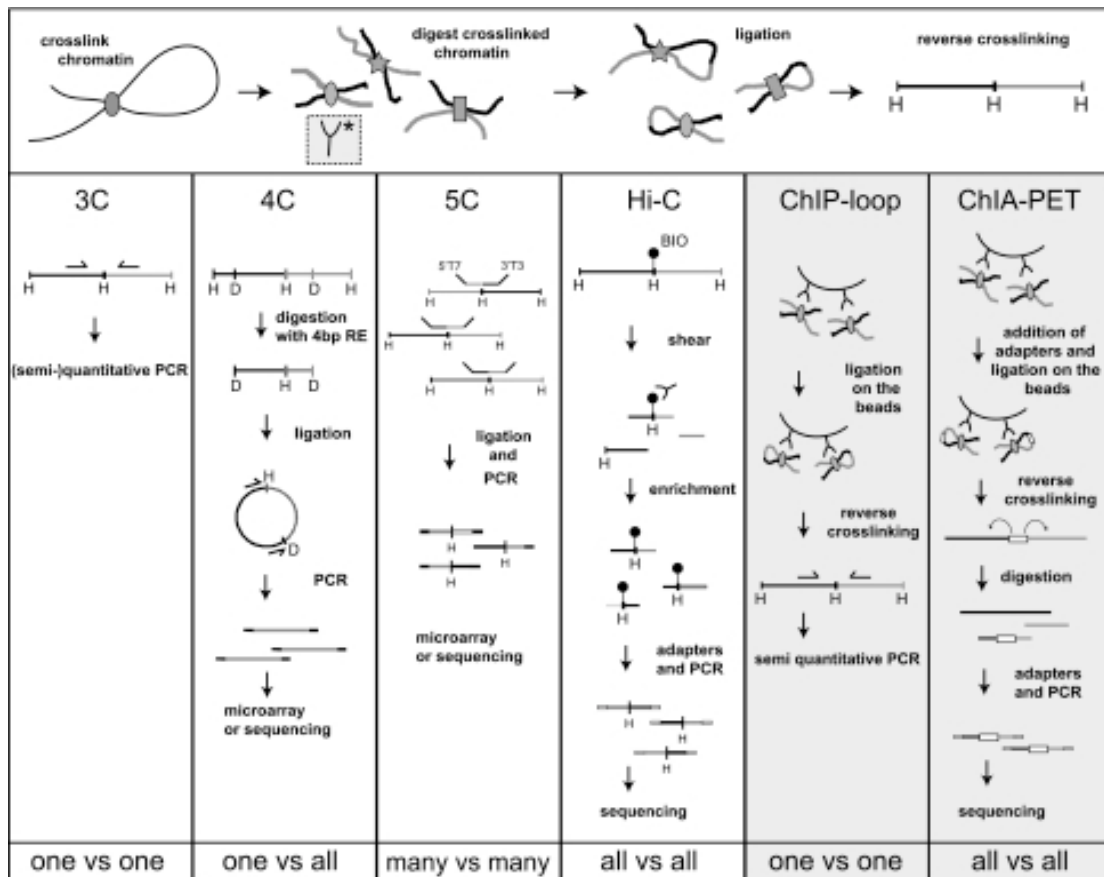


Figure 1.14: Overview of 3C-related technologies. Capture-C and capture Hi-C are not shown as they are more recent than the review that contains this figure^{104,112}. Original figure from de Wit and de Laat, 2012¹¹¹.

Selection of a specific 3C-based technique can depend on the level of throughput required:

-3C (chromosome conformation capture¹⁰⁹): this technique is used to analyse interactions of one genomic region with another region (one vs one interactions). The basic 3C technique has five experimental steps (cross-linking, restriction digest, intramolecular ligation, reverse cross-links and quantitation). Quantitation is usually performed with qPCR or agarose gel detection.

-Capture-C¹⁰⁴: this technique combines 3C library preparation with sonication fragmenting, oligonucleotide capture technology and high-throughput sequencing, allowing for many regions to be assayed in one experiment¹⁰⁴. The sonication step in Capture-C presents an important advantage compared to 4C and 5C as it allows Capture-C to accurately quantify unique ligation junctions within the 3C library (sonication randomly generates unique fragments and overamplified PCR duplicates can be computationally removed)¹⁰⁴.

-4C (circularised chromosome conformation capture¹¹³): this technique has a significant advantage over 3C in that only the sequence of one of the sites of interest needs to be known¹¹³. This fragment is known as the “bait”. 4C studies the interactions of the “bait” region against all other chromosome regions (one vs all). 4C follows the same steps as 3C, with the exception that additional processing is needed before quantification of the fragments of interest. 4C includes the 3C steps of cross-linking, restriction digest, intramolecular ligation, reverse cross-links and then the 4C-specific steps of a second restriction digest, self-circularisation, inverse PCR and quantitation. Quantitation can be performed by sequencing (4C-seq) or by array detection (4C-chip). The inverse PCR step is the step in which primers based on the “bait” sequence amplify circularised molecules in which the unknown captured fragment is also present.

-5C (carbon-copy chromosome conformation capture¹¹⁴): This technique allows for the parallel analysis of interactions between many selected loci (many vs many)¹¹⁴. 5C follows the same first four steps as 3C, but then relies on performing multiplex ligation-mediated amplification (LMA) after cross-links

reversal to generate a library with fragments representative of many genomic interactions. As in 4C, quantitation can be performed by sequencing or by array detection.

-Hi-C¹¹⁵: This technique allows for the parallel analysis of interactions between all genomic loci (all vs all)¹¹⁵. Hi-C follows the same first four steps as 3C, however, in Hi-C, a biotin-labelled nucleotide is incorporated at the ligation junction. This enables selective purification of chimeric DNA ligation junctions as fragments can be pulled down with streptavidin. Quantitation of chromatin interactions is then performed through sequencing. A variant of this technique exists, named Capture Hi-C, that uses specific RNA baits to enrich target loci in a Hi-C library¹¹².

-ChIP-loop (Chromatin Immunoprecipitation-loop¹¹⁶): This method differs from the previous techniques in that a specific protein mediates the interaction between two chromosomal regions. Following the first steps of 3C, after the cross-linking and digestion stages chromatin immunoprecipitation (ChIP) is performed to pull down the protein bound to the site of interest¹¹⁶. After this step normal 3C procedures are conducted (quantitation can be performed by qPCR)¹¹⁶. This technique can assay one vs one interactions.

-ChIA-PET (Chromatin Interaction Analysis by Paired-End Tag Sequencing¹¹⁷): this technique combines ChIP (chromatin immunoprecipitation)-based enrichment, chromatin proximity ligation, paired-end tags and high-throughput sequencing to determine de chromatin interactions on a genome-wide scale¹¹⁷.

After digestion of crosslinked chromatin, the steps of addition of adapters and ligation (of the crosslinked chromatin with these) are performed on antibody-coated beads. This is then followed by reverse crosslinking, restriction enzyme digestion, addition of PCR adapters and PCR. Quantification is performed by sequencing. ChIA-PET is used to assay all vs all interactions.

1.6 Genomics of disease.

Modern medical research into the genetics of disease began with mendelian disorders, such as the study of alkaptonuria by Archibald Garrod¹¹⁸. Insights into the genetics of non-mendelian disorders (such as asthma¹¹⁹, as well as many other autoimmune, neurodegenerative and metabolic diseases) were provided by studies on identical twins, through which it became apparent that these disorders are affected by non-genetic as well as genetic factors, and are thus termed complex diseases. After the completion of the human genome project¹²⁰, SNP arrays were used in large cohorts to investigate the association of common genetic variation and disease in an approach termed genome-wide association studies (GWAS)¹²¹. Despite initial encouraging findings for macular degeneration¹²¹, these efforts did not find common variants with a large effect size for common disorders (with a few exceptions, including ApoE variants and Alzheimer's disease). GWAS, however, did find thousands of replicated SNPs with small effect sizes. While a substantial portion of the genetic contribution to many diseases remains unexplained, efforts in the GWAS field continue to detect new associations, often using large sample sizes¹²².

A typical GWAS involves the comparison of genetic variants between a group of cases and controls for a specific disease. Sample sizes currently reach the hundreds of thousands for certain studies, to increase power in order to detect the small effect sizes observed for many GWAS SNPs. Genotyping of variants is performed using SNP arrays, such as those produced by Illumina or Affymetrix, which can identify millions of variants. Subsequent steps include data normalisation and further bioinformatics processing to exclude any technical variation generated during array or sample processing. Once variants are known for the sample under study a series of statistical approaches, including generalised estimating equations (GEEs) or linear mixed models (LMMs), are applied to identify those variants that are present in significantly higher (or lower) numbers in cases compared to controls. Those variants that pass the genome-wide significance threshold (5×10^{-8} for Bonferroni correction) are reported as associated with the disease under study. These are thus included in the GWAS catalogue¹²³, which of late also includes GWAS summary statistics. For the analysis of GWAS results it is important to consider linkage disequilibrium (LD). It is known that many genomic variants segregate independently from each other through evolutionary history. In these cases the presence of one variant at a given genomic position will not condition the presence of another independent variant at another position. However, when proximal variants form part of the same haplotype the phenomenon of LD will occur. LD is defined as the non-random association of alleles at different loci, that is, the presence of one allele will condition the presence of another¹²⁴. For GWAS results many variants in a haplotype may be significantly associated with a given disease, and it is hard to prove which of these SNPs is the most influential variant for a GWAS-reported

association. It must also be highlighted that GWAS SNP arrays may not cover all the SNPs in a given region, and thus specialised arrays (such as the Immunochip¹²⁵ and the MetaboChip¹²⁶) as well as fine-mapping efforts¹²⁷ are being applied to follow up GWAS findings. In addition, other experimental approaches and bioinformatics efforts may be applied to analyse GWAS data^{128,129}, including approaches that use epigenetic mapping information¹³⁰.

Given the use of a stringent Bonferroni genome-wide significance threshold of 5×10^{-8} , many SNPs that do not reach this threshold may have an influence on the disease under study. Some approaches involve the use of epigenetic data to prioritise candidate regions, which are then validated experimentally¹³¹. These efforts are aiding the detection of causal SNPs below this threshold.

In addition, many diseases and traits are a result of grouping cases belonging to different subtypes. To improve current understanding of the genetics of disease the use of intermediate phenotypes, or endophenotypes, has recently been championed as a way to gain further knowledge on many disease-specific associations¹³². Endophenotypes are defined as intermediate physiological or psychological traits, for example metabolite abundance, a specific neuronal function, or gene expression patterns¹³³.

While significant advances are being made in the understanding of disease-associated protein-coding regions¹³⁴, variation in non-protein-coding regions presents important challenges for interpretation, often requiring multiple techniques for the elucidation of mechanistic impact¹³⁰. These approaches can

include genome editing, 3D genomics and epigenetic mapping of histone marks and DHSs, as in the recent case of the mechanism underlying the obesity-associated variant rs1421085 located in an intron of the FTO gene¹³⁰. For rs1421085, several layers of evidence were used to uncover disease mechanism. Multidimensional epigenetic data analysis highlighted adipose tissue as a potential target tissue for GWAS SNP action¹³⁰. Specifically, rs1421085 was found to be located in an enhancer region in adipose tissue. In addition, a distal interaction between this region and distal gene IRX3 was identified through 4C-seq in human and mouse¹³⁵. Irx3 knockdown in mice led to a reduction in body weight and increased energy dissipation, which were not due to a change in physical activity or appetite¹³⁰. CRISPR/Cas9 genome editing of the rs1421085 risk allele in primary human adipocytes restored IRX3 expression (and expression of proximal gene IRX5), activated adipocyte browning expression programs and restored thermogenesis¹³⁰. These several layers of evidence point to rs1421085 variants underlying a distal enhancer for IRX3/IRX5, which in turn are involved in regulation of adipocyte browning and thermogenesis. This example highlights how multiple layers of evidence will be necessary for uncovering the action of many GWAS SNPs. Specifically, the use of 3D genomics data coupled with multidimensional epigenetic information (e.g. chromatin state, open chromatin and histone mark data) constitutes a powerful tool to aid the identification of non-protein-coding GWAS SNP mechanism. The need for more research in this area is further highlighted by the fact the majority of GWAS disease-associated SNPs locate to non-protein-coding regions¹³⁶.

Bioinformatics provides one way to gain information on the potential function of disease-associated SNPs located in non-protein-coding regions, by integrating SNP information with multidimensional epigenetic mapping data across cell types and tissues. Initial efforts in this direction include Haploreg¹³⁷, RegulomeDB¹³⁸ and FORGE¹³⁹. The concept underlying these approaches is the comparison of overlap counts between the input list and a selected background. Alternative approaches include permutation-based analysis to evaluate significance of overlap, as implemented in GOshifter¹⁴⁰. A more complete list of such methods can be found in Tak and Farnham, 2015¹⁴¹.

1.7 Epigenomics of disease

The study of non-genetic factors in disease presents important confounders. A particular example of this is the study of DNAm variation. Detected DNAm changes between cases and controls can be confounded by cell composition effects and genetic sequence between individuals, as well as by other factors such as age and sex. It is important to consider that DNAm changes can be a consequence of disease mechanisms. EWAS are therefore subject to reverse causation¹⁸. Evidence of causality in EWAS is thus very different to evidence of causality in GWAS (SNPs are not subject to this phenomenon of change in response to environmental influences). Inference methods (such as two step epigenetic mendelian randomisation, that uses an instrumental variable approach with genetic instruments to assess directionality of effect) are required in EWAS to test for evidence of causality¹⁴². In short, the correct design of EWAS is challenging. Despite this, the introduction of epigenome-wide association

studies (EWAS) in 2011¹⁷ has been followed by significant progress, one of the main advances being the development of software for cell composition correction^{48,49,50,51}. Additional approaches to reduce confounding in study design include the use of monozygotic twins to correct for genetic variation, sorted cell types to avoid confounding by cell composition and the annotation of subject information such as sex and age in an effort to correct for these confounders¹⁴³. A large percentage of published EWAS (the number of which exceeds 250 studies⁸) have correlated DNAm levels with case-control status in cross-sectional cohorts. Examples of externally replicated findings include differentially methylated positions (DMPs) for smoking behaviour¹⁴⁴ and Alzheimer's disease¹⁴⁵.

1.8 Thesis aims

Multiple genetic and epigenetic variants have been identified for a range of diseases through GWAS and EWAS. However, many of the identified variants are located in non-protein-coding regions and thus interpretation is challenging. In this thesis I have implemented both bioinformatics and experimental approaches to aid the challenging interpretation of these non-protein-coding variants. Two computational tools (FORGE2 and eFORGE) have been developed for the interpretation of GWAS and EWAS data, respectively. In addition, 4C-seq has been applied to study the three-dimensional context of non-protein-coding variants in mouse neural stem cells (mNSCs) and a range of primary human immune cells.

1.8.1 Design and implementation of FORGE2.

FORGE has provided an automated tool for the analysis of tissue-specific DNase I hotspot enrichment of GWAS SNPs in order to improve understanding of biological context and underlying disease mechanisms¹³⁹. However, a tool that extends the FORGE approach to the wide catalogue of histone mark data has not yet been implemented. My objective was to design and implement FORGE2, an automated tool that analyses the cell type-specific histone mark enrichment of GWAS SNPs. In addition, I aimed to analyse the whole GWAS catalogue in search of novel tissue-disease associations detected by this new tool (chapter 3).

1.8.2 Design and implementation of eFORGE

One way to shed light on the interpretation of non-protein-coding variants is to use epigenetic data, highlighting candidate tissues and cell types for further study. The FORGE tool has automated this analysis approach for GWAS variants. However, to date no such software has been designed for EWAS. My objective was to design and implement eFORGE (experimentally-derived Functional element Overlap analysis of ReGions from EWAS¹⁴⁶), a tool for the analysis of cell type-specific DNase I hotspot (and histone mark) enrichment of EWAS DMPs. Due to the radical differences between GWAS and EWAS, new approaches for background adjustment were necessary. In addition, eFORGE application across a range of different EWAS was required to assess both the potential insights and limitations of the tool (chapter 4).

1.8.3 4C-seq of neurological and autoimmune disorder-associated regions.

Some disease-associated variants act through conformation-dependent distal regulatory interactions, as exemplified in the case of FTO variant rs1421085¹³⁰. Large-scale 3D genomics efforts have been performed to map such interactions across a range of cell types and tissues^{112,147}. However, these efforts typically involve genome-wide approaches such as Hi-C or Capture Hi-C. Such methods lack the power of targeted approaches such as 4C-seq, which map regulatory interactions for specific genomic regions. My objective was to perform 4C-seq across a range of cell types to map the regulatory interactions of selected genetic and epigenetic non-protein-coding variants. Cell types included mNSCs and sorted primary immune cell types (CD14+, CD4+, CD8+, NK and B cells). The high-definition maps generated from this effort are intended to aid interpretation of selected neurological and autoimmune disease-associated variants respectively. Candidate variants were selected by applying FORGE2, eFORGE and additional analyses involving multidimensional epigenetic data. I then performed conformation capture on these candidate variants using 4C-seq to fine-map the 3D context of the genomic regions involved (chapter 5).

2 Materials and Methods

2.1 4C-seq of neurological and autoimmune disorder-associated regions.

2.1.1 Samples

Mouse Neural Stem Cells (mNSCs) were obtained from Suzana Hadjur's lab at the UCL Cancer Institute. Primary human CD14+, CD4+, CD8+, NK and B cells were obtained from the Cambridge Blood Donor Center. The ethical approval for these BLUEPRINT consortium samples was issued under REC reference 12/EE/0040 by the NRES Committee East of England-Hertfordshire. Sample-specific information is described in **table 2.1**.

4C-seq methods shown below are based on protocols which are further detailed in van de Werken et al. (2012)^{148,149}.

Donor Barcode	Gender	Cell Type	Cell count
S01086	M	Monocytes	3.2×10^7
S0104E	F	CD4 cells	9.8×10^7
S0104E	F	CD8 cells	5.5×10^7
S010FT	F	B cells	3×10^7
S010HP	F	NK cells	5.2×10^7

Table 2.1: BLUEPRINT samples. BLUEPRINT consortium sample IDs and gender, cell type and cell count information data for primary patient samples for which 4C-seq was performed.

2.1.2 4C-seq

2.1.2.1 Intranuclear chromatin fixation

Starting with 10 million cells, mNSCs were fixed in media with 1% formaldehyde for 10 minutes in rotation. After glycine quenching, pellets were flash-frozen for storage. BLUEPRINT samples were fixed according to the protocol listed in Javierre et al.¹⁵⁰, as samples from both studies were part of the same preparation pipeline at Mattia Frontini's group in Cambridge. Subsequent steps in 4C library preparation were the same for mNSCs and BLUEPRINT samples unless indicated otherwise.

2.1.2.2 Cell lysis and DpnII digestion

Cells were thawed and lysed for 20 minutes on ice, and were subsequently digested overnight using 750 U DpnII (New England Biolabs, in buffer with 50 mM Bis-Tris-HCl, 100 mM NaCl, 10 mM MgCl₂, 1 mM DTT, pH 6.0). The next day the DpnII buffer and enzyme were replaced and a second overnight digestion performed. On the third day the digestion profile was checked by electrophoresis on an agarose gel.

2.1.2.3 Intranuclear ligation and chromatin decrosslinking

Nuclei were spun down and resuspended in buffer with 1600 U of T4 DNA ligase. Ligation was performed overnight at 16°C without rotating. The next day the ligation profile was checked by electrophoresis on an agarose gel. Proteinase K was added, and samples were incubated overnight at 65°C to reverse crosslinks. The next day samples were treated with RNase A at 37°C for two hours to remove RNA. Phenol/chloroform extraction ensued and DNA was resuspended in TE buffer (ThermoFisher Scientific, 10 mM Tris-HCl, 0.1 mM EDTA, pH 8.0).

2.1.2.4 *Csp6I* digestion

The concentration of purified ligated DNA was measured according to the manufacturer's protocol (Qubit hs DNA assay, Invitrogen), and DNA digestion was performed overnight in a total volume of 300 microlitres using 120 U Csp6I (ThermoFisher scientific, in buffer with 10 mM Tris-HCl, 10 mM MgCl₂, 0.1 mg/mL BSA, pH 7.5). The digestion profile was checked on the next day by electrophoresis on an agarose gel. After confirmation of the correct digestion profile, enzyme was heat-deactivated for 20 minutes at 65°C. DNA was obtained through a second phenol/chloroform extraction and resuspended in nuclease-free water (Ambion).

2.1.2.5 *Intramolecular ligation*

A high-volume ligation reaction was performed using 1600 U of T4 DNA ligase, incubating samples overnight at 16°C without rotation. On the next day a third phenol/chloroform extraction was performed, and samples were resuspended in TE buffer.

2.1.2.6 *Target selection*

GWAS and EWAS regions were prioritised for 4C-seq through multidimensional epigenetic data analysis incorporating several bioinformatics tools. For this analysis data was included for DNase I hotspots, expression Quantitative Trait Loci (eQTL), FANTOM5 enhancer annotations, promoter Capture Hi-C contacts, genomic context and disease association (see **figures 2.1** and **2.2**). Data underlying the selection of each region is shown in **table 2.2** for mNSC loci and **table 2.3** and **figure 2.3** for immune cell loci.

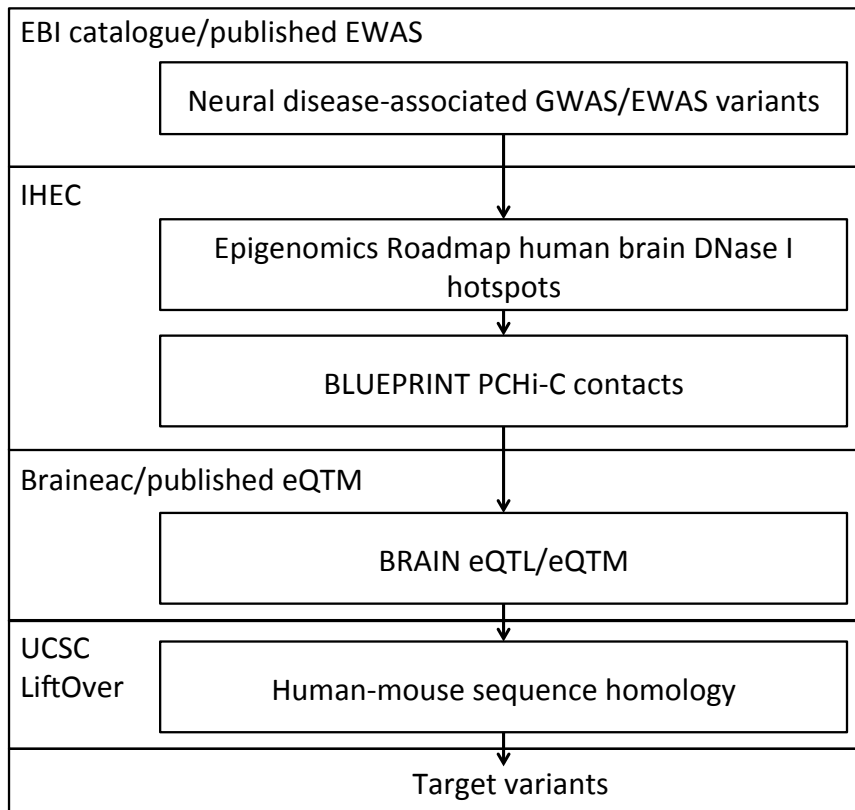


Figure 2.1: Process for selecting neurological disorder-associated variants.

GWAS and EWAS variants were selected depending on several factors including DNase I hypersensitivity in human brain, previous contacts in PCHi-C, eQTL/eQTM status in brain (eQTM data from Gutierrez-Arcelus et al., 2013¹⁵¹) and hg19-mm10 sequence homology.

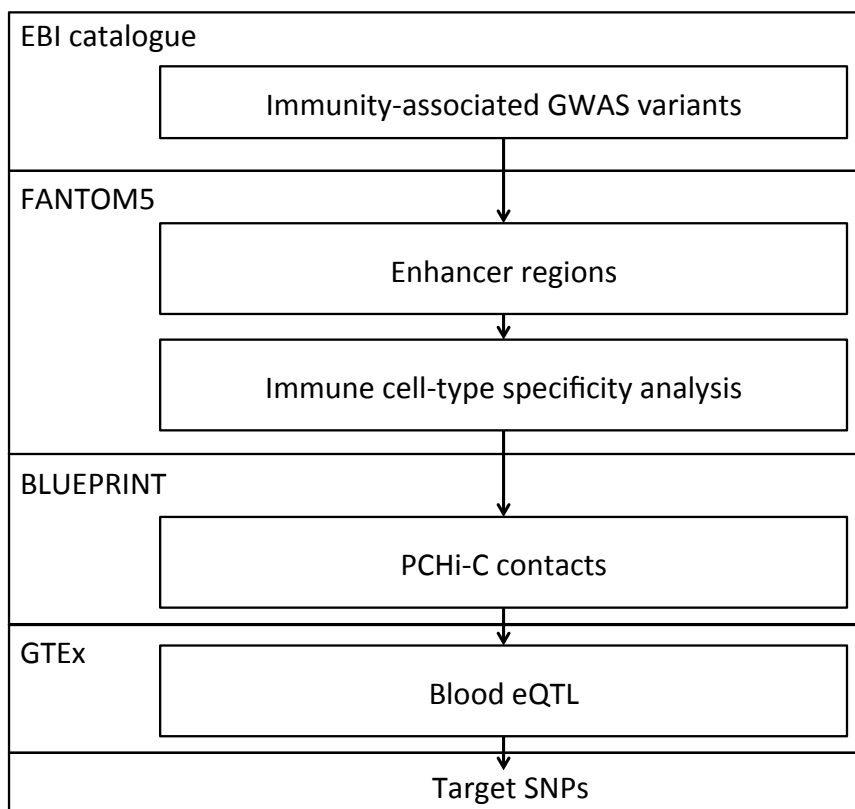


Figure 2.2: Process for selecting immunity-associated loci. GWAS variants were selected depending on several factors including FANTOM5 enhancer status in blood, FANTOM5 cell-type enhancer specificity, previous contacts in PCHi-C and eQTL status in blood.

ID	Previous PCHI-C data	eQTL/eQTM and target gene	Genomic context in human	Genomic context in mouse	Disease association
rs7618915	No	ITIH4 (brain).	1kb away from the promoter of PPM1M	1kb away from the promoter of Ppm1m	Bipolar disorder-associated SNP
rs8012	No	GCDH (brain)	Last exon of GCDH	Last exon of Gcdh	Blood metabolite level-associated SNP
rs1009014	Yes (blood)	SYNJ2 (cerebellum)	In an intron of SYNJ2	In an intron of Synj2	None, meQTL for SYNJ2 (Marzi et al., 2016).
rs548181	Yes (blood)	CCDC15 CHEK1 DDX25, FEZ1 HEPACAM, HEPN1, PUS3 ROBO3 SRPR STT3A TBRG1	In an intron of STT3A antisense RNA 1 (STT3A-AS1).	In an intron of AK018988 (potential homologue to STT3A-AS1).	Several (including bipolar disorder and schizophrenia).
cg05066959	Yes (blood)	Not known if eQTM	In an intron of ANK1 and NKX6-3	In an intron of Ank1, but not Nkx6-3.	Alzheimer's disease-associated DMP
cg02672452	No.	Not known if eQTM	In an intron of non-coding RNA LOC654342.	Between Otop1 and Drd5	Alzheimer's disease-associated DMP

Table 2.2: Mouse target loci. Multidimensional epigenetic analysis results for prioritised neurological disease associated-GWAS/EWAS regions including genomic context, eQTL status and previous promoter capture Hi-C contacts (using the CHiCP tool¹⁵², <https://www.chicp.org/>). Disease association category is also shown. One non-GWAS SNP was included due to evidence suggesting interaction between genetics and epigenetics, specifically rs1009014 was found to determine allele-specific DNAm in cerebellum in a recent study by Marzi et al.¹⁵³. eQTL data were obtained from Braineac and GTEx databases^{154,155}. All these loci were found to overlap DNase I hypersensitive sites from either Epigenomics Roadmap or ENCODE consortia using the FORGE and eFORGE tools^{139,146}

Enhancer position (hg19)	Cell specificity (FANTOM5 enhancer data)	ChIP in CD34	eQTL (Walsh et al.)	GTEx Whole Blood	Gene	GTEx Spleen	Gene	Genomic context	Immune disease/trait association by GWAS
chr1:90022562-90022945	Monocyte	LRRC8D	LRRC8B	rs4658279	LRRC8B	rs4658279	LRRC8B	Intron of LRRC8B	Neutropenia/leucopenia after chemotherapy
chr17:37912171-37912498	B cell, CD8+ cell, NK cell	None	GSDMB, ORMDL3, PGAP3	rs77013147	GSDMA	rs12946510	GSDMB, ORMDL3	Between GRB7 and IKZF3	Ulcerative colitis, Crohn's disease
chr22:37258335-37258758	Monocyte	NCF4 and PVALB	PVALB	rs4821544	PVALB	None	None	Intron of NCF4	Atopic dermatitis
chr5:131430055-131430830	Monocyte and NK cell	IL3 and P4HA2-AS1	ACSL6	rs657075	AC034228.7	None	None	Between CSF2 and P4HA2	Rheumatoid arthritis
chr6:167534229-167534357	B cell	CCR6 and RNASET2	RNASET2	rs3093023	RNASET2	None	None	Intron of CCR6	Rheumatoid arthritis
chr9:123652531-123653088	None	TRAF1, PSMD5/P SMD5-AS1	PSMD5-AS1, TRAF1, C5	rs881375	PSMD5-AS1	rs881375	PSMD5-AS1	Between PHF19, TRAF1	Rheumatoid arthritis

Table 2.3: Human target loci. Multidimensional epigenetic analysis results for prioritised human immune cell regions including cell type-specific enhancer status from FANTOM5, previous promoter capture Hi-C contacts (using the CHiCP tool¹⁵², <https://www.chicp.org/>), eQTL status from a study by Walsh et al. (2016)¹⁵⁶ and the GTEx consortium (in whole blood and spleen)¹⁵⁵, genomic context and immune disease/trait GWAS association. All these loci were found to overlap DNase I hypersensitive sites from either Epigenomics Roadmap or ENCODE consortia using the FORGE tool¹³⁹. All of these loci were also found to overlap H3K4me1 broadPeaks from the Epigenomics Roadmap consortium using the FORGE2 tool (**figure 2.3**), which I developed as part of this doctoral project.

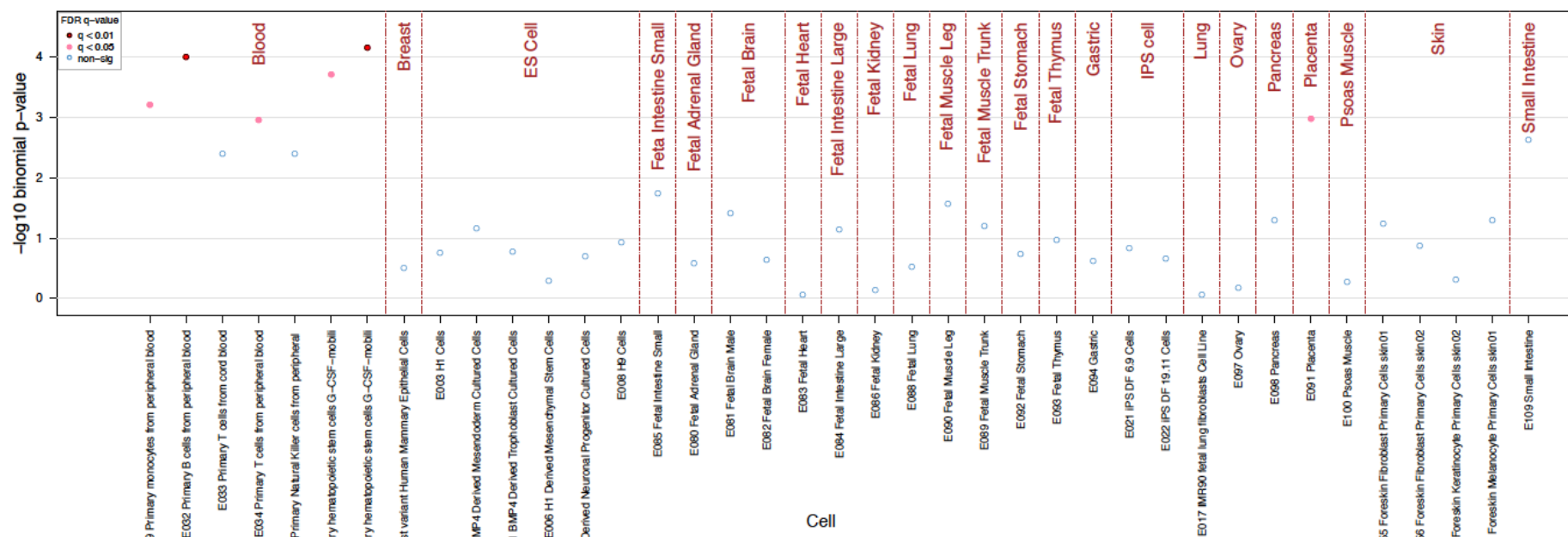


Figure 2.3: FORGE2 H3K4me1 analysis of human target loci. Analysis was performed on GTEx Whole Blood eQTL SNPs prioritised for 4C-seq analysis in BLUEPRINT consortium primary samples. H3K4me1 is a mark enriched for enhancer regions, and even from this small set of SNPs a clear immune cell enrichment signature can be noted. SNPs rs4658279, rs77013147, rs4821544, rs657075, rs3093023 and rs881375 were used for analysis. All these SNPs were found to overlap H3K4me1 broadPeaks in B cells, T cells and haematopoietic stem cells.

2.1.2.7 Library amplification

DNA concentration was measured according to the manufacturer's protocol (Qubit hs DNA assay, Invitrogen). I ordered oligonucleotides containing pre-designed PCR primers from van de Werken et al. (2012)¹⁴⁹ (http://compgenomics.weizmann.ac.il/tanay/?page_id=367) for target regions in mNSCs (**table 2.4**, *Appendices table S1*), and sorted blood cells (*Appendices table S2*). PCR reactions were set using Expand polymerase (Roche) and 100 ng of 4C template. PCR conditions were 94°C 3 minutes, (94°C 10 seconds, 55°C 1 minute, 68°C 3 minutes, 30 cycles) and 68°C 10 minutes. PCR product was purified according to the manufacturer's instructions (Roche High Pure PCR product purification kit) to remove primer dimers. 4C libraries were then concentrated down to 15 microlitres using Ampure XP beads according to the manufacturer's instructions. PCR amplification was checked on the Bioanalyser using the HS DNA kit (Agilent).

chr (hg19)	start (hg19)	end (hg19)	chr (mm10)	start (mm10)	end (mm10)	ID
chr3	52279344	52279844	chr9	106199189	106199646	rs7618915
chr19	13010270	13010770	chr8	84886598	84887064	rs8012
chr6	158486893	158487393	chr17	6012819	6013423	rs1009014
chr11	125461459	125461959	chr9	36768370	36768857	rs548181
chr8	41519308	41519309	chr8	23141313	23141314	cg05066959
chr2	91818189	91818190	chr5	38308967	38308968	cg02672452

Table 2.4: Human-mouse homologous regions. List of prioritised human GWAS/EWAS regions and corresponding homologous mouse loci identified for 4C-seq analysis. GWAS RSIDs and EWAS 450k array cg probe numbers are also

included. Human regions were lifted over to the mouse genome using UCSC LiftOver (minimum ratio of remapping bases set to 0.95)¹⁵⁷.

2.1.2.8 4C library sequencing

Libraries were combined into a single vial and the mix was quantified using the Qubit method (Qubit hs DNA assay, Invitrogen). The theoretical molarity of the mix was calculated using the formula:

$$[\text{Library mix [c] microgram/microlitre : (650x600)}] \times 10^9 = \text{nM library mix}$$

The molarity was adjusted to 2nM, and the DNA was then denatured for 5 minutes by adding 10 microlitres of 0.2 N NaOH, and taken for sequencing at a final molarity of 10pM.

Sequencing for mouse regions homologous to human GWAS/EWAS regions was performed on a MiSeq using a v2 (300 cycle) single-end sequencing kit. The sequencing process was subsequently validated using a v3 (150 cycle) single-end MiSeq kit. Finally, for the multiplexed regions across the BLUEPRINT primary samples (including CD14+, CD4+, CD8+, B cell and NK cells) a HiSeq run was performed using the following kits:

- One unit of HiSeq Rapid SBS Kit v2 (50 cycles): 50 cycle SBS kit for HiSeq 2500.
- Two units of HiSeq Rapid PE Cluster Kit v2: Paired-end cluster generation kit for HiSeq 2500.

2.1.2.9 4C bioinformatics analysis

I applied the 4Cseqpipe software described in van de Werken et al. (2012)¹⁴⁹ for analysis of 4C-seq data. 4Cseqpipe was downloaded from <http://compgenomics.weizmann.ac.il/tanay/>. This pipeline maps 4C-seq libraries to the genome (hg19/mm10), performs data normalisation and subsequently generates near-cis domainograms to visualise 3D-genomic contacts. For analyses I used 4Cseqpipe version 0.7 on the UCL computer science department cluster with trimming settings set to 50 bp.

2.2 eFORGE methods

2.2.1 Introduction

eFORGE (experimentally-derived Functional element Overlap analysis of ReGions from EWAS) is a computational tool that analyses whether there is cell type-specific enrichment of overlap for DMPs from a given EWAS compared to matched background sites from the EWAS array.

In short, eFORGE analyses the cell type-specific regulatory component of a set of EWAS DMPs. Enrichment is computed on a per cell type basis, since hotspots and histone mark broadPeaks vary between different cell types, and hence can expose cell type-specific signals of enrichment for the given test DMP set. This can reveal the regulatory sites underlying the EWAS signal, and, in addition, detect cell composition effects.

The functional elements considered in eFORGE are DNase I hotspots from the ENCODE, Epigenomics Roadmap and BLUEPRINT consortia, and histone mark broadPeaks from the Epigenomics Roadmap consortium. Hotspots (areas of enriched DNase I cleavage) were generated using the HOTSPOT program, and histone mark broadPeaks were generated by the MACS2 method. BroadPeaks correspond to broader domains of enrichment for a given histone mark, as opposed to narrowPeaks which are less broad, contiguous regions of enrichment. In a similar way, hotspots correspond to wider domains of DNase I hypersensitivity when compared to DNase I peaks, which are 150 basepair regions with a very high DNase I density relative to background. Hotspots have been shown to reveal more tissue-specific enrichment signal than peaks¹³⁹.

2.2.2 Sample processing

DNase I hotspots were obtained from three consortia: ENCODE, Roadmap Epigenomics and BLUEPRINT. ENCODE data (Thurman et al., 2012) were taken from the ENCODE European Bioinformatics Institute (EBI) ftp site (all URLs for sample processing listed in **table 2.5**). BLUEPRINT data were downloaded from the Genomatix interface. For Roadmap Epigenomics 2012 data DNase I sequencing tag alignments were downloaded from the Genboree webpage. Only files within the data use embargo agreement were used, these files belong to the section of Gene Expression Omnibus (GEO) accession number GSE18927. The alignments were processed using the Hotspot method^{158,159}, applying the default parameters. For consolidated Roadmap Epigenomics 2015 data, DNase I

hotspots and BroadPeak Histone mark files were downloaded from the Roadmap Epigenomics 2015 webpage. Cell and tissue names were assigned to each dataset using consortium data or custom perl scripts from the decodings that can be obtained from the ENCODE Data Coordination Centre tables or sample group SAMEG31306 from the BioSamples database.

Dataset	URL
ENCODE (EBI ftp site)	ftp://ftp.ebi.ac.uk/pub/databases/ensembl/encode/integration_data_jan2011/byDataType/openchrom/jan2011/combined_hotspots/
BLUEPRINT (Genomatix)	https://blueprint.genomatix.de/grid/experiments/browse
Roadmap 2012 (Genboree)	http://www.genboree.org/EdaccData/Current-Release/experiment-sample/Chromatin_Accessibility/ .
Hotspot method	http://www.uwencode.org/proj/hotspot/
Roadmap Epigenomics 2015	http://egg2.wustl.edu/roadmap/data/byFileType/peaks/consolidated/broadPeak/
ENCODE Data Coordination Centre	https://genome.ucsc.edu/encode/cellTypes.html
BioSamples	http://www.ebi.ac.uk/biosamples/

Table 2.5: eFORGE sample processing. URLs for eFORGE sample annotation and data processing.

2.2.3 Preparation of sample overlaps

450k array probes were overlapped with DNase I hotspots and histone mark broadPeaks using the *bedtools* tool¹⁶⁰. Overlaps were stored in an SQLite database (<http://www.sqlite.org>) that was organised by datasets. Data was coded in bitstrings: for each probe I generated a binary string. In this string 1 represented the presence of an overlap with a hotspot in each dataset (either Roadmap, BLUEPRINT or ENCODE). For each string 0 represented the absence of such an overlap. For 27k analysis the 94% of overlapping probes between both arrays were used. For location information I used the HumanMethylation450 v1.2 manifest file (https://support.illumina.com/array/array_kits/infinium_humanmethylation450_beadchip_kit/downloads.html).

2.2.4 Analysis strategy

eFORGE was designed in order to analyse cell type-specific enrichment signal for a list of 450k array probes. This can be performed by the following data processing strategy, as shown below in **figure 2.4**:

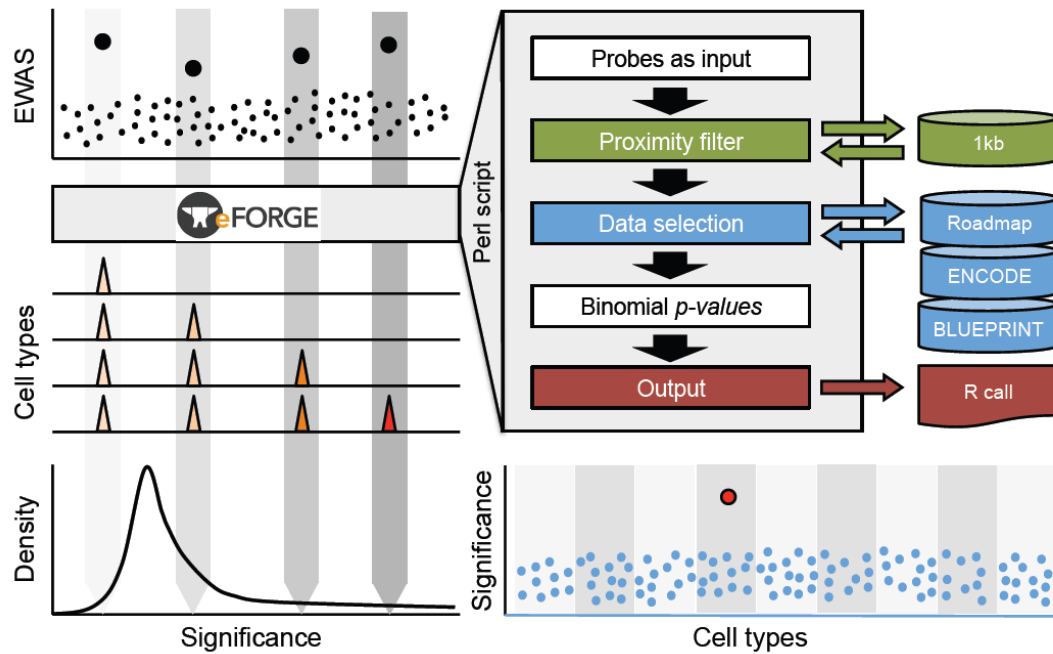


Figure 2.4: eFORGE analysis strategy. A list of top probes from an EWAS is input into eFORGE. The software then creates 1000 lists of random probes from an array of choice (27k or 450k). Background probes are selected for similar annotations to the input probes regarding gene relationship and CpG island relationship. The overlap of the 1000 random lists with DNase I hotspot data across cell types is used to create a background distribution. The overlap of the input list is then tested for enrichment against this background through a binomial test. The $-\log_{10}$ (p-value) of the EWAS list compared to the background lists is represented to visualise enrichment. Figure obtained from Breeze et al. (2016)¹⁴⁶.

The eFORGE analysis strategy evaluates the input probe list by comparison with 1000 randomly selected background probe lists. eFORGE selects similar background probes to the input list through annotation-based matching (e.g. if one probe from the input list is located in a promoter, then its matching probe in

each of the 1000 background lists is also in a promoter). DNase I hotspot enrichment for a particular tissue is computed by comparing the DNase I hotspot overlap of the input list to the DNase I hotspot overlap of the 1000 background lists. This process is also applied in the analysis of histone mark broadPeaks.

2.2.5 Annotation categories

Two main annotation levels were employed for classing probes: “gene annotation” (including categories 1stExon, 3'UTR, 5'UTR, Body, IGR, TSS1500, TSS200) and CpG island annotation (with categories Island, Shore_Shelf, NA or “open sea”). Probes were then classed into 21 bin categories, which result from all possible combinations of these two levels of annotation. The “Shelf” and “Shore” categories (from CpG island annotation) were merged in order to ensure that each bin category contained more than 1000 probes. This is important given that small bin sizes could introduce bias as background selection would be dependent on the overlaps of a few probe lists. The aforementioned 450k array Illumina annotation HumanMethylation450 v1.2 manifest file (https://support.illumina.com/array/array_kits/infinium_humanmethylation450_beadchip_kit/downloads.html) was used to create these data categories (tables 2.6 and 2.7).

		CpG island annotation		
		Island	Shore_Shelf	NA
Gene annotation	Intergenic region (IGR)	Island IGR	Shore_Shelf IGR	NA IGR
	Within the 200 bp prior to a transcription start site (TSS200)	Island TSS200	Shore_Shelf TSS200	NA TSS200
	1st Exon	Island 1st Exon	Shore_Shelf 1st Exon	NA 1st Exon
	Within the 1500 bp prior to a transcription start site (TSS1500)	Island TSS1500	Shore_Shelf TSS1500	NA TSS1500
	3' untranslated region (3' UTR)	Island 3' UTR	Shore_Shelf 3' UTR	NA 3' UTR
	5' untranslated region (5' UTR)	Island 5' UTR	Shore_Shelf 5' UTR	NA 5' UTR
	Gene Body	Island gene body	Shore_Shelf gene body	NA gene body

Table 2.6: eFORGE background probe matching. Background probe matching in eFORGE is performed using two levels of data categories. The first level is based on gene-centric annotation and the second level corresponds to CpG island annotation.

Gene annotation	CpG island annotation	Probe number
1stExon	Island	15581
1stExon	NA	4282
1stExon	Shelf_Shore	2874
3'UTR	Island	1992
3'UTR	NA	10274
3'UTR	Shelf_Shore	5228
5'UTR	Island	17581
5'UTR	NA	11855
5'UTR	Shelf_Shore	13249
Body	Island	38102
Body	NA	68162
Body	Shelf_Shore	55413
IGR	Island	22392
IGR	NA	57749
IGR	Shelf_Shore	39511
TSS1500	Island	21610
TSS1500	NA	14667
TSS1500	Shelf_Shore	32707
TSS200	Island	32996
TSS200	NA	9058
TSS200	Shelf_Shore	10229
	Total	485512

Table 2.7: eFORGE background bin subcategories. eFORGE background bin subcategories are listed corresponding to the two levels of annotation. Numbers of probes within each category are indicated.

2.2.6 Proximity filtering

One of the other issues in FORGE is the problem of analysing SNPs that are in close proximity, which could mean testing variants with strong Linkage Disequilibrium (LD) against the same DNase I hotspot. This is a problem as it

could result in higher tissue-specific enrichment signal than is real (due to the LD several variants may be associated with a given trait or disease when only one SNP underlies disease mechanism). Therefore FORGE includes an LD filter in its code, and all SNPs in strong LD ($r^2 > 0.8$) are filtered to leave one SNP for each haplotype. For eFORGE there is a similar problem with DNAm correlation in proximal regions. This correlation means that I am facing a similar problem to the LD problem that FORGE faces. Therefore a proximity filter of 1kb was included in eFORGE code, which, after selecting one EWAS probe, cannot select another probe within 1kb of that probe (500 bp upstream and downstream). The choice of selecting 1kb as a limit for filtering was based on previous studies showing strong correlation of DNA methylation levels between CpGs fewer than 1kb apart^{38,161}.

2.2.7 eFORGE code structure

The eFORGE tool consists of a series of Perl scripts that use SQLite databases and make R calls for plotting. For the development of the tool I had to first learn Perl and study the SQLite database call nomenclature.

Figure 2.5 displays a schematic of the files included in eFORGE, and **figure 2.6** shows a schematic of the code calls performed by the tool.

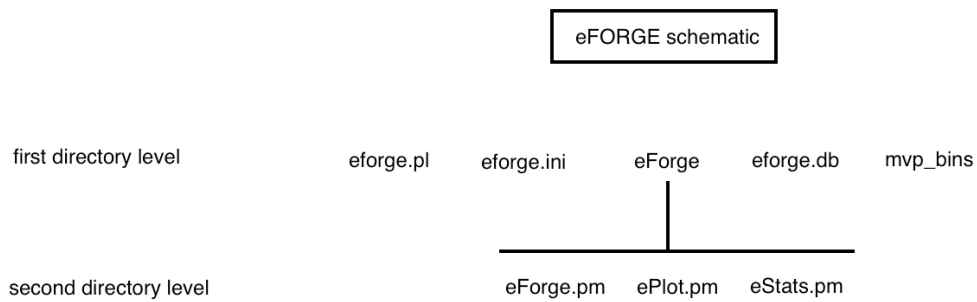


Figure 2.5: eFORGE directory schematic. This schematic shows the directory structure required for an eFORGE installation. `eforge.pl` is the main script, which also lists the eFORGE documentation (contained in a Perl Pod). `eforge.ini` contains a directory address which is used by the main script (some eFORGE files can be placed in another directory if `eforge.ini` is modified). The `eForge` folder contains the Perl modules `eForge.pm`, `ePlot.pm` and `eStats.pm`, which contain the functions accessed by the main script. `eforge.db` contains the SQLite database and `mvp_bins` contains the Perl hash required for background selection of similar probes (according to the two levels of annotation) and proximity filtering.

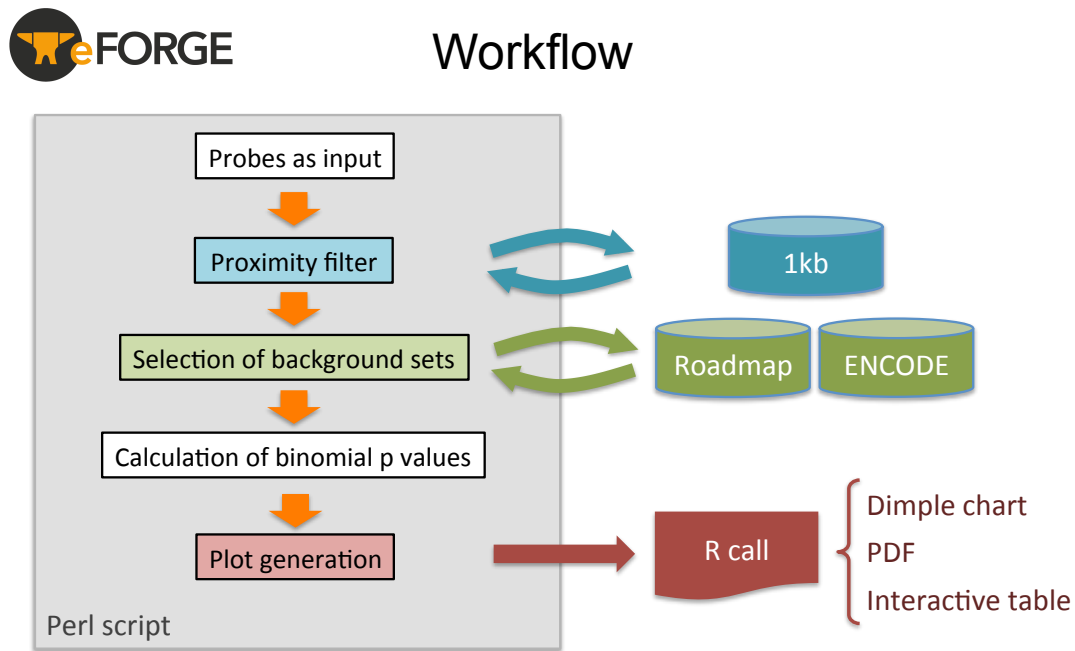


Figure 2.6: eFORGE code schematic. This diagram shows the code calls made by eFORGE. A Perl script performs the main processing of the data (supported by several Perl modules), and external calls are made in SQLite (green) for database requests and R (red) for plotting.

2.2.8 eFORGE input

DMPs can be input into eFORGE in two formats: Illumina 450k/27k probe IDs, or BED format (0 based BED format, with chromosome location input as chrN, corresponding to the human genome assembly GRCh37 genomic coordinates). eFORGE can retrieve probe IDs if genome location is input in this format. For analysis I suggest a maximum probe list size of 1000 and a minimum of 20 probes. Any probes that are not present on the 450k array (or on the 450k array-27k array shared set) are excluded from the analysis.

2.2.9 eFORGE outputs

eFORGE generates tables and graphic descriptions of the overlap enrichment for the test DMPs for each DNase I hotspot sample. A tab-separated values (TSV) file is output, including the following columns: z-score, p-value, cell, tissue, hotspot sample filename, overlapping DMPs, sample GEO accession number, and BY adjusted q-value. The Datatables (<https://datatables.net/>) plug-in for the jQuery Javascript library is used to generate an interactive table containing these data. Datatables is accessed through the rCharts package (<http://ramnathv.github.io/rCharts/>).

For eFORGE chart generation, each of the graphical outputs presents the $-\log_{10}$ (binomial p-value) by cell sample. PDF graphics are generated using base R graphics (<http://www.r-project.org>). The interactive JavaScript graphic is generated using the rCharts package (<http://ramnathv.github.io/rCharts/>), which interfaces with the dimple d3 libraries (<http://dimplejs.org>). Cells are grouped alphabetically within each tissue in the PDF and the interactive graphic (tissues also follow alphabetical order). For the interactive graphic replicate samples are stacked at the same X coordinate. In addition, blue (q-value > 0.05), pink (q-value < 0.05), and red (q-value < 0.01) are consistently used to colour BY-corrected q-value results in each of the graphics.

2.2.10 Multiple testing correction and false positive rates

One of the main differences in multiple testing correction between eFORGE and FORGE is that FORGE uses a Bonferroni correction, therefore dividing the p-values by the number of different tissues (and not cell types) present within the dataset. This does not take into account DNase I hotspot correlation both within a tissue and between tissues, and does not permit statements about the enrichment of a specific cell type to be made, as correction is tissue-based and not cell type-based.

For eFORGE I consider that it is important to be able to make statements about the enrichment of a specific cell type, and therefore I require a cell type-level correction that does not assume independence between cell types (as there is a correlation in DNase I hotspot/histone mark profiles that occurs both within a tissue and between tissues). Approaches to control for multiple testing in this context are found in the False Discovery Rate (FDR) approach. The FDR approach does not focus on keeping an alpha level of significance of 0.05 in each test but rather on controlling the level of false positives below a certain threshold. Within the FDR approach the appropriate multiple testing correction in this case is a Benjamini-Yekutieli (BY) correction, which controls the false discovery rate under positive dependence assumptions¹⁶². BY was compared with the FDR-based Benjamini-Hochberg correction (which is more appropriate for independent tests). For this comparison 5000 probe sets were tested containing between 5 and 100 DMPs on eFORGE. This analysis was performed using both BH and BY for multiple testing correction. Given that each of the 5000

DMP sets contained 299 samples (Roadmap 2012 data), the total number of sample-tests is 1,495,000. For all these tests I observed 1 false positive for BH, and 0 false positives for BY, for a significance level of $q\text{-value} < 0.01$. False positives at a $q\text{-value} < 0.05$ were estimated to be around 0.36% for 5000 tests applied across ENCODE and Roadmap (BY correction). However, for analyses with eFORGE I consider a $q\text{-value} < 0.05$ as only an intermediate level of significance. In addition, these false positive rates do not show a clear tendency with the size of the probe sets (**tables 2.8** and **2.9**). In regard to the variation in numbers for false positives (at a level $q < 0.05$) in both tables, further extensive testing (2,990,000 sample tests, Epigenomics Roadmap data) revealed more homogeneous results (1598 ± 54 sample-test false positives at $q < 0.05$ for set sizes 10-100). This suggests that the aforementioned variation is due to the number of tests performed. Furthermore, these results suggest that most false positives at $q\text{-value} < 0.05$ come from a low number of random probe lists that present borderline enrichment for many tissues/cell lines. Taken together, these results suggest that it is highly improbable to get cell type-specific enrichment with random probe lists in eFORGE.

SetSize	N	p<0.05	p<0.01	q<0.05	q<0.01	F	p<0.05	p<0.01	q<0.05	q<0.01
5	625000	2418	190	0	0	5000	936	104	0	0
10	625000	23686	3602	23	0	5000	2829	943	11	0
15	625000	5721	597	0	0	5000	1338	283	0	0
20	625000	9313	1226	32	0	5000	1887	426	3	0
30	625000	13487	1591	4	0	5000	2274	542	1	0
40	625000	10877	1356	6	0	5000	1999	475	5	0
50	625000	13714	1554	16	0	5000	2222	570	4	0
100	625000	12973	1593	1	0	5000	2229	554	1	0

Table 2.8: eFORGE ENCODE false positives. False positive results after testing 5000 random probe sets on ENCODE data. N indicates number of sample tests. F indicates number of tool tests.

SetSize	N	p<0.05	p<0.01	q<0.05	q<0.01	F	p<0.05	p<0.01	q<0.05	q<0.01
5	1495000	4161	313	0	0	5000	1013	142	0	0
10	1495000	54869	7930	264	0	5000	2998	1065	6	0
15	1495000	10519	904	0	0	5000	1544	316	0	0
20	1495000	16263	1570	1	0	5000	1894	431	1	0
30	1495000	28526	3279	0	0	5000	2214	602	0	0
40	1495000	22855	2557	129	0	5000	1997	496	2	0
50	1495000	30460	3430	1	0	5000	2341	672	1	0
100	1495000	23173	2480	1	0	5000	2051	510	1	0

Table 2.9: eFORGE Epigenomics Roadmap false positives. False positive results after testing 5000 random probe sets on Epigenomics Roadmap data. N indicates number of sample-tests. F indicates number of tool tests.

2.2.11 Database extension and generation

Given the need for continuity and tool extensibility in bioinformatics, a procedure was designed for generating the eFORGE database from consortium data. All the code to do this is openly available on GitHub (at <https://github.com/charlesbreeze/eFORGE/blob/master/database/README.txt> with examples given). To extend the eFORGE database, bitstrings from a new consortium or project can be added as new tables to the SQLite database, which, in contrast with the FORGE database, is modular. Indications are also provided for the production of bitstring tables from raw data from a new consortium.

2.2.12 tDMP and cDMP analysis

Analysed tissue-specific Differentially Methylated Positions (tDMPs) were provided by Queen Mary University of London (QMUL) researcher Dr Robert

Lowe, and were obtained through the following procedure. Normalised data were downloaded for tissues with at least 50 samples in Marmal-aid¹⁶³. Tissue-specific DNAm differences were initially called using the dmpFinder function from the minfi package¹⁶⁴. In order to do this two categories were defined: data for the tissue of interest were contained in group 1, and data for all other tissues were contained in group 2. In order to visually inspect the data the profiles of the top 2 probes were observed for each of the samples. If a sample was found to be closer to the mean of group 2 than group 1 the sample was removed. From the remaining data 50 samples for each tissue were randomly selected. Differences were then called using dmpFinder as described above. Analysed sample IDs are listed in **table S3** (see *Appendices*).

For cell type-specific Differentially Methylated Position (cDMP) obtention I used data from Jaffe and Irizarry, 2014¹⁶⁵. In order to define a cDMP I used two criteria: first the DNAm beta value for a probe in the cell type of interest should be lower than that of other cell types, second, the difference between this DNA methylation value and the next closest DNA methylation value should be greater than 0.4.

2.2.13 Source code

eFORGE source code can be downloaded from GitHub at <https://github.com/charlesbreeze/eFORGE>. eFORGE installation and analysis have been performed on Mac OSX 10.8.4, 10.10.1 and Red Hat Linux. For eFORGE installation download of the eforge.db database and the background selection

hash table is required, and this can be done from <http://eforge.cs.ucl.ac.uk/?download>. eFORGE is also available as a web tool (<http://eforge.cs.ucl.ac.uk/>). The web page includes documentation on eFORGE code and the different data analysis options available.

2.3 FORGE2 methods

2.3.1 Introduction

The FORGE2 (Functional element Overlap analysis of Regions from GWAS Experiments 2) tool performs a Functional Overlap analysis on histone mark broadPeaks to identify cell type-specific signal for a given set of GWAS SNPs.

The main aim of FORGE2 is to analyse the cell type-specific regulatory component of a set of GWAS SNPs. FORGE2 takes a set of SNPs, such as those variants from a GWAS that are above the genome-wide significance threshold, and detects whether there is enrichment of overlap in regulatory elements compared to matched background SNPs from a similar array technology. This analysis is repeated for each cell type, since histone mark broadPeaks vary between different cell types. This cell type-specificity of epigenetic marks can reveal associations between cell types and the test SNP set through FORGE2 enrichment patterns. This enrichment can thus uncover the regulatory sites underlying GWAS signal. It may also help to confirm GWAS findings where a cell type-specific mechanism is projected or known for a specific phenotype. In

addition, novel cell type involvements that were previously unknown can also be revealed.

Epigenetic data considered for the initial implementation of FORGE2 include histone mark broadPeaks from the Epigenomics Roadmap project generated by the MACS2 method. BroadPeaks correspond to broader domains of enrichment for a given histone mark, as opposed to narrowPeaks which are less broad, contiguous regions of enrichment. Previous functional overlap analysis tools such as FORGE¹³⁹ have focused on other epigenetic datasets, such as DNase I hotspots. To further explore the associations obtained through FORGE hotspot analyses it is necessary to separate the different classes of genomic elements driving the observed GWAS-hotspot enrichment (e.g. promoters, enhancers). One way to do this is by analysing histone mark data. I have developed FORGE2 to explore this wide and rich dataset. Thus, in addition to computational and algorithmic advances when compared to FORGE, FORGE2 extends functional overlap analyses to histone mark data.

2.3.2 Sample processing

I obtained histone mark broadPeak files for H3K4me1, H3K4me3, H3K27me3, H3K9me3 and H3K36me3 from <http://egg2.wustl.edu/roadmap/data/byFileType/peaks/consolidated/broadPeak/>. The Epigenomics Roadmap consortium had previously generated all files using MACS2. Cell and Tissue assignments were obtained using labels from the Epigenomics Roadmap consortium.

2.3.3 Preparation of sample overlaps

Histone mark broadPeak regions in BED format were overlapped with SNP positions using *bedtools*¹⁶⁰, resulting in a substantial improvement in computing efficiency compared to the approach implemented by FORGE, which computed overlaps using tabix in a distributed approach on the EBI computing farm. I stored the overlaps for each SNP and histone mark in an indexed SQLite database named *forge2.db*. I generated a binary string for each SNP, thus representing the presence (1) or absence (0) of an overlap with a histone mark broadPeak file. SNP-sample overlaps are precomputed and stored to increase analysis speed and minimise computational requirements.

2.3.4 Analysis strategy

For each set of test SNPs, FORGE2 performs a separate overlap analysis against the regulatory elements of each cell sample (39 samples for Epigenomics Roadmap), and counts the number of overlaps. FORGE2 then picks background SNP sets of the same number of SNPs as the test set, matched in bins for minor allele frequency (MAF), distance to transcription start site (TSS) and GC content. The background sets are overlapped with the regulatory elements of each cell sample and overlap numbers computed. These background overlap numbers are then used to generate a background distribution of the expected overlap counts for the test SNP set. 1000 matched sets are used by default. The enrichment value for the test SNP set is then expressed as the $-\log_{10}$ (binomial p-value). Enrichments outside the nominal 95th and 99th percentile of the binomial

distribution (after Benjamini-Yekutieli multiple testing correction) are considered significant. **Figure 2.7** contains a schematic of FORGE2 analysis.

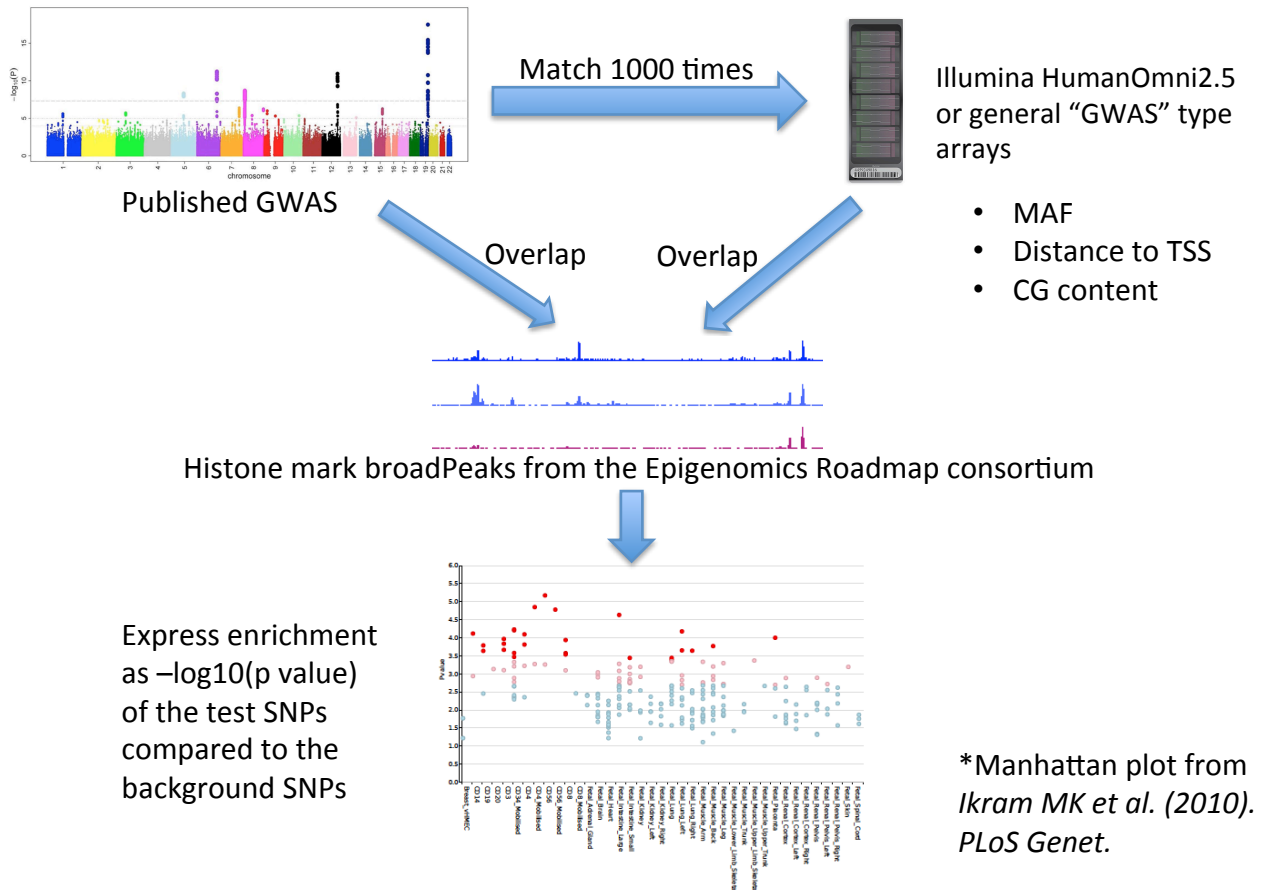


Figure 2.7: FORGE2 analysis schematic. For the input list of SNPs FORGE2 retrieves all overlap counts in bitstring format from the SQLite database. For each SNP in the list the corresponding bitstrings are then unpacked and overlaps with histone mark broadPeaks for each cell type are computed. FORGE2 then generates a background distribution by identifying SNPs matching the test SNPs by minor allele frequency (MAF), distance to transcription start site (TSS) and GC content. 1000 background SNP sets are generated, and the overlap analysis is repeated for each background set. Finally FORGE2 computes the $-\log_{10}$

(binomial p-value) of the overlap count of the input list compared to the background distribution.

2.3.5 Annotation categories

SNP annotation is important as background SNPs to the test SNP set are selected by picking SNP sets equal in SNP number to the test SNP set, with SNPs matching for GC content (GC), minor allele frequency (MAF) and distance to the nearest transcription start site (TSS). Decile bins are used to class SNPs in a specific category. Annotation for all these categories was obtained from the FORGE tool¹³⁹. Selection for these annotation categories is required when choosing background sites to avoid spurious enrichments due to a genomic bias for certain regions known to be enriched for certain histone marks (e.g. promoters). By matching the background to the input set for 3 levels of annotation FORGE2 limits enrichments caused by a bias in the input list for certain classes of genomic elements.

2.3.6 LD filtering

Proximal SNPs in strong linkage disequilibrium (LD) will tend to segregate together, and through this phenomenon of unequal segregation they may be associated with the same trait, echoing the association signal of a single causal SNP that underlies disease mechanism. Inclusion of a group of proximal SNPs without filtering for LD will over-estimate the enrichment signal as the same

regulatory element will be counted as a separate overlap several times. To avoid such an event FORGE2 includes an LD filter, which by default selects only one SNP for each LD cluster ($r^2 \geq 0.8$). FORGE2 reports SNPs that are removed to the user. Filter options include no filter, $r^2 \geq 0.1$ for a highly stringent filter, and $r^2 \geq 0.8$ (default and used in all analyses unless reported otherwise). For filtering the first SNP that appears in the filter set is retained and other SNPs from the same haplotype are filtered from analysis.

2.3.7 FORGE2 input

The user can supply a comma-separated list of SNP RSIDs or can run the RSIDs supplied from the default dataset of PR interval-associated SNPs.

Alternatively you can use a file. File data should be input in one of the following formats:

- a list of SNPs by RSID, one per line
- a list of genomic positions in BED format, corresponding to SNP positions

SNPs have to be present either on the Illumina_HumanOmni2.5 array (*Omni array SNPs*) or on the general set of arrays used in GWAS (*GWAS typing arrays*). *GWAS typing arrays* include the following arrays: Illumina_Human1M-duoV3, Illumina_Human660W-quad, HumanHap300v2, HumanHap550v3.0, Illumina_Cardio_Metabo, Affy_GeneChip_100K_Array, Affy_GeneChip_500K_Array, Affy_SNP6, HumanCNV370-Quadv3.

In addition to being present on one of the aforementioned arrays, SNPs also have to be present in the 1000 genomes phase I integrated call dataset (ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/phase1/analysis_results/integrated_call_sets/) to work in the analysis. FORGE2 excludes any SNPs that are not in this set from analysis. A minimum number of 5 test SNPs and a maximum number of 1000 test SNPs may be analysed in a given run.

2.3.8 FORGE2 outputs

Outputs include:

1. A PDF chart of FORGE2 -log10 (binomial p-values) generated using base R graphics. This chart is designed for use in slides and publications.
2. An interactive table showing the overlaps of individual SNPs against regulatory elements from each of the cell types.
3. An interactive chart to visually explore the data.
4. R code for regenerating these files (PDF, and interactive chart and table).
5. A raw text format file containing the original input data.
6. A tab-separated format (tsv) file with FORGE2 analysis results. Columns include z-score, p-value, cell, tissue, file (filename of the histone mark file analysed), SNP (a list of SNPs that overlap broadPeaks in the histone mark file analysed), accession (URL for the Epigenomics Roadmap data) and q value.

The $-\log_{10}$ (binomial p-values) are presented in each of the graphics by cell sample. Samples are listed alphabetically both by tissue and by cell type name. Point locations are plotted by $-\log_{10}$ (binomial p-value), and points are coloured according to BY-corrected q-values. The colouring is consistent for each of the graphics, blue ($q > 0.05$), pink ($q < 0.05$), and red ($q < 0.01$).

Results are presented by sample in either graphical (i.e. static PDF or interactive Dimple chart) or tabular forms (such as an interactive DataTables table or tab-separated file). A typical FORGE2 result for a given GWAS SNP set is the detection of enrichment (coloured in red or pink) in a cell type known or projected to be involved in the mechanism underlying the GWAS phenotype, for example blood cell types for rheumatoid arthritis SNPs (H3K4me1 analysis).

Alternatively no enrichment may be present, and all points will be blue. Causes for this could be technical reasons (for example, low overlap numbers), the absence of the relevant tissue, or the lack of a regulatory component underlying the GWAS association.

A list of GWAS catalogue SNPs associated with PR interval is available as a default dataset in FORGE2.

2.3.9 Multiple testing correction and false positive rates

To measure false positive rates, 1000 sets of SNPs at each of a series of SNP counts of 5, 10, 15, 20, 30, 40, 50, 100, 200, and 300 SNPs were randomly chosen from the 1000 genome phase 1 integrated SNP set. FORGE2 analysis was run for each set across the H3K4me1 Roadmap Epigenomics data, and the number of tests with q values below a threshold of 0.05 were recorded. These represent the false positives from 10 groups of 1000 trials at each of 39 samples i.e. 390,000 tests, and were used to calculate false positive rates at each significance threshold. FORGE2 incorporates an FDR-based multiple testing correction method (Benjamini-Yekutieli¹⁶²). The choice of Benjamini-Yekutieli was established in the eFORGE tool¹⁴⁶, which demonstrated this correction approach to be the most appropriate method when dealing with epigenetic data from multiple samples that exhibit dependence due to cellular lineage relationships. The other main FDR-based correction method, Benjamini-Hochberg¹⁶⁶, assumes that different samples are independent.

FORGE false positive levels are reported in the FORGE paper as 0.5-0.75% after 424,000 overlap tests (Bonferroni correction). FORGE2 presents a false positive rate (FPR) of 0% after 380,000 overlap tests. However, it is important to point out that methods such as BY that control the False Discovery Rate (FDR) are not directly comparable to methods such as Bonferroni that control the Familywise error rate. As these methods are doing two different things, I can only conclude that for FORGE2 (disregarding comparisons to FORGE) I have not been able to detect false positives after 380,000 overlap tests.

2.3.10 GWAS analysis

I applied FORGE2 to disease-associated SNPs from the GWAS catalogue¹²³. The entire GWAS catalogue was downloaded as of 12-6-2015 from <https://www.ebi.ac.uk/gwas/> and a parallel approach to analyse every disease-associated SNP list with FORGE2 was implemented.

2.3.11 Source code

FORGE2 source code can be downloaded from GitHub at <https://github.com/charlesbreeze/FORGE2>. FORGE2 installation and analysis have been performed on Mac OSX 10.10.1.

3 eFORGE Results

3.1 tDMP analysis

I first applied eFORGE to analyse probe sets of known tissue and cell type-specificity, focusing on probe sets obtained from studies on tissue-specific DNAm. These probe sets would serve as a positive control for confirmation of eFORGE function. I thus analysed 3 sets of tissue-specific DMPs (tDMPs)¹⁶⁷ and 3 sets of cell type-specific DMPs (cDMPs)¹⁶⁵. **Figure 3.1** shows analysis results, confirming the ability of eFORGE to detect tissue and cell type-specific patterns in DNAm data. Tissues analysed include whole blood, kidney and lung. For cell types I analysed data for CD14+ cells, NK cells and T cells.

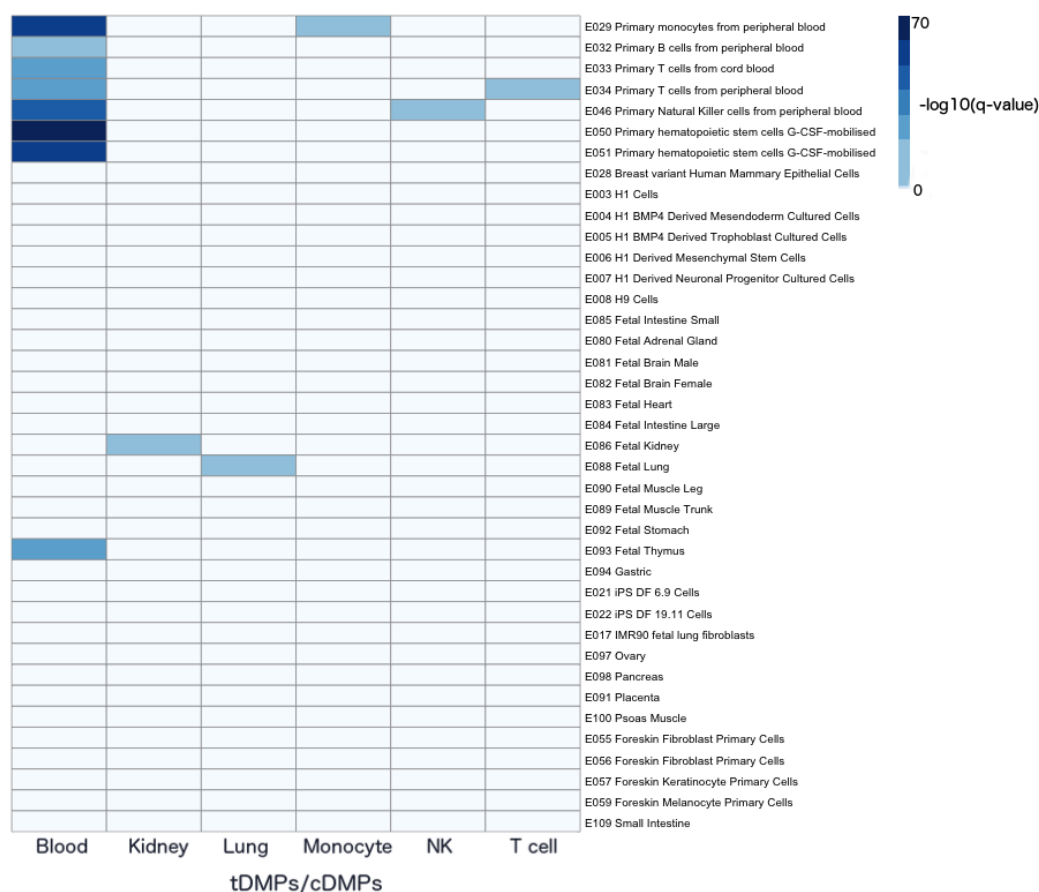


Figure 3.1: eFORGE tDMP analysis. eFORGE analysis results for tDMPs from whole blood, kidney and lung¹⁶⁷ and cDMPs from CD14+ cells, NK cells and T

cells¹⁶⁵. For each of these sets of tissue and cell type-associated positions eFORGE can detect the correct category using DNase I hotspot data from the Epigenomics Roadmap project. For each row (tissue) and column (450k probe set) eFORGE generates a random background. eFORGE then computes the enrichment in open chromatin sites of the 450k probe set against the random background. Original figure from Breeze et al., 2016¹⁴⁶.

3.2 EWAS findings

For eFORGE analysis I initially focused on autoimmune diseases. Given that autoimmune diseases are known to affect white blood cells, the important question in this area is the identification of the precise leucocyte cell types (e.g. CD4+, CD8+) that are most enriched for regions associated with a specific condition. Knowledge of the precise cell types involved would aid therapeutic target development, potentially helping to uncover key processes in immune system function. I analysed top probes from 3 EWAS on autoimmune diseases: rheumatoid arthritis (RA), systemic lupus erythematosus (SLE) and Sjögren syndrome. Each of these studies showed a different blood-specific enrichment in eFORGE analysis (**figure 3.2**). Specifically, the RA EWAS highlighted CD14+ cells (q value=5.53e-04). CD14+ cells have been shown to present accelerated maturation in RA¹⁶⁸. This finding is in contrast to the SLE EWAS, which pointed to T cells (q value=2.56e-05). T cells (especially CD4+ T cells) are crucial to the development of SLE¹⁶⁹. eFORGE also pointed to T cells for the Sjögren's

syndrome EWAS (q value= $1.31e-49$). For Sjögren's syndrome T cells have been put forward as specific therapeutic targets¹⁷⁰.

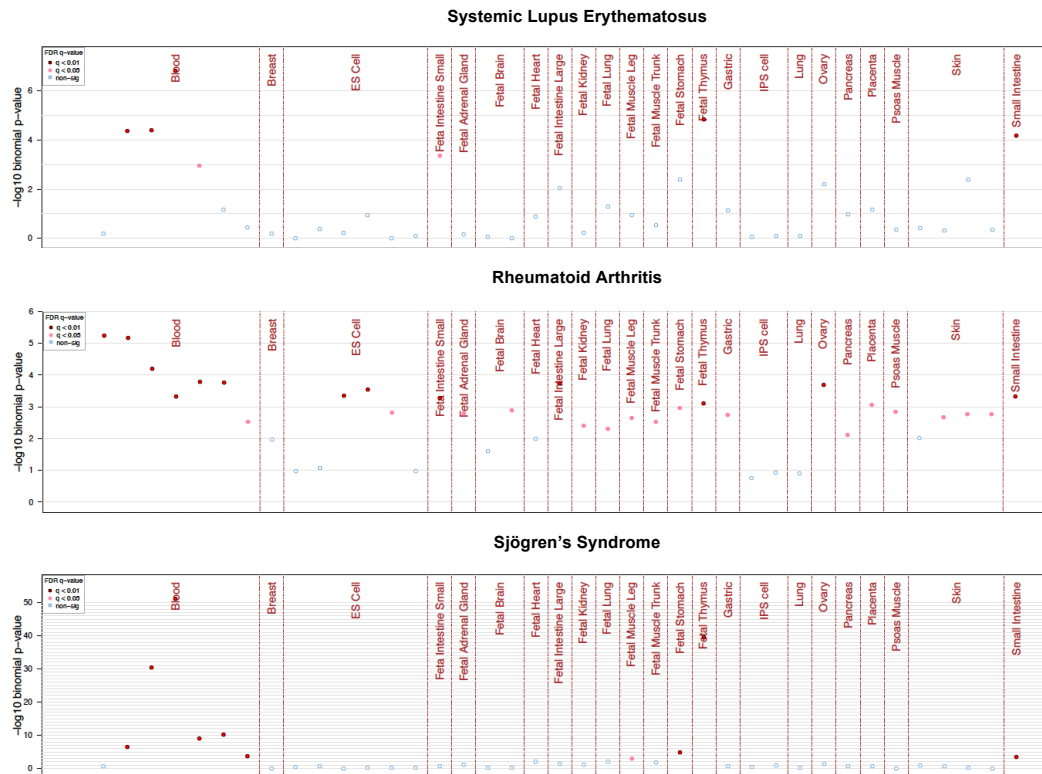


Figure 3.2: eFORGE analysis of autoimmune disease EWAS. This figure shows a tissue-specific signal for top probes from 3 autoimmune disease EWAS. In the top panel it can be observed that for a given SLE EWAS set of 86 probes I obtain a blood, intestine and thymus-specific signal beyond the cutoff threshold (red points). In the middle panel I observe a more general pattern of enrichment, with a strong blood signal for a set of 100 RA EWAS probes. In the lower panel I observe a blood and thymus-specific enrichment for a set of 753 probes for an EWAS on Sjögren syndrome. I obtained all top probes from the supplementary files of the studies ^{171,172,35}. Original figure from Breeze et al., 2016¹⁴⁶.

I then applied eFORGE analysis to autoimmune EWAS performed on non-blood tissue, focusing on an EWAS on Multiple Sclerosis (MS)¹⁷⁴. This EWAS measured DNAm differences in non-pathological brain tissue between MS cases and controls, using the aforementioned 450k array. The study attributed the DNAm differences mainly to neuronal processes and not to inflammation (hence the specification of “non-pathological brain tissue”, measured by the aspect of the tissue under the microscope). However, the Gene Ontology analysis performed by the study shows high enrichment for some immune processes. I decided to investigate the matter further, especially given that there is a resident immune cell type, termed microglia, that constitutes up to 15% of all cells in the central nervous system of mammals¹⁷⁵.

Preliminary analyses with eFORGE showed a strong immune signal for the top 1000 hypomethylated probes across DNase I hotspot data from several different consortia. In addition, eFORGE analysis for the same probe set across all histone marks narrowed the signal down to an immune enhancer-specific signal (H3K4me1) underlying the DNase I hotspot enrichment. I then studied the intersection of the top 1235 hypomethylated probes with active enhancers (n=1158) identified previously in microglial cells (Lavin et al. 2014). Significant co-localisation of these probes with the microglial-specific active enhancers was confirmed by a Fisher’s exact test (p-value= 2.70e-07, OR= 5.88, 95% CI= 3.19-9.96). This suggests that microglial activation may be a driving signal behind the observed DNAm differences.

One other aspect of EWAS research is the use of surrogate tissues. In this context,

researchers measure changes in DNAm from easily accessible tissues (e.g. whole blood or buccal cells) instead of the target tissue that is most relevant to disease mechanism. In the literature it has been suggested that changes in DNAm in surrogate tissues can “mimic” or reflect epigenomic perturbations happening in the target tissue¹⁷⁶. An alternate perspective would state that the observed DNAm changes are not mimicking DNAm changes in the target tissue but instead are specific to the surrogate tissue. To resolve a particular case of surrogate tissue use in EWAS, eFORGE analysis was performed on the top 110 probes from an EWAS on ovarian cancer²⁹. This EWAS was performed on whole blood using a pre-treatment discovery cohort, with the aim of identifying a DNAm signature separating ovarian cancer cases from healthy controls. Enrichment was found for CD14+ cells (q-value=1.37e-12, see **figure 3.3**). However, enrichment was not detected for ovary (q-value=1) or solid cancer tissues (q-value=1). This analysis pointed to a surrogate-tissue based EWAS in which the DNAm differences were independent of the target tissue, despite predicting ovarian cancer status. For a mechanistic understanding of this biomarker, I postulate that this immune process reflects an immune reaction to the ovarian cancer, rather than mimicry of DNAm changes occurring during ovarian cancer transformation.

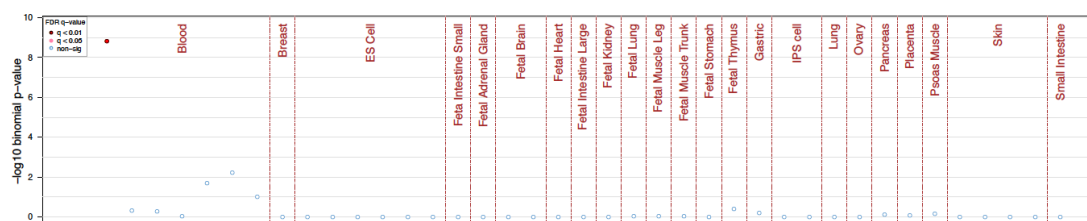


Figure 3.3: eFORGE analysis of an EWAS performed on a surrogate tissue.

For analysis I took 110 top probes from an ovarian cancer prediction EWAS

measured in whole blood²⁹. Results indicate a CD14+ cell-specific signal for Epigenomics Roadmap DNase I hotspot data.

As I have previously mentioned, among the potential confounders in EWAS, cell composition effects form one of the main sources of variance. The detection and correction of this large confounder has been one of the major breakthroughs that have allowed the field to perform large-scale EWAS for a variety of diseases. This is epitomised when using the main surrogate tissue for EWAS, namely whole blood. DNAm differences measured in this heterogeneous tissue, when uncorrected, many times result from a difference in cell numbers. For example, in a given EWAS there may be a higher proportion of CD4+ cells in cases than in controls, and therefore CD4+ cDMPs drive the DNAm differences observed in the study, despite these not being DNAm differences associated directly with disease mechanism, but rather with cell type ratios. Thus a difference in cell numbers is defined as a confounder in EWAS.

Because cell composition effects are driven by cell type-specific Differentially Methylated Positions, which in turn are co-localised with cell type-specific DNase I hotspots and detected by eFORGE, eFORGE is also a tool for detecting cell composition effects. One particular example of this is the detection of a skew in the granulocyte/lymphocyte ratio in the aforementioned EWAS on ovarian cancer²⁹. The DNAm signature detected (which is predictive of ovarian cancer status) was driven by a difference in cell counts between cases and controls. Specifically, the difference in CD14+ cells was the main driver of the DNAm

differences found in the study. CD14+ cell-specific Differentially Methylated Positions were the main source of variance in the study, and make up most of the study top hits. If analysed by eFORGE, this whole blood-based study strikingly points to CD14+ cells only. This is unexpected as typically different immune cell types will show epigenetic differences as part of an immune response or an environmental perturbation. This is further highlighted if we take into account that in a typical humoral response T cell activation and B cell activation are both required for antibody production¹⁷⁷. In addition, cytokine signalling constitutes a standard component of immune response, in which a crosstalk between different immune cell types occurs¹⁷⁷. Therefore it is unexpected to only find one cell type linked to the vast majority of DNAm changes in a study, and a much simpler explanation can be found by considering this phenomenon a result of cell composition effects.

One of the clearest areas for eFORGE application is in the study of cancer. Using eFORGE I discovered a clear stem cell signal across 5 cancer EWAS (**figure 3.4**), indicative of a preferential co-localisation of DNAm differences with stem cell-specific DNase I hotspots. No significant overlaps were observed for any other cell type. The following cancer EWAS were analysed: breast cancer (Fang et al., 2011)¹⁷⁸, colorectal cancer (Kibriya et al., 2011)¹⁷⁹, sporadic colorectal cancer (Laczmanska et al., 2013)¹⁸⁰, clear cell renal cell carcinoma (Arai et al., 2012)¹⁸¹, and adrenocortical carcinoma (Barreau et al., 2013)¹⁸². eFORGE analysis was performed for the top 330, 450, 240, 801 and 362 EWAS hits, respectively.

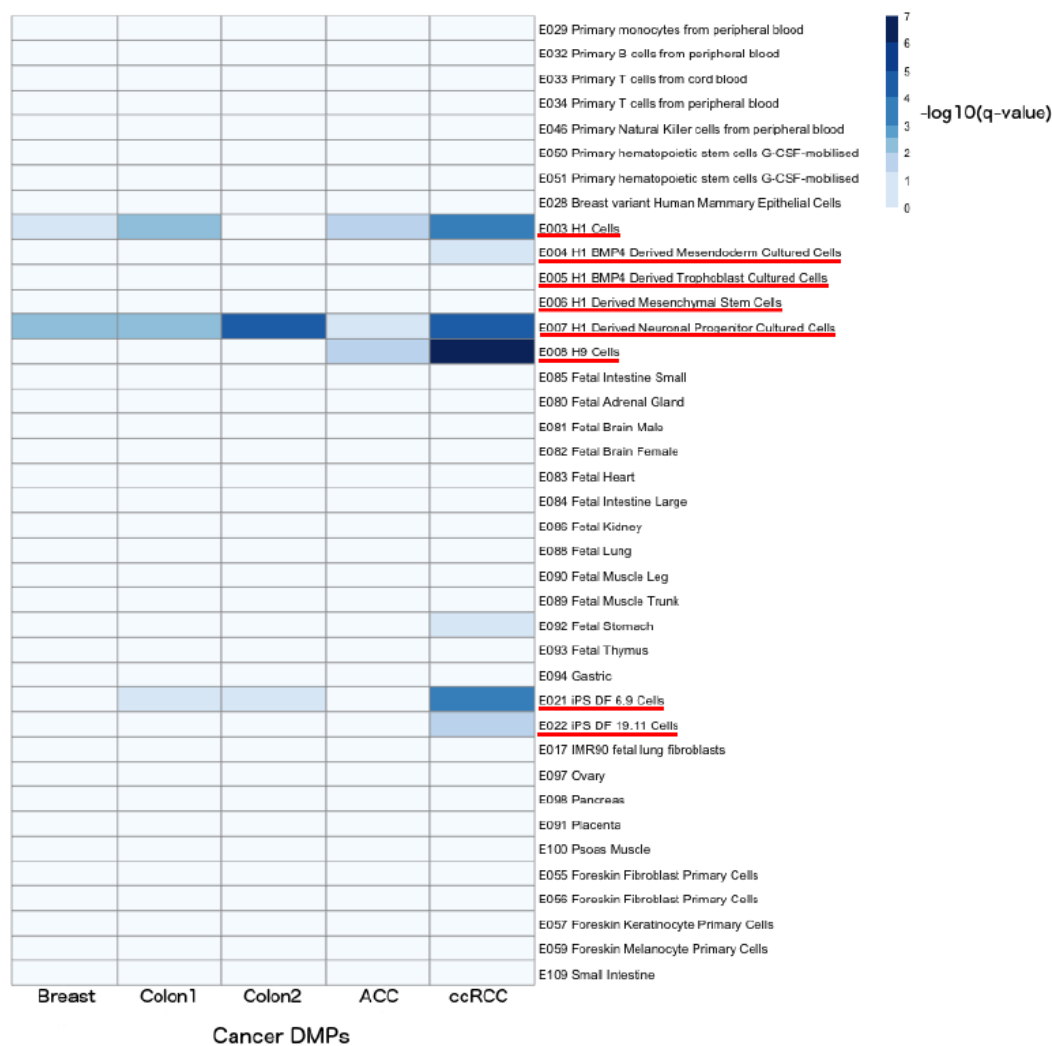


Figure 3.4: eFORGE analysis of 5 cancer EWAS. A stem cell-specific eFORGE DNase I hotspot signal can be observed. Cancer-associated probe sets analysed (columns) include 330 top probes from an EWAS on breast cancer metastatic behaviour¹⁷⁸ (left), 450 probes from an EWAS on colorectal carcinoma (second from left)¹⁷⁹ and 240 probes from an EWAS on sporadic colorectal cancer (centre)¹⁸⁰. In addition, I analysed 362 probes from an EWAS on adrenocortical carcinoma¹⁸² (second from right), and 801 probes from an EWAS on clear cell renal cell carcinoma¹⁸¹ (right). Stem cell row names are highlighted in red (corresponding to Epigenomics Roadmap DNase I hotspot stem cell samples). Original figure from Breeze et al., 2016¹⁴⁶.

However, until now I have only focused on specific diseases and traits. It is also important to perform an unbiased analysis applying eFORGE across a variety of previously unselected traits, in order to ascertain if tissue-specific eFORGE enrichment is a general phenomenon or rather a partial phenomenon that only occurs for a few pre-selected traits. In order to perform an unbiased selection of EWAS, I selected studies with $n > 100$ from Supplementary Table 1 of the latest relevant EWAS review⁸. Analyses of the 20 studies that presented eFORGE signal confirmed that tissue-specific enrichment is a general phenomenon (figures 3.5 and 3.6). 14 studies showed a pattern of tissue-specific enrichment, including blood-specific enrichment for 6 blood-based EWAS and stem cell-specific enrichments for five ageing and cancer EWAS (**figure 3.5**).

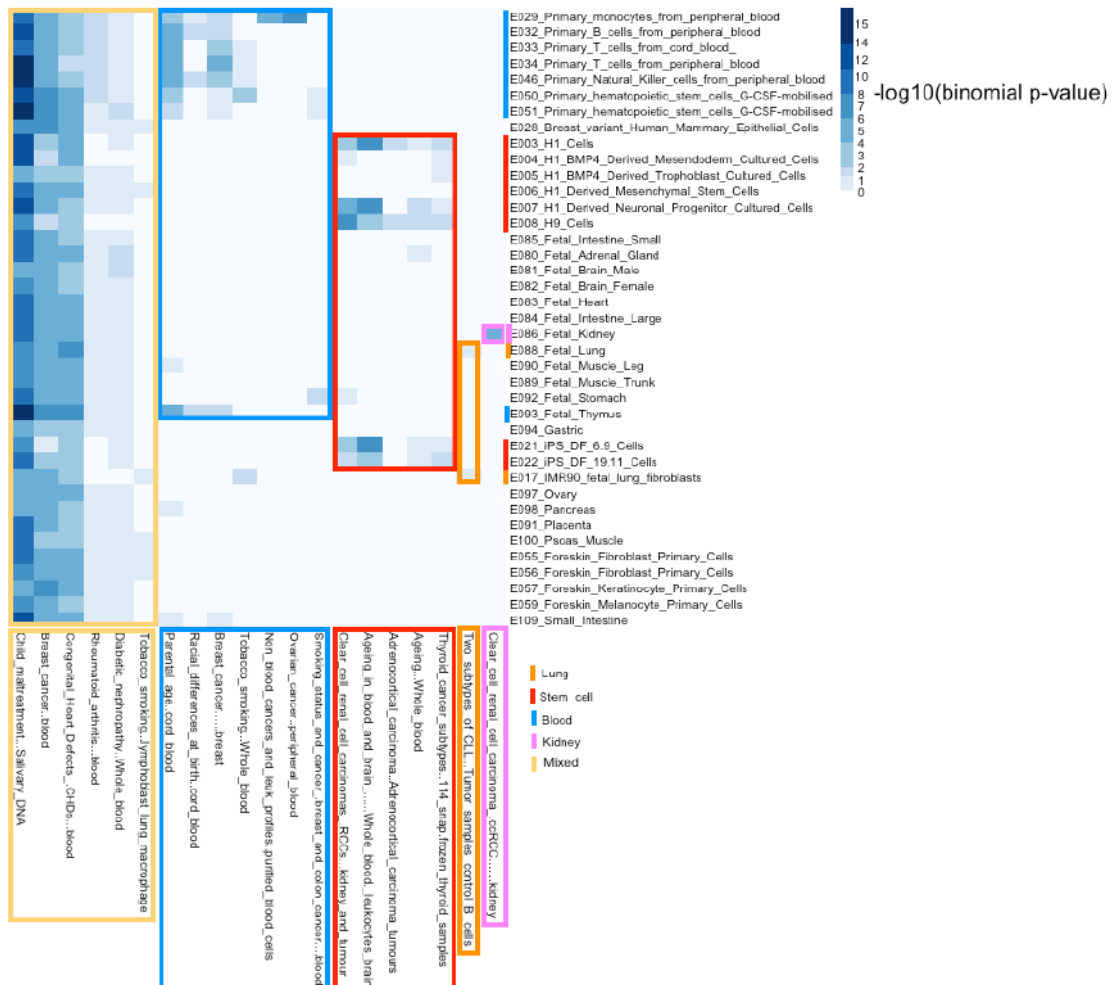


Figure 3.5: Aggregated eFORGE results for analysis across 20 EWAS. This analysis confirms the general nature of tissue-specific eFORGE enrichment. In the red box a clear stem cell signature for cancer and ageing EWAS can be observed. In the blue box an immune blood cell signature for EWAS performed on whole blood can be seen. The orange and pink boxes highlight lung- and kidney-specific enrichments observed for particular studies. The yellow box highlights the remainder of studies (6) which did not show enrichment for a specific tissue but rather a mixed enrichment across several tissues. Further research is needed to deconvolute these mixed signals. It is to be noted that this graph confirms that the majority of EWAS with eFORGE enrichment (70%) showed an enrichment that was tissue-specific, thus confirming the general

nature of EWAS tissue-specific enrichment for an unbiased selection of studies. Original figure from Breeze et al., 2016¹⁴⁶.

Unlike the field of GWAS, which has a well developed and extended catalogue (<https://www.ebi.ac.uk/gwas/>), the field of EWAS does not have a centralised repository with a list of studies, analyses, and related data for use by the community¹²³. I assembled the selected studies, along with eFORGE analyses, to form part of the first EWAS catalogue, the eFORGE catalogue (**figure 3.6 and table 3.1**). Inclusion of eFORGE analyses provides key data on the predicted cell type- or tissue-specific action of top EWAS hits for each study, thus aiding study interpretation, which has previously been posited as challenging¹⁸. This preliminary list is the starting point for a general EWAS catalogue, projected in years to come to be an invaluable resource for the epigenetics community.

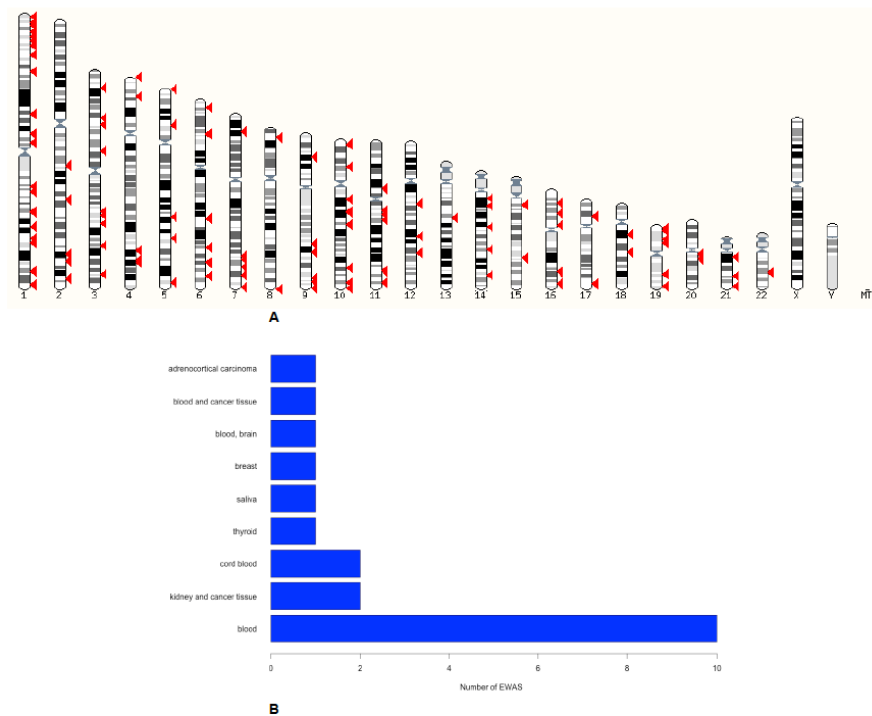


Figure 3.6: Genomic distribution and tissue origin from studies in the eFORGE catalogue. [A]: The karyotype plot contains top 10 study hits from each of the eFORGE catalogue EWAS. Bioinformatically the image was generated with [ensembl KaryoView \(http://www.ensembl.org/Homo_sapiens/Location/Genome\)](http://www.ensembl.org/Homo_sapiens/Location/Genome). No signal can be observed in X and Y chromosomes (probes on these locations are filtered in EWAS pipelines). Other than this, the location of top EWAS probes seems to show no particular genomic bias. [B]: This bar chart shows the tissues analysed by the EWAS in the catalogue. Blood, the most easily accessible tissue, is also unsurprisingly the most analysed category (10 studies). Original figure from Breeze et al., 2016¹⁴⁶.

Study number	Number of samples	PMID	Disease/Trait	Platform	Analysed tissue	eFORGE tissue	eFORGE link ID
1	2442	23034122	Ageing in blood and brain **,***	27 and 450k	Whole blood, leukocytes, brain	Stem cell signal	http://eforge.cs.ucl.ac.uk/eFORGE.v1.2/files/0xC8FF623812ED11E6B81D459BF8274B10/
2	135	23093492	Adrenocortical carcinoma**	27k	Adrenocortical carcinoma tumours	Stem cell signal	0x4D980C8425B311E6B2D23DDCF8274B10
3	910	23578854	Breast cancer	27k	Blood	Mixed enrichment	0x439DAF16242011E6B310C6D3F8274B10
4	138	22610075	Clear cell renal cell carcinomas (RCCs)**	27k	29 renal cortex, 109 tumors, 107 normal	Stem cell enrichment and other mixed enrichment	0x5550140A115211E6AF74B300F9274B10
5	367	21297937	Congenital Heart Defects (CHDs)	27k	Blood	Mixed enrichment	0xF3288FF2240A11E6A6E27FB6F8274B10
6	192	20687937	Diabetic nephropathy**	27k	Whole blood	Mixed enrichment	0x149CA7F4120E11E68EF480F5F8274B10
7	1230	22714737	Non-blood cancers, leukocyte profiles	27k	Purified peripheral blood cell types	CD14+ cell signal	0x62FCC88C240D11E6A13229BAF8274B10
8	168	21453505	Parental age**	27k	Cord blood	Blood signal	0x451A5388160D11E6945594A5F8274B10
9	201	21308978	Racial differences at birth**	27k	Cord blood	Blood signal	0xBC47687C161411E6BF88B9AFF8274B10
10	255	23579546	Breast cancer	27k, 450k	Breast	Blood and thymus signal	0xC4946164242011E68EC468D4F8274B10
11	317	23526956	Clear cell renal cell carcinoma (ccRCC)	27k, 450k	Kidney (tumour and non-tumour tissue)	Kidney signal	0xE35B9F46241F11E693220FD2F8274B10
12	656	23177740	Ageing***	450k	Whole blood	Stem cell signal	0x2B74E1A8137B11E6AF31C3D2F8274B10
13	192	23332324	Child maltreatment***	450k	Salivary DNA	Mixed enrichment with blood skew	0x6B81FA2E12EA11E6A1B50A97F8274B10
14	691	23334450	Rheumatoid arthritis	450k	Blood	Blood enrichment, with other tissues	0xA38CF988241E11E68AF555D0F8274B10
15	184	23175441	Smoking, breast/colon cancer	450k	Blood	Stomach enrichment	0x0E0FEA7E240E11E6A42C34BBF8274B10
16	1793	23691101	Tobacco smoking***	450k	Whole blood	Blood and thymus	0x3C0CE156121211E68B61CBFAF8274B10
17	138	22232023	Tobacco smoking***	450k	119 lymphoblasts and 19 lung macrophages	Breast epithelial signature	0x81C7A7041D2911E687D76CFFF8274B10
18	153	23064414	Two subtypes of CLL***	450k	Tumor samples and B cells from the controls	Mixed enrichment	0xDA59362C0E2311E6A0BCC9DDF8274B10
19	261	20019873	Ovarian cancer**	27k	Peripheral blood	CD14+ blood signal	0x33752BF89CD311E4BEA3AC33AA596114 (v1.1)
20	114	23666970	Thyroid cancer subtypes**	27k	114 snap-frozen thyroid samples	Cross-tissue mixed enrichment	0xA8797ECC0E0111E694B84EB2F8274B10

Table 3.1: list of EWAS in the eFORGE catalogue. Columns 1-6 contain study information, column 7 lists enriched cell types and tissues, and column 8 contains the eFORGE link ID (an example of a full URL is given in line 1, all IDs correspond to eFORGE v1.2 unless indicated).

4 FORGE2 results

4.1 FORGE2 analysis of the GWAS catalogue

Functional interpretation of disease-associated GWAS variants is a challenging task given that most variants are located in non-protein-coding regions¹³⁶. The lack of data for non-protein-coding regions is one of the main obstacles impeding the correct interpretation of these variants. This problem has been addressed by consortia such as Epigenomics Roadmap, ENCODE and BLUEPRINT which have generated abundant genome-wide epigenomic data covering both protein-coding and non-protein-coding regions^{81,183,184}. Epigenomics Roadmap and ENCODE data have been applied by the FORGE tool to analyse the enrichment of GWAS SNP sets in DNase I hotspots across a variety of tissues, yielding tissue-specific associations that take us one step closer to variant interpretation¹³⁹. I have developed FORGE2 to analyse the cell type-specific enrichment of GWAS SNP sets across 5 histone marks (H3K4me1, H3K4me3, H3K27me3, H3K9me3 and H3K36me3).

FORGE2 analysis was applied to the complete GWAS catalogue as of 12-6-2015 (see Methods). I observed many cell type- and tissue-specific patterns, a number of which were new, and many of which were also detected by FORGE for the same traits (**table 4.1, figure 4.1**). For example, in my analysis of SNPs associated with PR interval FORGE and FORGE2 both detect heart tissue as enriched (using DNase I hotspot and H3K4me1 data respectively, **figure 4.2**). This suggests that in certain cases regulatory elements represented by FORGE DNase I hotspots and FORGE2 histone marks overlap. In addition, in the case of mean platelet volume I observe two different associations in FORGE2: an enhancer

signature (H3K4me1) and a promoter signature (H3K4me3). Both associations point to CD34+ haematopoietic progenitor cells, a precursor of platelet-producing megakaryocytes and thus a plausible candidate for trait mechanism. This same cell type is enriched in FORGE DNase I hotspot analysis. The FORGE2 dual histone mark enrichment provides evidence for the utility of separating distinct classes of regulatory elements in functional overlap analysis. Importantly, I also report a series of novel findings not detected by FORGE. Specifically, four types of novel findings were observed.

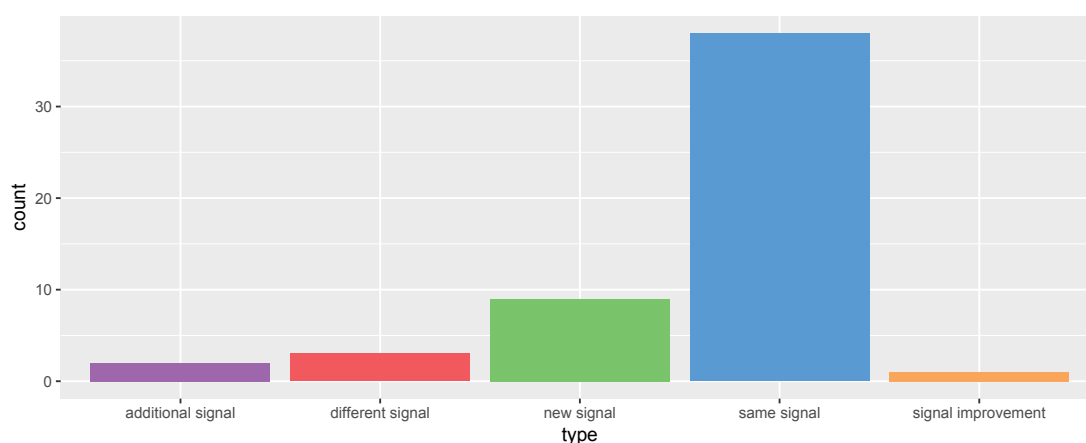


Figure 4.1: Types of FORGE2 findings when compared to FORGE. Most FORGE2 findings detect the same tissue-specific signal as FORGE for the same GWAS SNP sets (n=38). However, FORGE2 also detected new signals for SNP sets for which FORGE showed no signal (n=9). In addition, FORGE2 showed a lack of agreement with FORGE for a few studies, with each software tool pointing to different tissue-specific signals (n=3). FORGE2 also demonstrated the ability to extend signal detection to additional tissues not detected by FORGE (n=2) and

also showed evidence of signal improvement (defined as FORGE2 signal <0.01 for GWAS SNP sets with a FORGE p-value < 0.05 , $n=1$).

In one case I found that a low enrichment signal (significance level <0.05) already present in FORGE was found to present a higher enrichment in FORGE2 (significance level <0.01), this was the case for an H3K4me1 NK-specific signal in my analysis of “Systemic Lupus Erythematosus (SLE) and Systemic Sclerosis (SSc)” GWAS data. The overlapping SNP sets in FORGE2 analyses were found to include the majority (66%) of SNPs overlapping in FORGE analysis, which suggests that some NK cell-specific H3K4me1 sites in FORGE2 analysis are reflecting NK cell-specific DNase I hotspots in FORGE. These regions may potentially be NK cell-specific enhancers, as H3K4me1 is an enhancer-enriched mark. Mechanistically, SLE involves an immune response against multiple tissues while SSc involves the targeting of connective tissue. NK cells have an established role in the pathology of both SLE and SSc¹⁸⁵.

In other cases a completely new signal is detected, with no signal present in FORGE for the same SNPs (for example in my analysis of late-onset Alzheimer’s disease GWAS data, **table 4.2**). The specific case of late-onset Alzheimer’s disease data pointed to an immune response, as several immune cell types, including haematopoietic stem cells and monocytes, showed enrichment (**figure 4.3**). This is in agreement with previous findings on Alzheimer’s disease⁸¹. FORGE2 analysis also identified an involvement of T cells in two immune-related conditions (vein graft stenosis in coronary artery bypass grafting and Graves’

disease). In the former T cells constitute a candidate immune cell type associated with inflammation, a key factor in the pathophysiological process of stenosis after coronary artery bypass grafting. In the latter (Graves' disease), a defect in immune system function causes the production of abnormal TSH-mimicking antibodies, that overstimulate the thyroid leading to a pathological condition. FORGE2 analysis points to T cells as one of the main cell types involved in the regulatory process leading to antibody production in Graves' disease.

In stark contrast to previous approaches that have applied histone mark data to analyse GWAS SNPs⁸¹, I report FORGE2 enrichment for a previously GWAS-uninformative mark, namely H3K27me3 (a repressive mark linked to the polycomb repressive complex 2 -PRC2). Among other cases (**tables 4.1, 4.2 and 4.3**), H3K27me3 enrichment was observed for B cells when analysing a GWAS on C-reactive protein (CRP) levels. Considering the aforementioned FORGE2 FPR analysis (see Methods), it is statistically unlikely that this enrichment is a false positive. FORGE2 results suggest an immune mediation of CRP level-associated GWAS SNP action, potentially through disruption of PRC2-repressed regions (which would represent a novel mechanism of action for GWAS-linked variants).

In certain cases a completely different tissue is indicated by FORGE2 when compared to FORGE results (for example in my analysis of food allergy GWAS data, **table 4.3**). In the case of food allergy FORGE shows enrichment for spinal cord, whereas FORGE2 shows H3K4me3 stomach-specific enrichment. I propose that stomach represents a more plausible mechanistic tissue for this condition, which may be potentially mediated through mucosa-associated lymphoid tissue

(MALT). For F-cell distribution, that is, the abnormal presence of foetal haemoglobin in adults, previous FORGE data pointed to an intestinal signal. However, the main candidate cell type proposed by FORGE2 is CD34+ haematopoietic progenitor cells (H3K4me1 signal). This finding links GWAS SNPs to an erythrocyte precursor, providing a series of haematopoietic enhancers for further study (locations corresponding to variants rs10172646, rs10189857, rs1427407, rs6545816, rs6706648, rs6738440, rs7565301, rs7606173, rs9399137 and rs9399137).

Disease	FORGE2 signal	FORGE signal
Rheumatoid arthritis	H3K4me1 blood enrichment	DNase I hotspot blood enrichment
UC	H3K4me1 blood enrichment	DNase I hotspot blood enrichment
Crohn's Disease	H3K4me1 blood enrichment	DNase I hotspot blood enrichment
IBD	H3K4me1 blood enrichment	DNase I hotspot blood enrichment
T1D	H3K4me1 blood enrichment	DNase I hotspot blood enrichment
Self-reported allergy	H3K4me1 blood enrichment	DNase I hotspot blood enrichment
Red blood cell traits	H3K4me1 CD34+ cell enrichment	DNase I hotspot CD34+ cell enrichment
QT interval	H3K4me1 heart enrichment	DNase I hotspot heart enrichment
MS	H3K4me1 blood enrichment	DNase I hotspot blood enrichment
PR interval	H3K4me1 heart enrichment	DNase I hotspot heart enrichment
Restless legs syndrome	H3K4me1 heart enrichment	DNase I hotspot heart enrichment
Breast cancer	H3K4me1 general enrichment	DNase I hotspot general enrichment
Waist-hip ratio	H3K4me1 lung enrichment	DNase I hotspot lung and kidney enrichment
Waist-to-hip ratio adjusted for BMI	H3K4me1 general enrichment	DNase I hotspot general enrichment
Height	H3K4me1 general enrichment	DNase I hotspot general enrichment
Electrocardiographic traits	H3K4me1 heart enrichment	DNase I hotspot heart enrichment
Platelet count	H3K4me1 CD34+ cell enrichment	DNase I hotspot CD34+ cell enrichment
Mean corpuscular haemoglobin	H3K4me1 CD34+ cell enrichment	DNase I hotspot CD34+ cell enrichment
Mean platelet volume	H3K4me1 CD34+ cell enrichment	DNase I hotspot CD14+, CD34+ cell enrichment
Chronic lymphocytic leukaemia	H3K4me1 blood lymphoid enrichment	DNase I hotspot blood and thymus enrichment
Coeliac disease	H3K4me1 blood and thymus enrichment	DNase I hotspot blood and thymus enrichment
Mean platelet volume	H3K4me3 CD34+ cell enrichment	DNase I hotspot CD14+, CD34+ cell enrichment
IBD	H3K4me3 CD34+ cell and blood enrichment	DNase I hotspot blood enrichment
Platelet count	H3K4me3 CD34+ cell enrichment	DNase I hotspot CD34+ cell enrichment
QT interval	H3K4me3 heart enrichment	DNase I hotspot heart enrichment
Chronic lymphocytic leukaemia	H3K4me3 blood enrichment	DNase I hotspot blood and thymus enrichment
Crohn's disease	H3K4me3 CD34+ cell enrichment	DNase I hotspot blood enrichment
MS	H3K4me3 blood enrichment	DNase I hotspot blood enrichment
Height	H3K4me3 skin enrichment	DNase I hotspot general enrichment
Coeliac disease	H3K4me3 T cell enrichment	DNase I hotspot blood and thymus enrichment
Breast cancer	H3K4me3 immune and skin enrichment	DNase I hotspot general enrichment
UC	H3K27me3 several organ enrichment	DNase I hotspot several organ enrichment
Height	H3K36 very high enrichment in all tissues except stem cells	DNase I hotspot general enrichment
MS	H3K36 blood enrichment	DNase I hotspot blood enrichment
Rheumatoid arthritis	H3K36 thymus enrichment	DNase I hotspot blood and thymus enrichment
IBD	H3K36 T cell and B cell enrichment	DNase I hotspot blood enrichment
Mean platelet volume	H3K36 thymus enrichment	DNase I hotspot blood and thymus enrichment
Primary biliary cirrhosis	H3K36 NK enrichment	DNase I hotspot NK enrichment

Table 4.1: Common findings between FORGE2 and FORGE. Findings in which FORGE2 identified a tissue listed among FORGE tissues are also included. The most common tissue category for the traits analysed is blood, with 20 out of 36 categories (55.55%) showing a blood-related enrichment. BMI: Body mass index. IBD: Inflammatory bowel disease. MS: Multiple sclerosis. UC: Ulcerative colitis. T1D: Type 1 diabetes.

Disease	FORGE signal	FORGE2 signal
Vein graft stenosis in coronary artery bypass grafting	No signal	T Cell signal (H3K4me1)
Graves' disease	No signal	T Cell signal (H3K4me1)
Attention deficit hyperactivity disorder combined symptoms	No signal	Stem cell (H3K4me1)
Late-onset Alzheimer's disease	No signal	Strong blood signal (H3K4me1)
C-reactive protein levels	No signal	B Cell (H3K27me3)
Political ideology	No signal	Brain (H3K27me3)
Blood metabolite levels	No signal	Many cell types (H3K36me3)
Total ventricular volume	No signal	H3K27me3 low gastric enrichment
Oesophageal cancer squamous cell carcinoma	No signal	H3K36 general enrichment

Table 4.2: FORGE2 findings not detected in FORGE analyses.

Disease	FORGE signal	FORGE2 signal
F cell distribution	Intestinal signal	Brain and CD34+ cells (H3K4me1)
Food allergy	Spinal cord	Stomach signal (H3K4me3)
Thyroid cancer	Stem cells	CD34+ cells (H3K27me3)

Table 4.3: FORGE2 findings that show a different tissue from FORGE.

Disease	FORGE signal	FORGE2 signal
Primary biliary cirrhosis	CD56+ cells	Signal across blood (H3K4me1)
Psoriasis	Blood	Skin and blood (H3K36me3)

Table 4.4: Additional tissue FORGE2 findings. These FORGE2 findings show enrichment for additional tissues when compared to FORGE (data for these tissues is also present in FORGE).

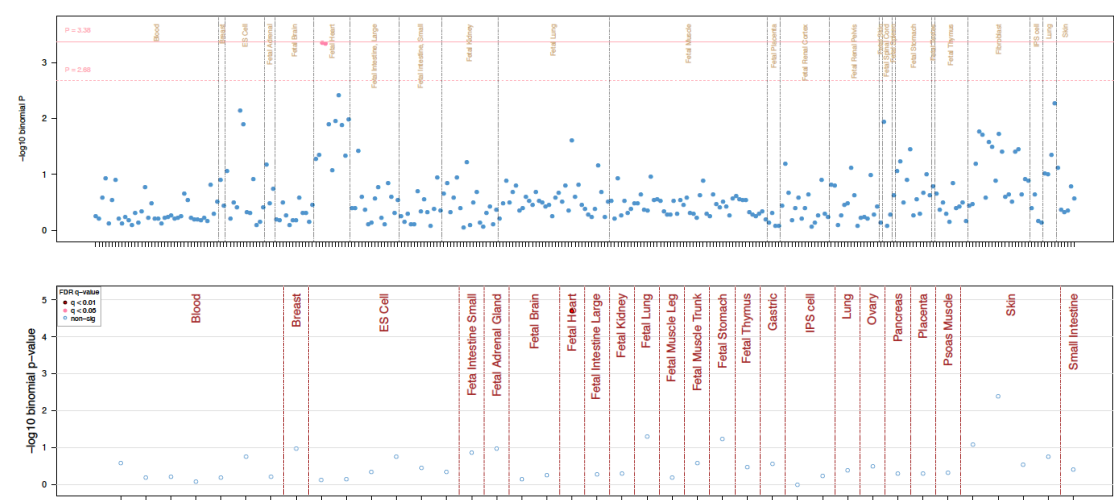


Figure 4.2: FORGE and FORGE2 PR interval GWAS analysis. FORGE results for PR interval SNPs show enrichment for DNase I hotspots in heart cells. FORGE2 H3K4me1 results for the same SNPs also highlight heart tissue.

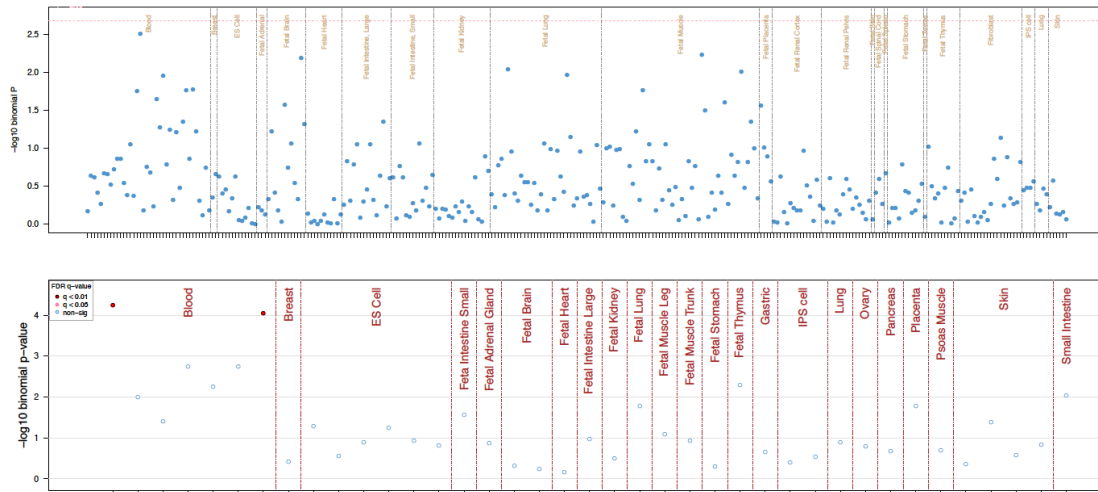


Figure 4.3: FORGE and FORGE2 Alzheimer’s disease GWAS analysis. FORGE results for Alzheimer’s disease SNPs show no significant enrichments, FORGE2 results show enrichment for immune cell types.

In some cases additional tissues not enriched in FORGE are detected in FORGE2 analysis (for example in my analysis of psoriasis GWAS data, **table 4.4**). Analysis of psoriasis GWAS data was found to extend the range of tissues associated with this autoimmune condition that targets the skin. Surprisingly, I observed not only enrichment for the proposed effector tissue (immune blood cells), but also for the target tissue (skin). This enrichment was driven by the transcription-associated mark H3K36me3. For psoriasis GWAS data no enrichment was observed for other histone marks. This H3K36me3-specific enrichment suggests that the variants linked with psoriasis detected through GWAS do not preferentially affect enhancers or promoters (unlike many of the analyses that have been presented here), but rather sites that direct transcription-related processes (e.g. transcription elongation, activation and alternative splicing). This is yet another mechanism of action for GWAS-related regions, and I propose that,

as more and more data become available, different mechanisms of action of GWAS SNPs on gene regulation will appear.

4.2 Analysis of psoriasis GWAS data

Further analysis of the psoriasis signal revealed that some of the regions underlying enrichment in blood (with B cells as the highest category) are different from the regions underlying enrichment in skin. Six out of 34 B cell-overlapping SNPs are specific to B cells (12%) and 12 out of 40 skin-overlapping SNPs are specific to skin (26.1%). Twenty-eight SNPs were found to overlap H3K36me3 broadPeaks in both skin and B cells (60.9%). Applying FORGE2 analysis I can detect separate H3K36me3 signal for these independent SNP sets (**figure 4.4**). This shows that some of the SNPs do not only overlap distinct elements between B cells and skin but also that the independent SNP sets tested show specificity to B cells and skin cells when compared to a range of other tissues.

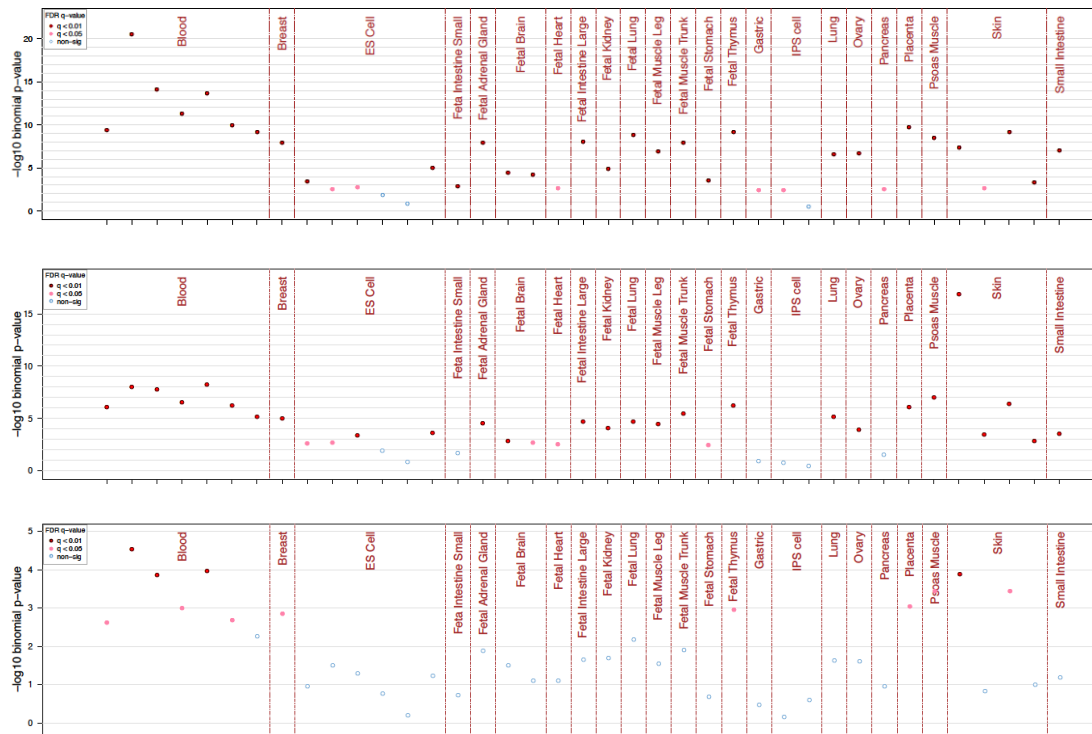


Figure 4.4: FORGE2 psoriasis GWAS analysis. FORGE2 analysis with “B cell only” SNPs shows a clear skew towards B cells and blood (top panel). Analysis with the “skin only” SNPs shows skin as distinct top category, with other tissues at a much lower level (note the logarithmic scale). Analysis with all 138 psoriasis-associated GWAS SNPs shows signal for both blood and skin.

I also report data on the distinct molecular pathways underlying these findings based on the genes where this H3K36me3 signal is found (using Ensembl data through biomaRt in R)¹⁸⁶. Many of the genes linked to B cells showed links to immune-specific mechanisms. For example IFNLR1 (interferon lambda receptor 1) and IFIH1 (interferon induced with helicase C domain 1) show key links to viral response, of which B cells are a key component¹⁸⁷. Another immune-related gene, ELM01 (engulfment and cell motility 1), is related to phagocytosis and motility-linked immune response. Historically B cells were considered distinct from cells with a phagocytic activity (which include, among others, monocytes

and macrophages). However, recent findings have shown that certain B cell subsets have a phagocytic activity across a range of vertebrates¹⁸⁸, including mammals¹⁸⁹. Confirming the immune-associated nature of the list of genes obtained for B cell regions, analysis with amiGO⁴⁵ revealed immune response pathways among the highest categories. Many of the genes linked to skin were related to dermal and connective tissue processes, with connective tissue as the top category in amiGO⁴⁵ analysis. These genes include COG6 (component of oligomeric Golgi complex 6) and TNIP1 (TNFAIP3 interacting protein 1). COG6 is expressed in skin tissue, specifically sweat glands, and is associated with skin-related mendelian conditions¹⁹⁰. In addition, research on TNIP1 has evidenced links between this gene and psoriasis¹⁹¹. Proliferation of human keratinocytes was shown to increase when TNIP1 was downregulated, and this was further linked to more severe psoriasis-like conditions in a mouse model of dermatitis¹⁹¹. In addition, genes also linked to skin include IL13 (interleukin 13), a cytokine shown to induce B cell proliferation¹⁹², TRAF3IP2 (TRAF3 interacting protein 2), a gene involved in immune regulation and nuclear factor kappa B signalling (also a characteristic of TNIP1). Nuclear factor kappa B signalling has been described as an essential process in psoriasis¹⁹³. Four other genes were linked to skin including PTRF (polymerase I and transcript release factor), ETS1 (ETS proto-oncogene 1), ZNF816-ZNF321P (annotated as ZNF816 ZNF321P readthrough) and ZNF816 (zinc finger protein 816).

This analysis of psoriasis GWAS data shows that there is a separation in H3K36me3 enrichment signal between two distinct SNP sets that act differently in different

tissues. Through FORGE2 I can separate the original GWAS SNP list into these 2 subsets and show that they contain SNPs located in genes linked to distinct biological processes that act both in skin cells and B cells, potentially underlying the mechanisms driving psoriasis pathology.

4.3 Conclusions on GWAS analyses

I have uncovered many cell type- and tissue-specific enrichments for disease-associated GWAS variants. Some of these enrichments replicate findings made by FORGE, and many are novel, such as the uncovering of a previously unreported H3K27me3 enrichment across several traits, and the discovery of skin regions associated with psoriasis GWAS SNPs. I have explored the case of psoriasis further using computational approaches, showing potential gene targets associated with this H3K36me3 enrichment signal, and separating skin and blood components for further study. It is important to consider that one of the most important applications of FORGE2 is to prioritise variants for experimental analysis. As an example of this application I have used FORGE2 to prioritise variants for 4C-seq analysis, complementing FORGE and demonstrating the utility of studying variants in a tissue- and histone mark-specific context.

5 4C-seq results

5.1 Introduction

Intergenic and intronic disease-associated GWAS and EWAS variants present a challenge for functional interpretation^{17,136}. Improved data and novel computational approaches are required to rise up to this challenge. In previous sections I have shown the application of eFORGE and FORGE2 to uncover the cell type-specific action of disease-associated variants. I have also highlighted that these bioinformatics approaches prioritise targets in a cell type-specific context for experimental study. In addition to designing and developing these computational approaches, I have implemented 4C-seq experiments to analyse regions prioritised by eFORGE and FORGE2, providing valuable data on the cell type-specific interactions of disease-associated variants.

5.2 4C-seq of regions homologous to neural disease loci in mNSCs

The laboratory mouse is one of the main model organisms in biology¹⁹⁴. Substantial resources are available when studying the mouse genome, which has provided many insights in the study of human diseases^{194,195}. In addition, an important reason to use mNSCs as a test system for initial 4C-seq experiments was that mNSCs were an established cell type for 4C-seq in the protocols of my collaborators (Hadjur group)^{196,197}. Using mNSCs I can focus on regulatory interactions in a developmental context where many variants are posited to act¹⁹⁸.

In order to study neural disease-associated loci I prioritised 6 neural disease-associated regions from published GWAS and EWAS for 4C-seq analysis (see Methods for prioritisation strategy). As mentioned previously, to prioritise these SNPs I took into account multiple levels of information, including DNase I hotspot data, eQTL status, PCHi-C data and sequence homology (**figure 5.1**). While some caveats can be made regarding the evolutionary distance between human and mouse, including the fact that only 35.6% of DHSs are conserved between the two species¹⁹⁹, evidence suggests that for homologous sites a much higher proportion of DHSs (59.8%) is conserved¹⁹⁹. Examples of sites with conserved regulation between human and mouse include GWAS sites such as the obesity-associated FTO locus^{130,135} and the QRS duration-associated SCN10A locus²⁰⁰.

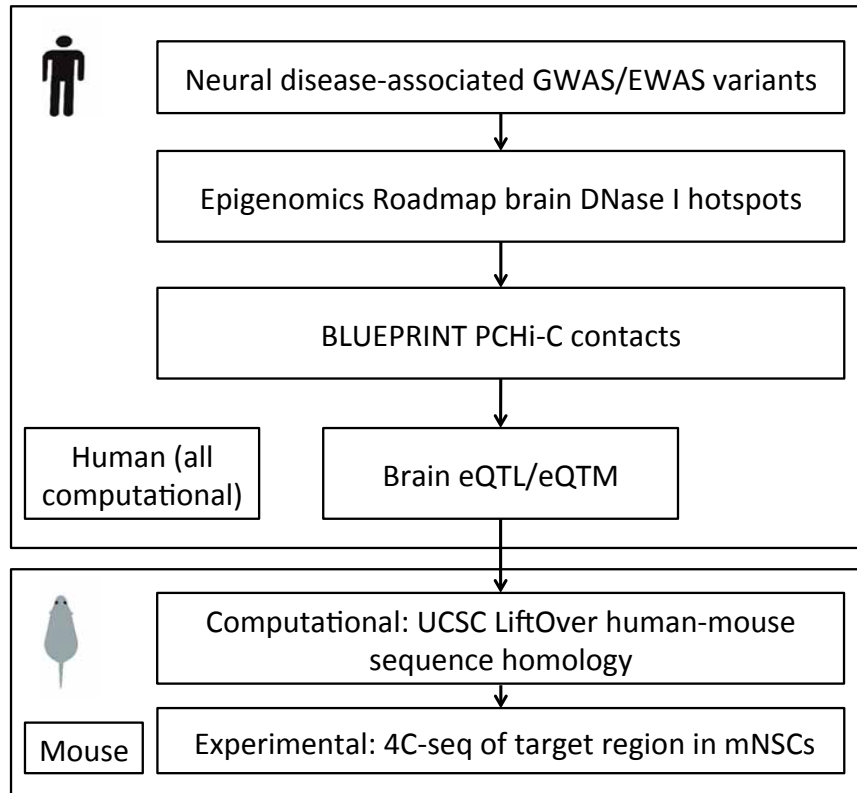


Figure 5.1: Schematic of 4C-seq mNSC computational and experimental analysis. Human and mouse sections are clearly separated. Computational analysis included GWAS/EWAS data which were intersected with data for DNase I hotspots, PChi-C contacts and eQTL/eQTM. Evidence for region homology was taken from UCSC LiftOver (remap ratio 0.95) and loci were subject to visual examination on the UCSC genome browser. 4C-seq experiments were then performed in mNSCs for each of the selected regions. Human and mouse icons were taken from Odom et al., 2007²⁰¹.

In my analyses of mNSC interactomes I found evidence of distal interaction in two cases. Specifically, homologous regions to cg02672452 and rs548181 showed interaction with *Wdr1* and *Fez1*, with relevance for the aetiology of Alzheimer's disease and schizophrenia.

cg02672452 is an EWAS DMP found to be associated with Alzheimer's disease by two separate studies^{202,203}. These EWAS studied different areas of the brain in search of DMPs associated with Alzheimer's disease. To my knowledge, the region containing cg02672452 was not previously linked to any gene by Hi-C or PCHi-C. In addition, this region is not known to be an eQTM (expression Quantitative Trait Methylation, a genomic CpG associated through DNA methylation to the expression of a particular gene)¹⁵¹. cg02672452 is located in an intron of Homo sapiens lymphocyte-specific protein 1 pseudogene (LOC654342). The homologous region in mouse was located between Otop1 and Drd5 genes. cg02672452 was found to overlap DNase I hotspots across many human tissues (including brain samples from the Roadmap Epigenomics Consortium).

For the homologous region to cg02672452 in mNSCs I observe downstream interaction (**figure 5.2**), associating cg02672452 with Wdr1. WDR1 in humans codes for a protein that induces disassembly of actin filaments in conjunction with the ADF/cofilin family of proteins²⁰⁴. ADF/cofilin-actin rods have been proposed as a therapeutic target for Alzheimer's disease²⁰⁵. The proposed mechanism is thus: the EWAS DMP cg02672452 is located in an enhancer region (marked by a DNase I hotspot) that loops with Wdr1, potentially regulating expression of Wdr1, which would have downstream effects in the mechanism of Alzheimer's disease through the regulation of actin rod formation.

rs548181 is a GWAS SNP found to be associated with schizophrenia by several studies, including a family-based replication and refinement study²⁰⁶. To my knowledge, the functional role of this SNP in schizophrenia is unknown, and all previous studies, including the aforementioned refinement study, could not elucidate a functional role for this SNP. rs548181 is located in an intron of Homo sapiens STT3A antisense RNA 1 (STT3A-AS1), a long non-coding RNA. This GWAS SNP overlaps DNase I hotspots in human brain samples from the Roadmap Epigenomics Consortium, and, in addition, is an eQTL for many human genes, including CCDC15, CHEK1, DDX25, EI24, ESAM, FEZ1, HEPACAM, HEPN1, PUS3, ROBO3, SRPR, STT3A and TBRG1 (source: Braineac database¹⁵⁴). A homologous region in mouse was identified using UCSC LiftOver. This homologous region is located in an intron of Mouse gene AK018988 (uc009otz.1, RIKEN identified cDNA), potentially a non-classified homologue of Homo sapiens STT3A antisense RNA 1 (STT3A-AS1).

For the region homologous to the human locus containing rs548181 in mNSCs I observe downstream interaction (**figure 5.3**), associating rs548181 with Fez1, a gene strongly linked to schizophrenia by GWAS. I thus propose the following mechanism: rs548181 is found in an enhancer (marked by a human brain DNase I hotspot) that loops with the promoter and the gene body of Fez1. This variant, which is an eQTL for FEZ1 in human, is posited to directly regulate expression levels of FEZ1, thus potentially affecting the aetiology of schizophrenia. FEZ1 is already a schizophrenia-associated gene²⁰⁷. In this example I observe eQTL mapping, GWAS variants and 4C-seq converging on functional disease mechanism.

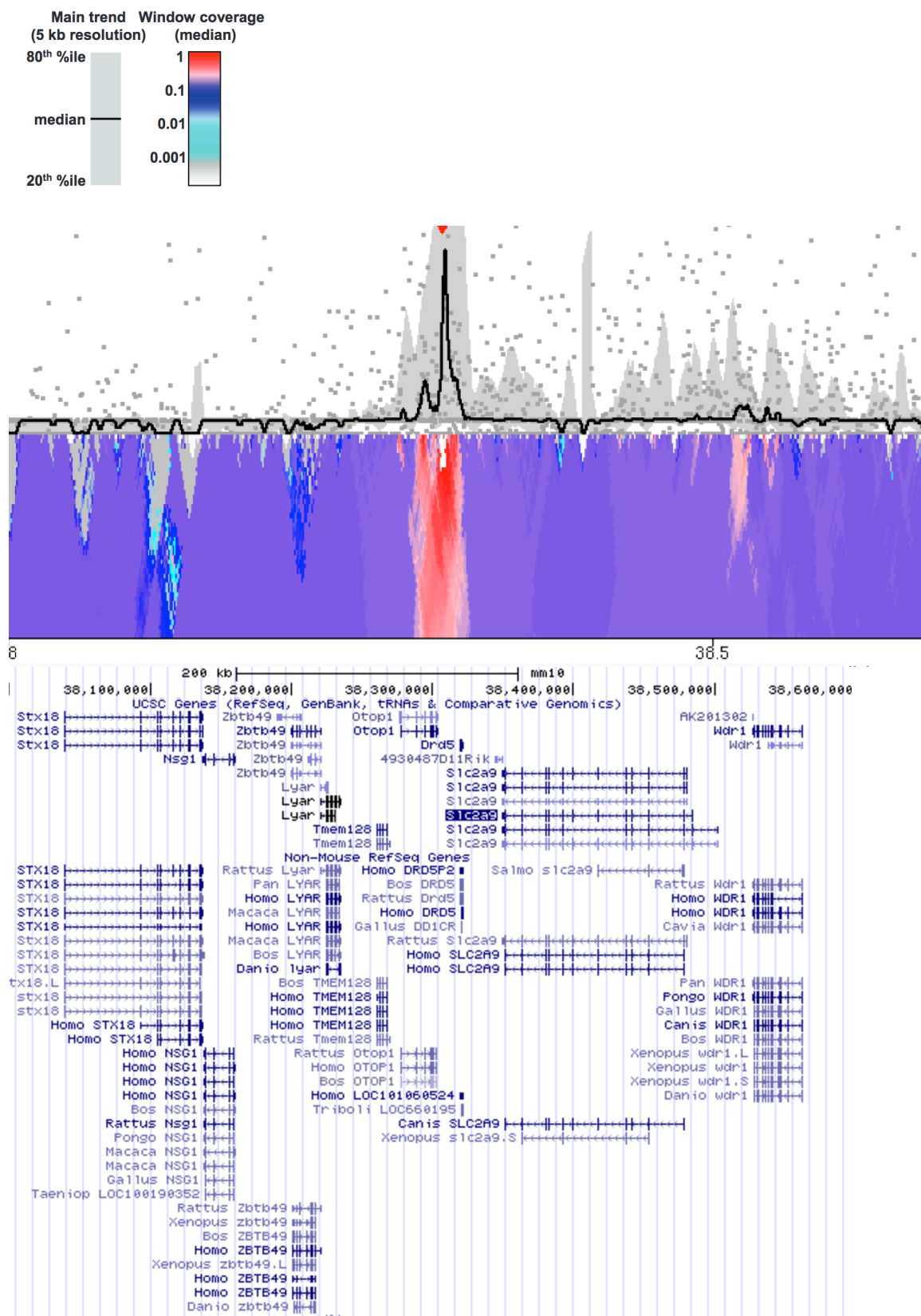


Figure 5.2: 4C-seq results for the mouse region homologous to the human locus that contains Alzheimer's disease EWAS DMP cg02672452. The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The colour scale was taken from Van de Werken et al., 2012¹⁴⁹. The positions of local genes for mm10 (middle) and homologous genes in other species (lower section, taken from the UCSC genome browser, <https://genome.ucsc.edu/>) are also indicated in blue.

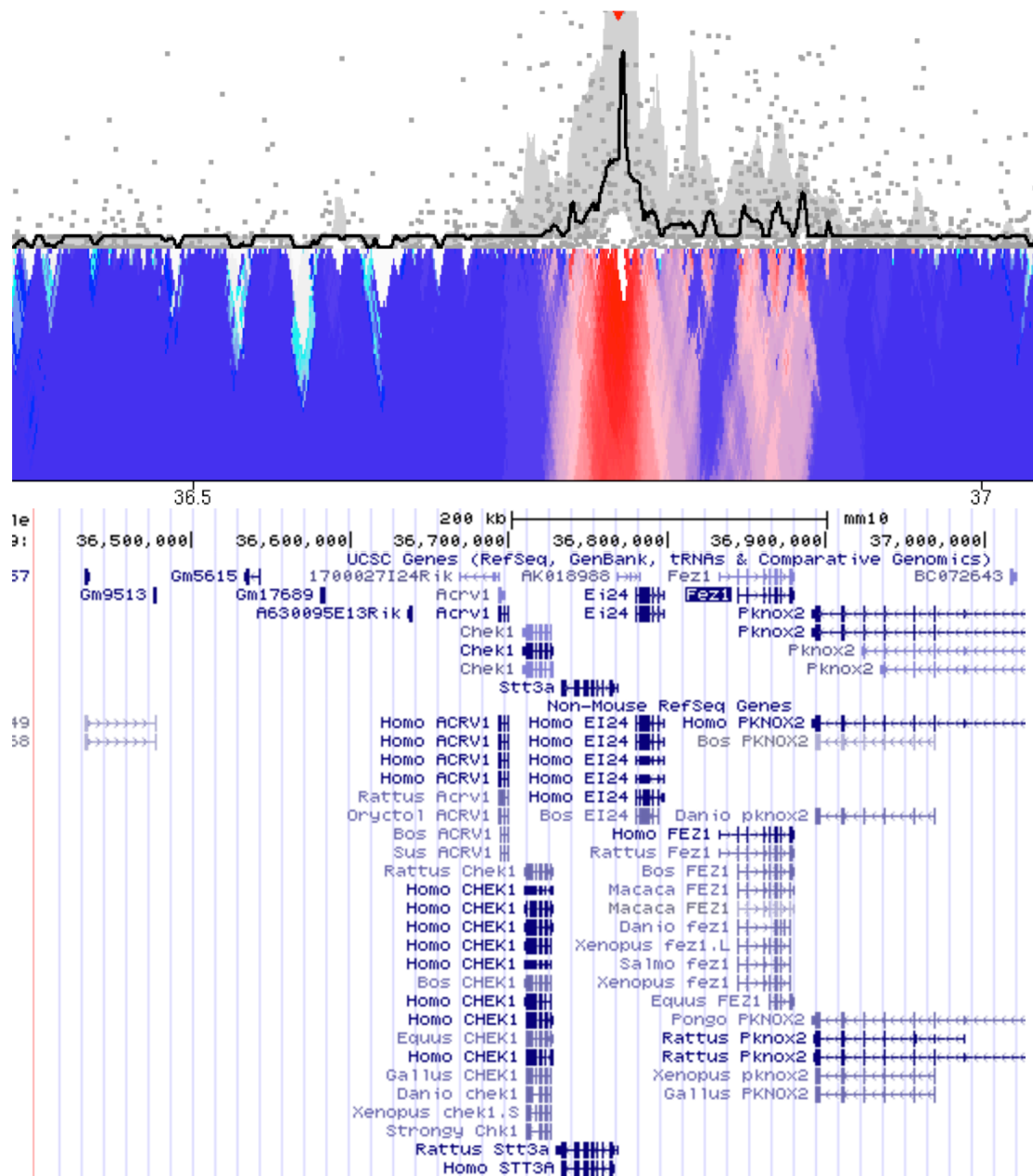


Figure 5.3: 4C-seq results for the mouse region homologous to the human locus that contains schizophrenia-associated GWAS SNP rs548181. The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to

grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes for mm10 (middle) and homologous genes in other species (lower section, taken from the UCSC genome browser, <https://genome.ucsc.edu/>) are also indicated in blue.

Of the loci analysed, I also found several cases in which only local interactions could be detected within 2Mb (see Methods for loci features and association data). For these variants evidence seems to point to a local mechanism of action of the GWAS/EWAS signal.

Specifically, for the Gcdh locus I can observe proximal interactions with Calr, Farsa, Syce2, MIR7069, Gcdh and Klf1 (**figure 5.4**). Of all these genes three have a role in neurological processes. Farsa codes for a protein related to axonal development of hippocampal neurons²⁰⁸, GCDH deficiency has been linked to late-onset neurological disease²⁰⁹ and the human KLF family of genes has been linked to several neurological conditions²¹⁰. In my opinion these three candidates merit further study to elucidate the action of GWAS SNP rs8012.

For the Synj2 locus I can observe interactions with Snx9, Synj2 and Serac1 (**figure 5.5**). Of these three genes two are involved in neuronal processes: the specific role of Synj2 is vesicle uncoating in neurons, linking this region to this key neurological process²¹¹. Serac1 is linked to the neurological conditions Leigh

syndrome and MEGDEL syndrome²¹². In my opinion these two candidates merit further study to elucidate the meQTL action of SNP rs1009014.

For the Ppm1m locus interactions can be observed with a range of genes including Glyctk, Mirlet7g, Wdr82, Ppm1m, Twf2, Tlr9, Alas1 and Poc1a (**figure 5.6**). Of all these genes Glyctk plays an important metabolic role in neural tissue²¹³ and Tlr9 activation has been linked to effects on spatial learning and memory²¹⁴. In my opinion these two candidates merit further research to elucidate the action of GWAS SNP rs7618915.

In order to validate the 4C-seq pipeline, which I established in the Beck lab for the first time, I chose to run a positive control for the 4C-seq experiment on mNSCs (**figure 5.7**). Region 6381 (a region proximal to the Deptor gene with an established 4C-seq profile in the experiments of the Hadjur team) was chosen and identical primers were used for my 4C-seq run. Furthermore, in addition to using positive controls from the same lab and with the same PCR probes for generating the 4C-seq libraries (**figure 5.7**), I also performed a duplicate sequencing run on the same 4C-seq library. Results obtained were identical to the previous sequencing run, showing no differences introduced by the sequencing process. Given the performance of positive controls and the replicate sequencing runs, I am confident that library generation was performed adequately, and that 4C-seq results are robust.

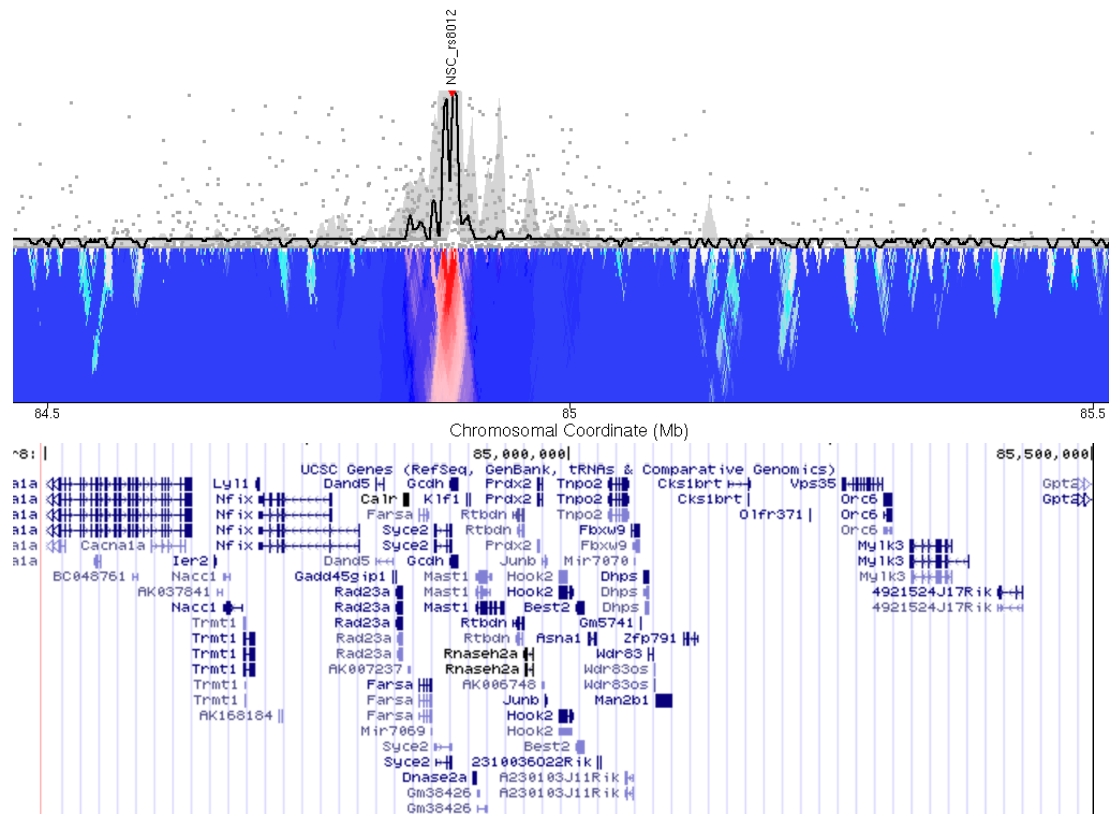


Figure 5.4: 4C-seq results for the *Gcdh* locus in mouse NSCs (homologous to human variant rs8012). The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, <https://genome.ucsc.edu/>) are also indicated in blue. For this region positional information for homologous regions in human is shown in **figure S1** (see Appendices).

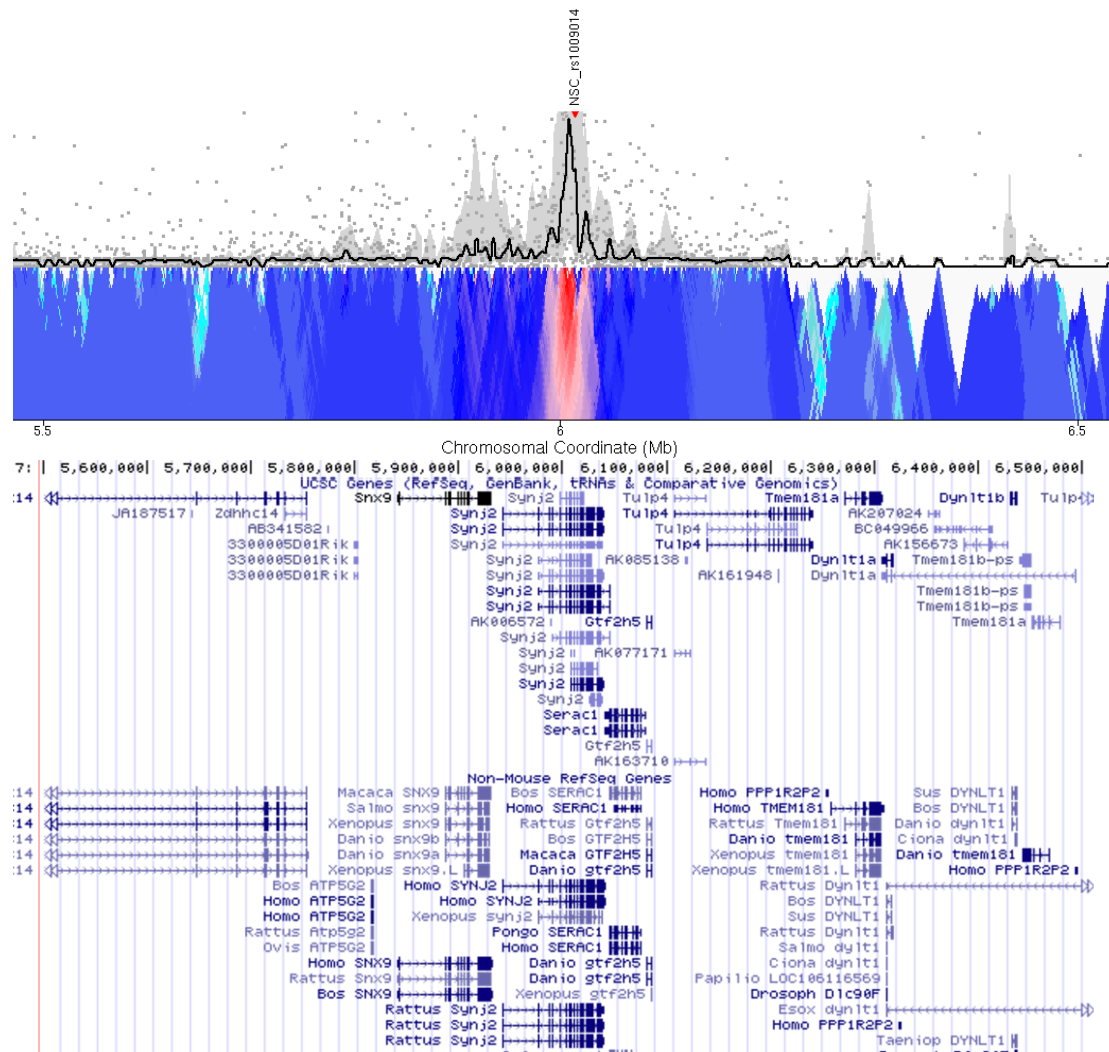


Figure 5.5: 4C-seq results for the Synj2 locus in mouse NSCs (homologous to human variant rs1009014). The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, <https://genome.ucsc.edu/>) are also indicated in blue.

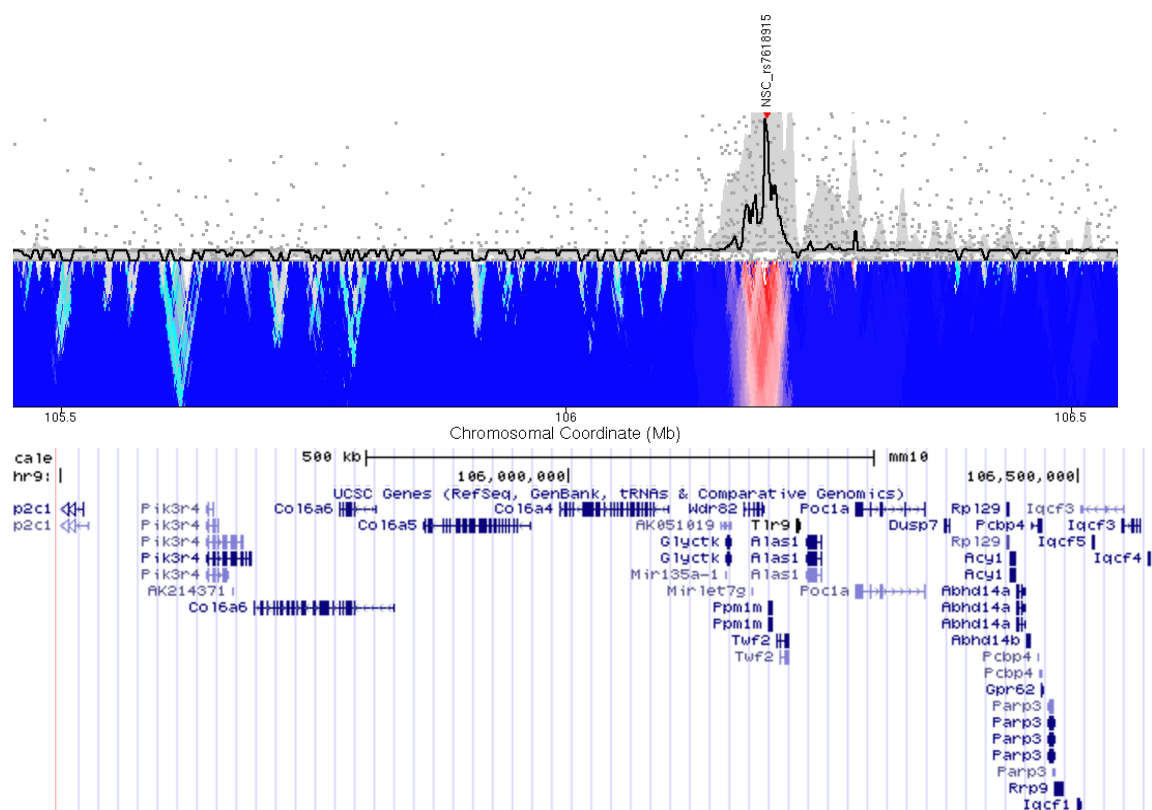


Figure 5.6: 4C-seq results for the *Ppm1m* locus in mouse NSCs (homologous to human SNP rs7618915). The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, <https://genome.ucsc.edu/>) are also indicated in blue. For this region positional information for homologous regions in human is shown in **figure S2** (see Appendices).

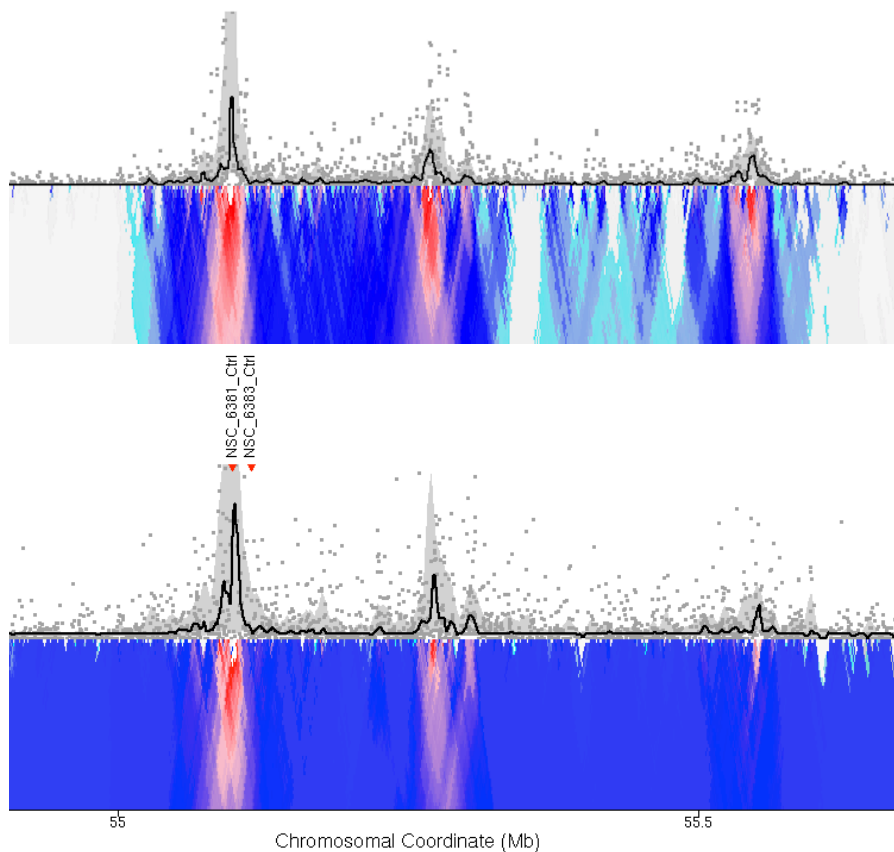


Figure 5.7: 4C-seq positive control. Positive control for a prior 4C-seq experiment (upper panel) compared to the same 4C-seq control in my experiment (lower panel). Peaks are located at the same genomic positions. The PCR primers for generating 4C-seq libraries were identical and obtained from the same group (the Hadjur lab at UCL). The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (left red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the

enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value.

5.3 4C-seq of immune disease loci in sorted blood cells

eFORGE, FORGE2, and other approaches have shown that variants associated with several autoimmune conditions are preferentially located in leucocyte enhancer loci^{81,146}. However, the identification of target genes for such enhancers is challenging as different immune cell types may present radically different interactomes at a local level¹⁴⁷. In addition, even if only one cell type is affected, this cell type may have far-reaching importance in immune regulation through cytokines and other signalling molecules, affecting multiple cell types and pathways¹⁷⁷. Therefore it is important to obtain high-resolution interactome data for different immune cell types in order to gain a precise understanding of the functional consequences of disease-associated variants. I have thus performed 4C-seq across a range of sorted immune cell types for selected candidate loci, with the aim of uncovering cell type-specific interactions. These highly-specific immune interactions may offer key insights into the mechanism of action of disease-associated variants (**figure 5.8**).

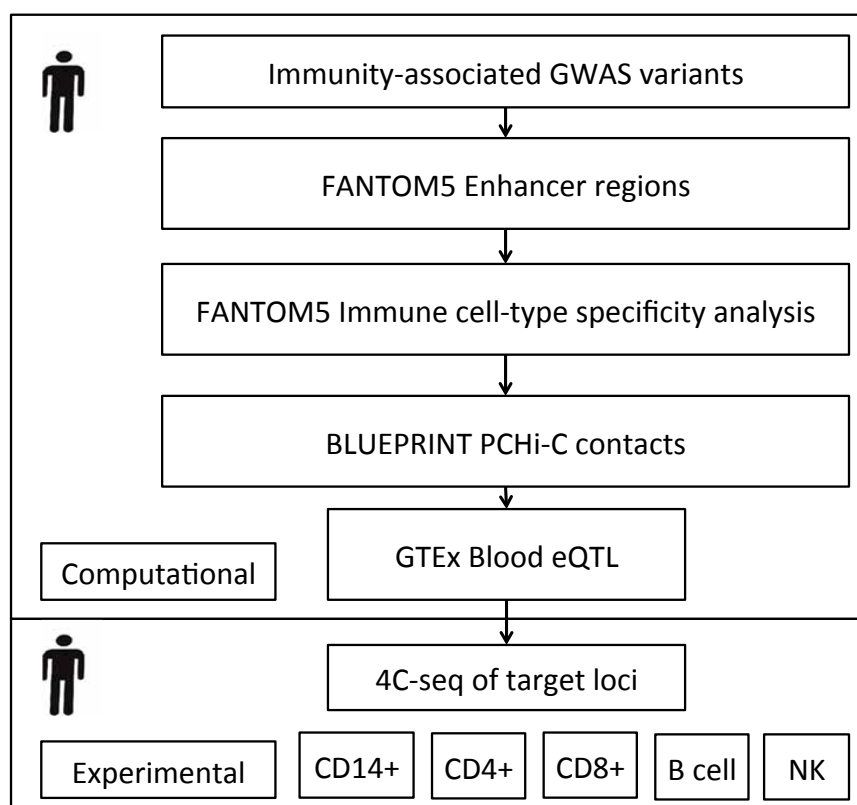


Figure 5.8: Schematic of computational and experimental analysis for human immune GWAS regions. Computational analysis included the intersection of GWAS data with data for enhancers, PCHi-C contacts and eQTL. 4C-seq experiments were then performed in CD14+, CD4+, CD8+, B cells and NK cells for each of the selected regions. The human icon was taken from Odom et al., 2007²⁰¹.

5.3.1 4C-seq analysis of the LRRC8B locus

For the LRRC8B locus I sought to characterise a potential enhancer located within an intron of LRRC8B (hg19 coordinates chr1:90022562-90022945). Variants within this region have been associated with adverse response to paclitaxel/carboplatin chemotherapy (neutropenia/leucopenia).

For this region I can observe a higher number of interactions in CD4+ cells (**figure 5.9**), both within the body of LRRC8B (an enhancer region, marked by an orange chromatin state in GM12878 B lymphocyte-derived cells), the LRRC8B promoter (marked by a red chromatin state across the 9 ENCODE cell lines), regions proximal to the promoter (2 potential enhancers, however, the chromatin state track does not find an enhancer state for these locations in the 9 ENCODE cell lines) and also with LRRC8C. B cells also seem to show interactions with one of the potential enhancers (which also is present in very low signal for CD8+ and NK cells, but not CD14+ cells, this seems to point to the fact that this is a lymphoid-specific enhancer). The absence of interactions with other genes within 2Mb (e.g. LRRC8D) seems to point to LRRC8B and LRRC8C as the interacting genes for this GWAS-associated region. CD14+ cells seem to also present enhancers for LRRC8B (the first CD4+ cell enhancer, and not the lymphoid-specific enhancer). In addition, CD14+ cells seem to also present interactions with LRRC8C.

LRRC8 is a component of volume-regulated anion channels (VRAC). VRACs have been shown to mediate cisplatin/carboplatin uptake, and are also linked to cisplatin/carboplatin resistance²¹⁵. I propose that variants within this enhancer region have a role in VRAC formation in immune cells by affecting LRRC8B and LRRC8C, thus mediating adverse response to paclitaxel/carboplatin chemotherapy and leading to neutropenia/leucopenia. Of note, one candidate variant within the region (rs4658279) was shown to be an eQTL for LRRC8B in GTEx data¹⁵⁵.

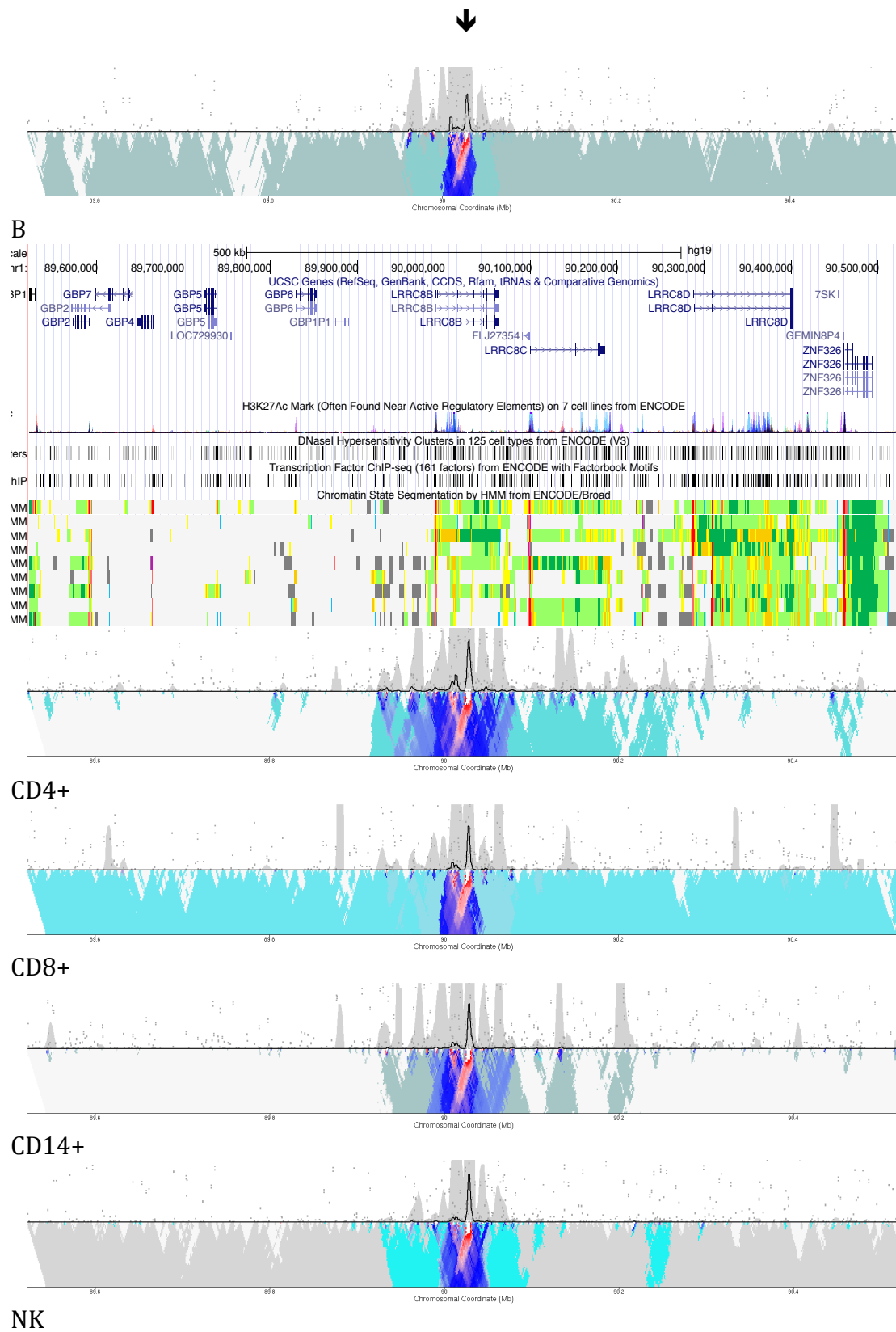


Figure 5.9: 4C-seq results for the LRRC8B locus. Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD8+, CD14+, and NK). For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, <https://genome.ucsc.edu/>) are also indicated in blue. In addition, several epigenetic data tracks are included comprising H3K27ac data across 7 ENCODE cell lines (GM12878, H1-hESC, K562, HUVEC, HSMM, NHEK and NHLF), DNase I hypersensitivity clusters across 125 ENCODE cell types, ENCODE transcription factor ChIP-seq data (161 factors), and Hidden Markov Model (HMM) chromatin state segmentation across 9 ENCODE cell lines (GM12878, H1-hESC, K562, HepG2, HUVEC, HMEC, HSMM, NHEK and NHLF, from top to bottom). Different colours in the HMM track correspond to different chromatin states: Active Promoter (Bright Red), Transcriptional elongation/transition (Dark Green), Polycomb-repressed (Gray), Heterochromatin (Light Gray), Weak transcribed (Light Green), Weak Promoter (Light Red), Strong enhancer (Orange), Inactive/poised Promoter (Purple), Weak/poised enhancer (Yellow), Insulator (Blue).

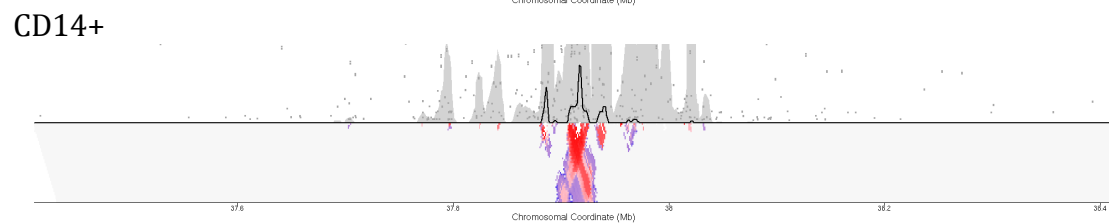
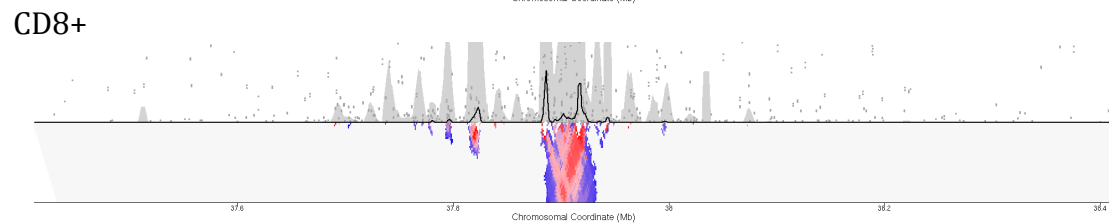
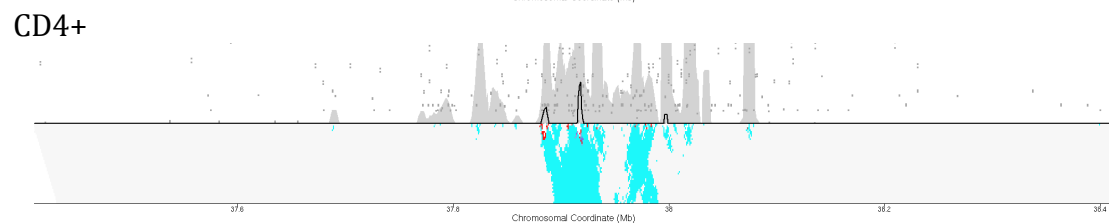
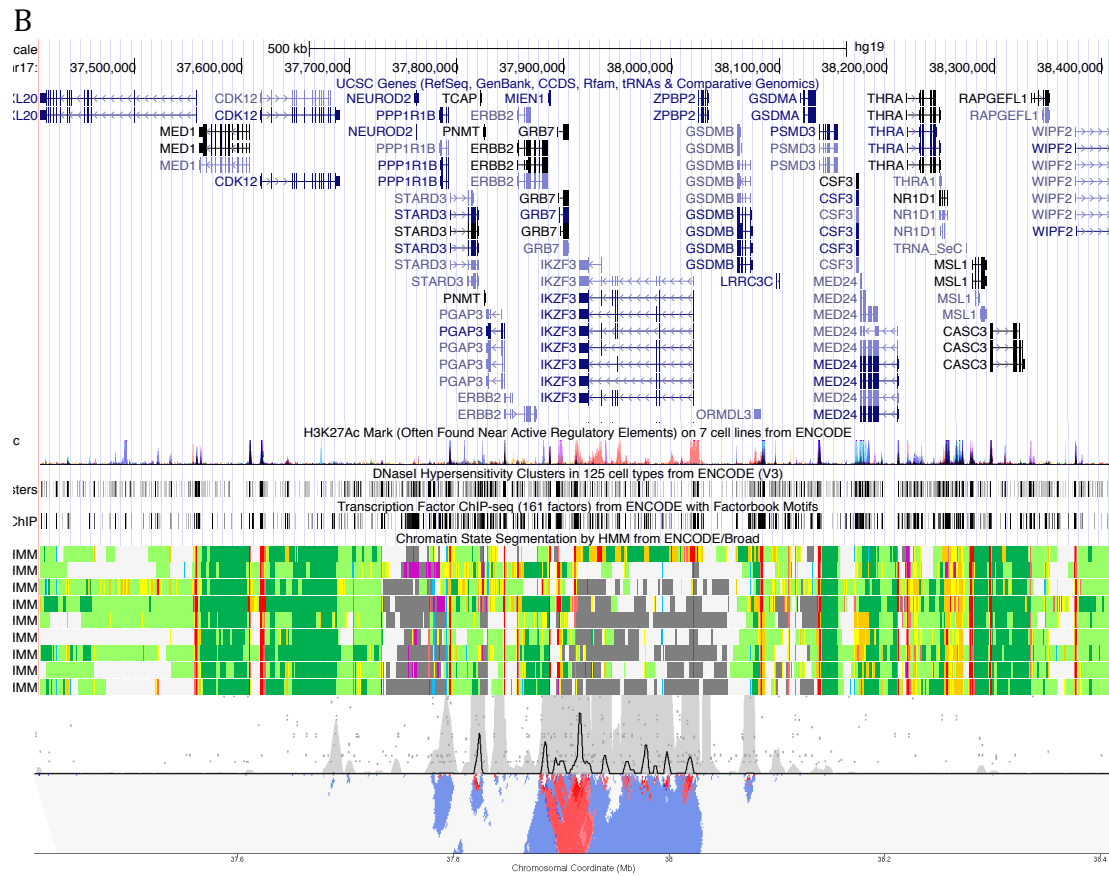
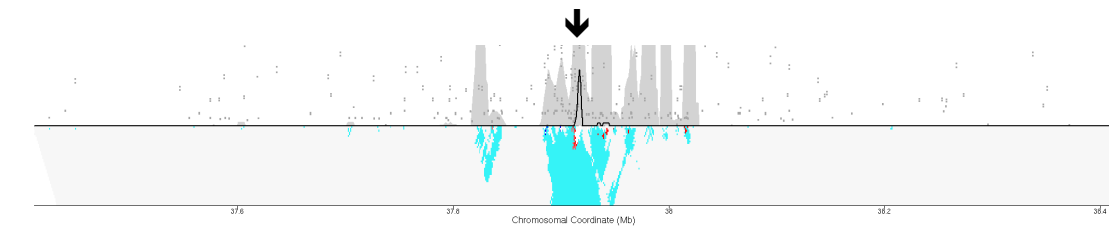
5.3.2 4C-seq analysis of the IKZF3 locus

For the IKZF3 locus I sought to characterise a potential enhancer located between GRB7 and IKZF3 (hg19 coordinates chr17:37912171-37912498). Variants within this region have been associated with inflammatory bowel disease (IBD), specifically ulcerative colitis and Crohn's disease.

For this region I can observe contacts with the promoter of a short isoform of IKZF3 in B cells (marked by a red chromatin state in GM12878 B lymphocyte-derived cells, **figure 5.10**). In CD4⁺ cells I can observe strong contacts with the promoters of several isoforms of IKZF3 (marked by red chromatin states in GM12878 cells), as well as with several parts of the IKZF3 gene body. I can also observe a cluster of interactions with a group of proximal genes, including ERBB2, MIEN1, GRB7 and MIR4728 (cluster 1). This suggests a concerted set of genomic contacts between these genes, which has previously been suggested in studies on the coordinated activation of these genes in breast cancer²¹⁶. In addition, I can observe interactions with the promoter of STARD3. In CD8⁺ cells I can only observe contacts with ERBB2 and IKZF3 (and not STARD3 or the rest of the cluster previously observed in CD4⁺ cells). In CD14⁺ I can observe strong contacts with the short isoform of IKZF3. Contacts are also present with ERBB2 and cluster 1. I can also observe contacts with STARD3 and two new regions, one proximal to NEUROD2 and the other proximal to PPP1R1B. In NK cells I can observe strong interactions with the promoter of the short isoform of IKZF3, and

weak interactions with the longer isoform of IKZF3. Strong interactions are present with cluster 1 and with ERBB2 (and MIR4728). No interactions with STARD3 can be observed.

Experimental results show evidence for an intricate regulation of this critical immune-related region, epitomising the complex mosaic of alternate enhancer contacts in different immune cell types. I propose that variants within this region have a role in IBD through the regulation of IKZF3 expression, leading to effects on downstream immune regulation. This is supported by previous findings that have linked differential expression of IKZF3 with IBD²¹⁷. To further this association I uncover the cell type-specific nature of enhancer-IKZF3 interactions for different IKZF3 isoforms. IKZF3 has also been associated with other autoimmune diseases such as MS¹²⁷. Previous studies on IKZF3 have proposed enhancer activity for a variant in my candidate region (rs12946510)¹²⁷. I have demonstrated that this proposed enhancer loops to IKZF3 and provided the first high-resolution 4C-seq map across several immune cell types for this locus. Importantly, these contacts cannot be observed at the resolution of BLUEPRINT promoter Capture Hi-C data¹⁵⁰.



NK

Figure 5.10: 4C-seq results for the IKZF3 locus. Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD8+, CD14+, and NK). For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, <https://genome.ucsc.edu/>) are also indicated in blue. In addition, several epigenetic data tracks are included comprising H3K27ac data across 7 ENCODE cell lines (GM12878, H1-hESC, K562, HUVEC, HSMM, NHEK and NHLF), DNase I hypersensitivity clusters across 125 ENCODE cell types, ENCODE transcription factor ChIP-seq data (161 factors), and Hidden Markov Model (HMM) chromatin state segmentation across 9 ENCODE cell lines (GM12878, H1-hESC, K562, HepG2, HUVEC, HMEC, HSMM, NHEK and NHLF, from top to bottom). Different colours in the HMM track correspond to different chromatin states: Active Promoter (Bright Red), Transcriptional elongation/transition (Dark Green), Polycomb-repressed (Gray), Heterochromatin (Light Gray), Weak transcribed (Light Green), Weak Promoter (Light Red), Strong enhancer (Orange), Inactive/poised Promoter (Purple), Weak/poised enhancer (Yellow), Insulator (Blue).

5.3.3 4C-seq analysis of the NCF4 locus

For the NCF4 locus I sought to characterise a potential enhancer located in an intron of NCF4 (hg19 coordinates chr22:37258335-37258758). Variants within this region have been associated with atopic dermatitis.

For this region I can observe several interactions with NCF4 in B cells (**figure 5.11**). I can also observe interactions with NCF4 for CD4+ cells, and also with the promoter of one of the isoforms of CSF2RB. FANTOM5 found this region to be a monocyte-specific enhancer. I find interactions to be the strongest in monocytes, covering various points across NCF4 and CSF2RB. In addition, there are monocyte-specific contacts with an intergenic region (potential enhancer) between CSF2RB and TEX33. For NK cells I observe a radically different landscape, with no contacts detected within 2Mb.

I propose that variants within this region have a role in atopic dermatitis through the cell type-specific modulation of NCF4 and CSF2RB in CD14+ cells. NCF4 is a component of phagocyte NADPH-oxidase, which has a key role in myeloid-mediated host defence. CSF2RB is a high-affinity receptor for several cytokines including IL-3, IL-5 and CSF2 (which has a role in monocyte activation²¹⁸). Of note, the NCF4 variant rs4821544 has been associated with other autoimmune conditions such as ileal Crohn's disease²¹⁹. Importantly, I uncover contacts for this region outside NCF4 (specifically with CSF2RB and a proposed enhancer located between CSF2RB and TEX33), with a strong cell type-specific CD14+ signal.

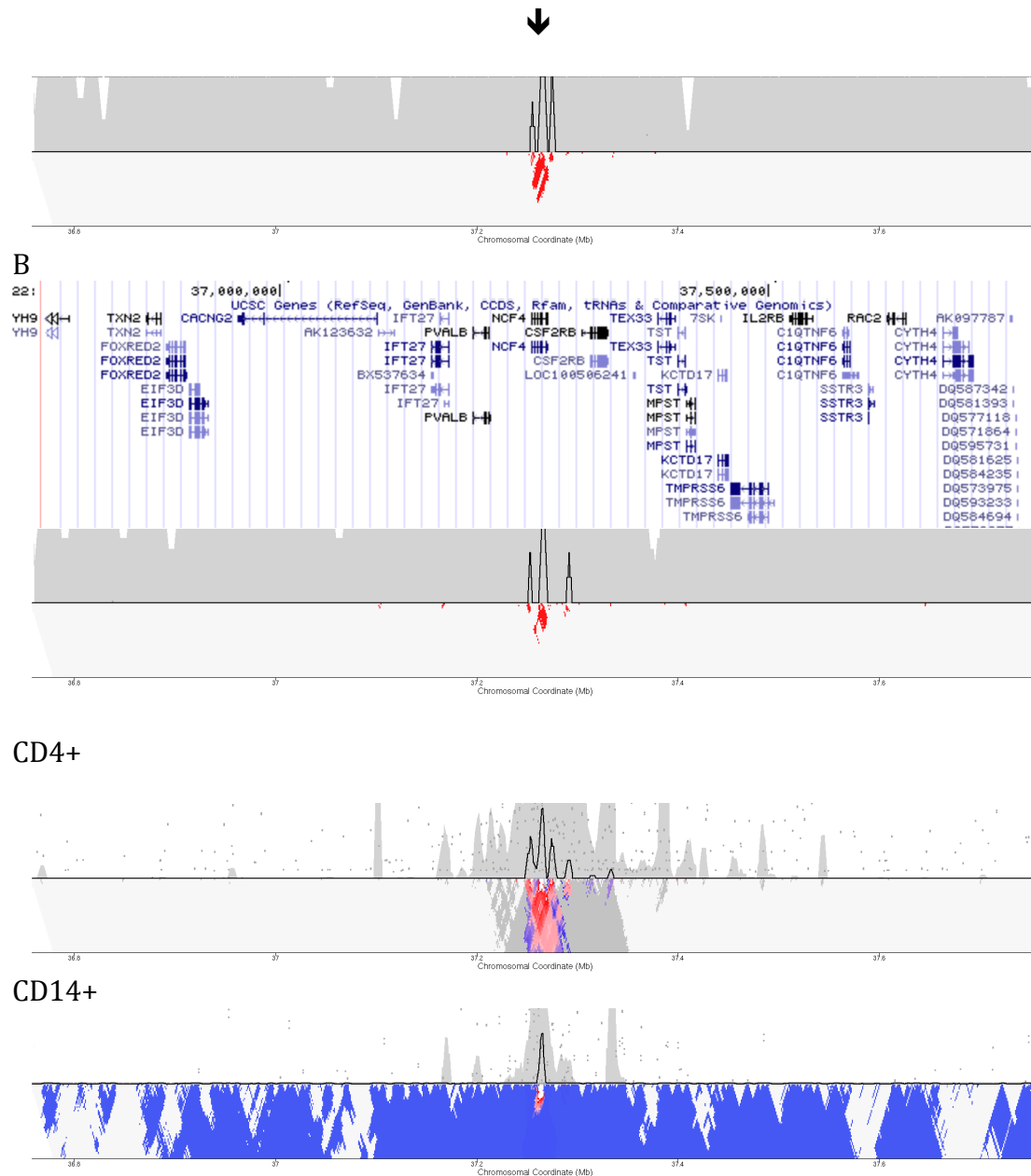


Figure 5.11: 4C-seq results for the NCF4 locus. Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD14+, and NK). For this region CD8+ 4C contacts were excluded from final set due to quality filters. For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and

lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, <https://genome.ucsc.edu/>) are also indicated in blue.

5.3.4 4C-seq analysis of the CSF2 locus

For the CSF2 locus I sought to characterise a potential enhancer located between CSF2 and P4HA2 (hg19 coordinates chr5:131430055-131430830). Variants within this region have been associated with rheumatoid arthritis.

For this region I can observe a highly conserved interaction downstream from the bait for CD4+ cells, CD8+ and CD14+ cells (but not NK or B cells, **figure 5.12**). Other downstream interactions are present for B cells (interaction 1), CD4+ cells (interaction 2, with P4HA2) and CD14+ cells (interaction 3), these interactions are highly cell type-specific. B cells also present an upstream interaction with CSF2. NK cells present a non-interacting landscape with no contacts outside the bait.

I propose that variants within this region have a role in RA through the cell type-specific modulation of P4HA2 in CD4+ cells and CSF2 in B cells. P4HA2 codes for a key enzyme in collagen synthesis. Collagen has been proposed as an autoantigen in Rheumatoid Arthritis (RA)²²⁰ and has been used in trials as a treatment for RA²²¹. P4HA2 is expressed in whole blood and lymphoblastoid cells, among other tissues²²².

CSF2 (also known as granulocyte-macrophage colony-stimulating factor) has been proposed as a target in RA²²³. The cytokine produced by the CSF2 gene can be detected in high quantities in joints with rheumatoid arthritis²²⁴ and drugs are being developed to block CSF2²²⁴. Of note, this particular genomic region has also been associated with inflammatory bowel disease (IBD), juvenile idiopathic arthritis and asthma²²⁵. P4HA2 has also been associated with other autoimmune conditions such as giant cell arteritis²²².

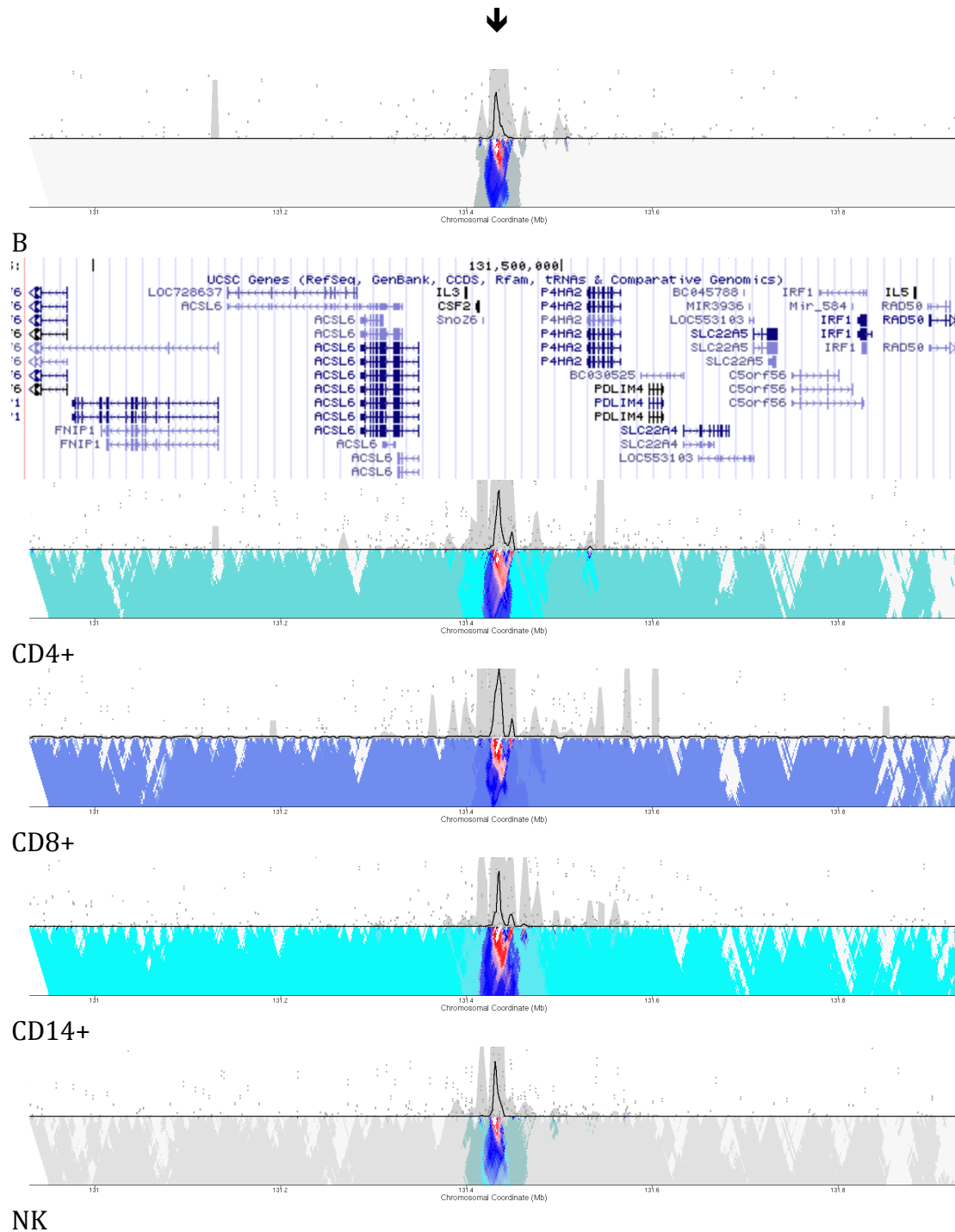


Figure 5.12: 4C-seq results for the CSF2 locus. Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD8+, CD14+, and NK). For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80th and 20th percentiles

of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, <https://genome.ucsc.edu/>) are also indicated in blue.

5.3.5 4C-seq analysis of the CCR6 locus

For the CCR6 locus I sought to characterise a potential enhancer located in an intron of CCR6 (hg19 coordinates chr6:167534229-167534357). Variants within this region have been associated with rheumatoid arthritis.

For this region I can observe clear differences in interactions between cell types (**figure 5.13**). B cells present several interactions within CCR6. In CD4⁺ cells I can observe interactions within CCR6 short isoforms (and GPR31 and the promoter of TCP10L2), and also 6 upstream interactions, some of which overlap the longer CCR6 isoform and the body of the FGFR10P gene. The furthestmost interaction overlaps the RNASET2 gene. In CD8⁺ cells I can observe strong interactions within CCR6, with the promoter of TCP10L2 and with FGFR10P. A weaker interaction with GPR31 can be noted. In CD14⁺ cells I can see some interactions shared with CD4⁺ and CD8⁺ cells, with CCR6, GPR31, TCP10L2 and FGFR10P targeted. Furthermore, the RNASET2 interaction observed in CD4⁺ cells is also

observed here. In addition, a novel interaction with TCP10 only present in CD14+ cells can be observed. In NK cells I can observe interactions with CCR6, TCP10L2 (but not GPR31) and FGFR10P.

SNP rs3093023 (within this region) is an eQTL for RNASET2 in GTEx data and other variants within this region have been associated with RNASET2 expression¹⁵⁶. I detect RNASET2 interactions for CD4+ and CD14+ cells. This suggests that this region is an enhancer that directly interacts with RNASET2 in CD4+ and CD14+ cells. RNASET2 is a ribonuclease that has previously been associated with AI conditions such as Graves' disease²²⁶. Another variant within the interacting area (rs3093024) was found to be a CCR6 eQTL²²⁷. This SNP has also been associated with interleukin-17 quantities in the serum of subjects with RA. rs3093024 has also been associated with other autoimmune conditions such as Graves' and Crohn's diseases²²⁷. CCR6 is a chemokine with a key role in inflammatory response. I propose that variants within this region have a role in rheumatoid arthritis through the modulation of the inflammatory cytokine CCR6 across the 5 immune cell types, and the cell type-specific modulation of RNASET2 in CD4+ and CD14+ cells.

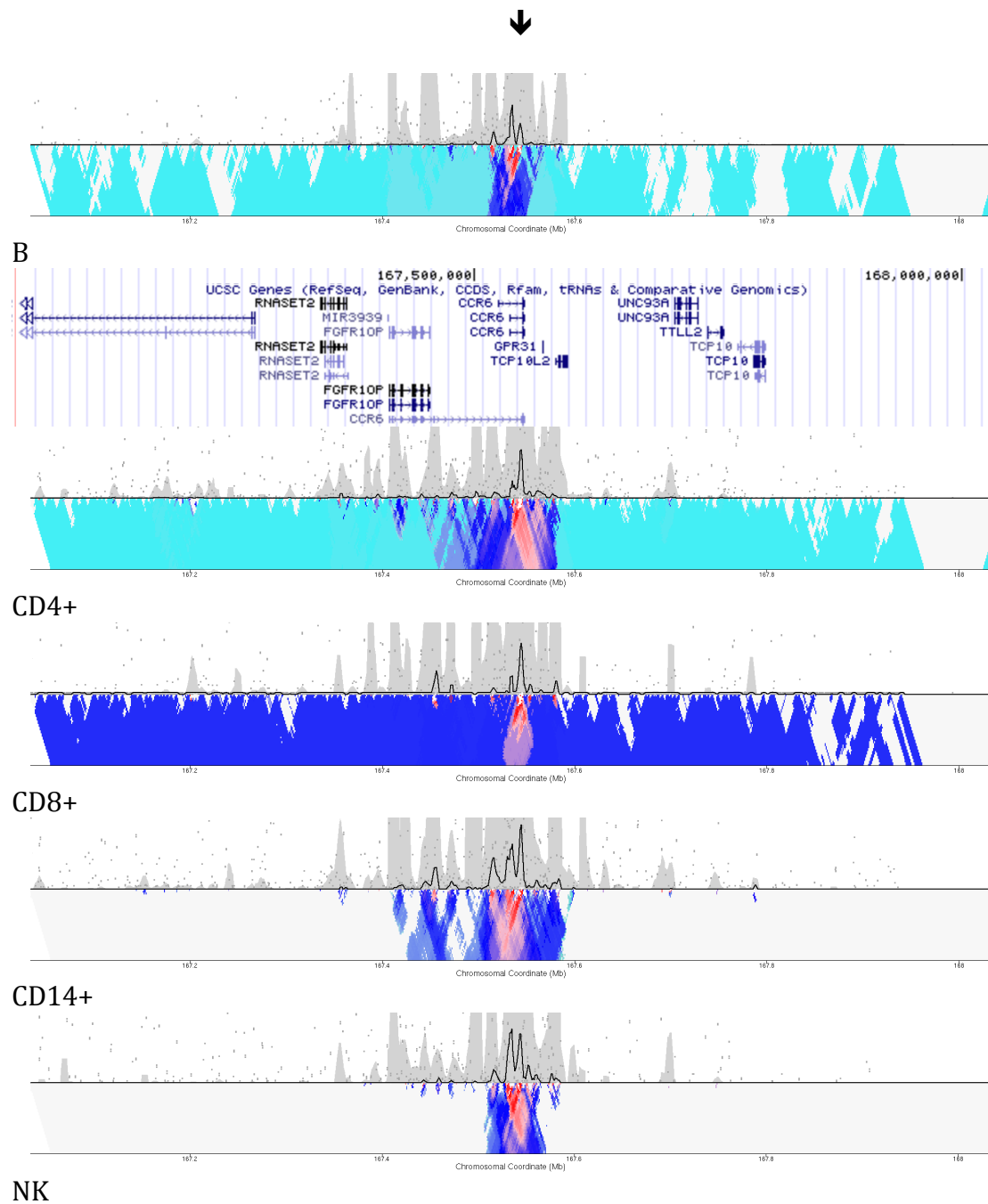


Figure 5.13: 4C-seq results for the CCR6 locus. Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD8+, CD14+, and NK). For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and

lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, <https://genome.ucsc.edu/>) are also indicated in blue.

5.3.6 4C-seq analysis of the TRAF1/C5 locus

For the TRAF1/C5 locus I sought to characterise a potential enhancer located between PHF19 and TRAF1 (hg19 coordinates chr9:123652531-123653088). Variants within this region have been associated with rheumatoid arthritis.

For this region I can observe several interactions with PHF19 and TRAF1 in B cells (**figure 5.14**). In CD4⁺ cells I can observe interactions with TRAF1, PHF19, C5 and PSMD5. In CD8⁺ cells I can observe similar interactions to B cells, within PHF19 and TRAF1. In CD14⁺ cells I can observe similar interactions to CD4⁺ cells, that is, with TRAF1, PHF19, C5 and PSMD5. In addition, a weak interaction (potentially an enhancer) can be observed upstream of FBXW2. In NK cells I can only observe interactions with TRAF1 and PHF19, in a similar way to B cells. To summarise, I can observe a few local interactions for B cells, CD8⁺ cells and NK cells, and radically different interaction maps for CD4⁺ and CD14⁺ cells, with more distal genes involved.

Despite the fact that interactions are detected for PHF19 and PSMD5, the main biological evidence supports a role for TRAF1 and C5 in this area²²⁸. TRAF1 is a negative regulator of inflammation with a role in rheumatic diseases²²⁹. C5 is a crucial component of the complement system in innate immunity. C5 has been targeted successfully in treatments for autoimmune disease models such as experimental anti-phospholipid antibody syndrome²³⁰. I propose that variants within this region have a role in rheumatoid arthritis through the modulation of TRAF1 across the 5 immune cell types, and the cell type-specific modulation of C5 in CD4+ and CD14+ cells. Of note, variants in this region are eQTL for C5, PSMD5-AS1 and TRAF1 in several immune cell types¹⁵⁶.

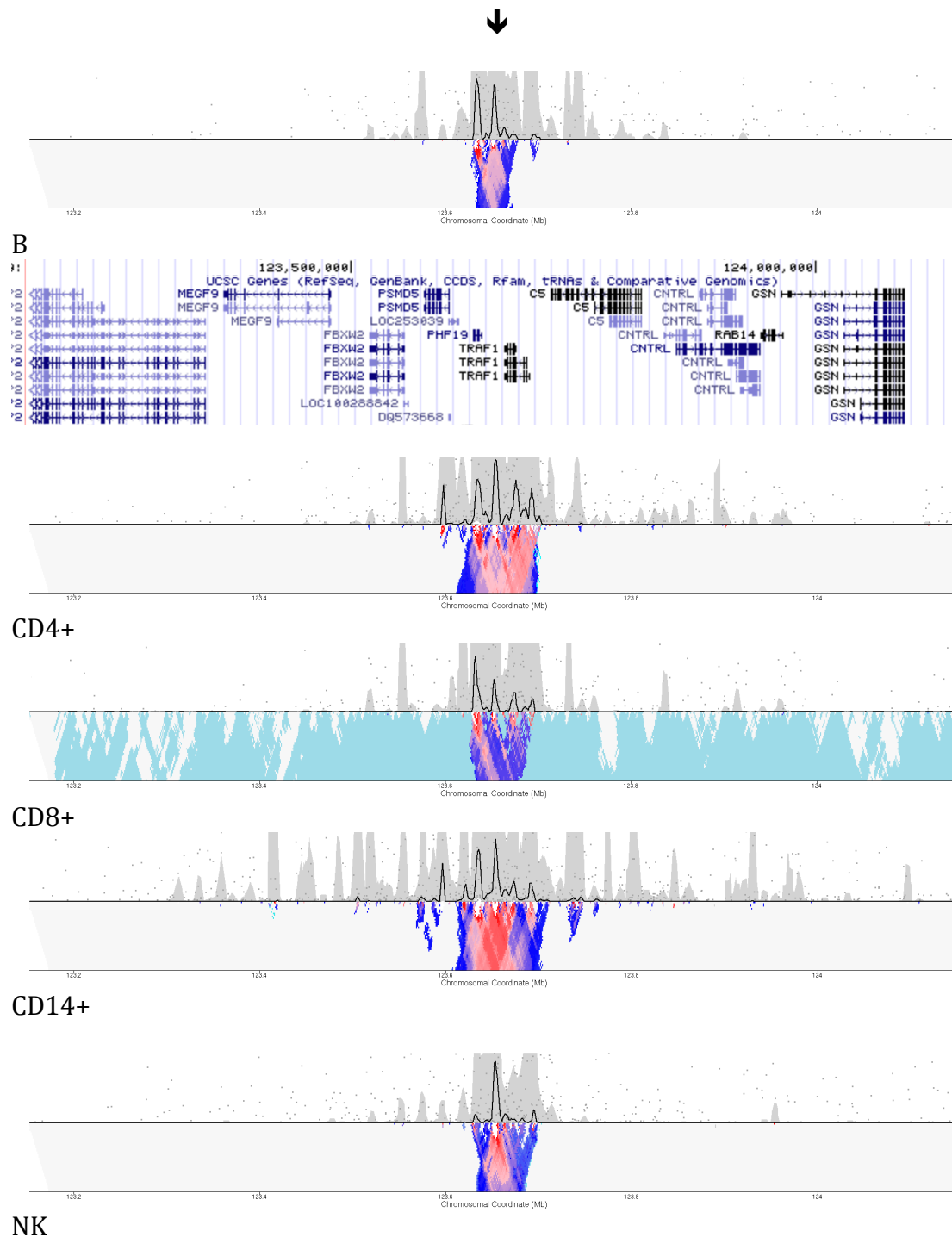


Figure 5.14: 4C-seq results for the TRAF1/C5 locus. Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD8+, CD14+, and NK). For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80th and 20th percentiles

of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, <https://genome.ucsc.edu/>) are also indicated in blue.

In summary, I have applied 4C-seq to disease-associated regions across several immune cell types, uncovering a range of cell type-specific interactions, the most notable of which is a previously unreported IKZF3 interaction for a region containing IBD-associated SNP rs12946510. IKZF3 is a druggable target, and anti-IKZF3 drug lenalidomide (a variant of thalidomide) has already been developed for multiple myeloma^{231,232,233}. Evidence of repurposing of thalidomide variants for IBD is inconclusive, and despite adverse effects there is a scope for thalidomide application in certain cases of IBD²³⁴. Further research and target validation efforts are necessary, yet the detection of IKZF3 as a target gene is an important step forward in view of uncovering druggable IBD-associated pathways.

In conclusion, the associations uncovered by my approach merit further experimental study. I have highlighted cell type-specific interactions across a range of cell types, generating the first 4C-seq maps for an important immune

cell type (NK cells), and revealing a range of novel interactions for non-protein-coding regions of the genome.

6 Discussion

6.1 eFORGE

6.1.1 eFORGE aims and context

One of the major objectives of performing an EWAS is to obtain novel biological information about a certain disease or trait. When dealing with large amounts of complex data the analysis of the results of the study (in order to obtain this novel biological information) is typically performed using bioinformatics methods. Currently, many methods used to analyse EWAS results are gene-centric, that is, they are based on gene annotations (or relevant proteins). Standard examples of this approach include Gene Ontology (GO) or Kyoto Encyclopaedia of Genes and Genomes (KEGG) pathway analysis^{39,41}. Many EWAS obtain gene lists from differentially methylated regions (e.g. differentially methylated promoters). These lists are then analysed with gene ontology (GO) algorithms to detect over-represented annotation groups⁸. Such methods are the general bioinformatics analysis choice, not only for EWAS, but also for other fields including GWAS and gene expression studies^{235,236}.

Intergenic probes make up a quarter of the 450k array¹⁴ and a higher proportion of the 850k array²³⁷. Many intergenic EWAS DMPs do not easily map to target genes, and are therefore discarded when performing GO analysis or similar gene-centric approaches. In addition, gene-centric analyses typically result in weaker enrichment scores than those obtained from gene expression studies⁸. Alternative approaches

are thus required to allow for a more appropriate method to analyse EWAS results, using information for all differentially methylated positions. A range of EWAS analysis tools have been developed that use genome-wide features⁸ (and thus are distinct from gene-centric approaches). It is in this context that eFORGE has been designed, bringing forwards the FORGE genome-wide functional overlap analysis approach to EWAS data.

When analysing genome-wide data for a given cell type one can focus on different epigenetic marks: histone marks (e.g. the activating H3K4me3, or H3K27me3, which is found in facultatively repressed genes⁷¹), DNase I Hypersensitive Sites (which signal for classical cis-regulatory elements¹⁵⁹) and DNA methylation (which is associated with the regulation of gene expression²³⁸). In the FORGE project (the analysis method designed for GWAS), the focus was on DNase I Hypersensitive Sites (or DHSs), as DHSs mark all major classes of cis-regulatory elements¹⁹⁹. The FORGE tool thus performs a functional overlap analysis (using a processed form of DNase I cleavage data known as DNase I hotspots¹⁵⁸) with top GWAS SNPs. The idea behind this functional overlap analysis is that top GWAS SNPs will tend to overlap with more DNase I hotspots in tissues where the disease is relevant, as a portion of the non-protein-coding GWAS SNPs will be acting through classical cis-regulatory elements, regulating gene expression in target tissues. For gene expression to be affected in target tissues, these cis-regulatory elements must be active in those tissues. In addition, it is expected that there will be a higher number of top GWAS SNPs overlapping DNase I hotspots in those tissues when compared with tissues with no disease relevance. The proposed underlying phenomenon is that top GWAS SNPs are

enriched for tissue-specific cis-regulatory elements in relevant tissues for the trait or disease under study.

The analysis approach performed by FORGE avoids the aforementioned gene-centric bias and points to a link between particular diseases and tissue-specific DNase I hotspot enrichment. Using epigenomic data to obtain disease-associated tissue-specific enrichment signals is one of the aims of ENCODE, Epigenomics Roadmap and BLUEPRINT data generation. The objective of the eFORGE project was to develop a bioinformatics tool that analyses top EWAS hits and returns tissue-specific enrichment results for Epigenomics Roadmap, ENCODE and BLUEPRINT tissues. For my eFORGE project hypothesis, I proposed that the tissue-specific signal that is observed for GWAS in FORGE would also occur for EWAS as a systematic and consistent phenomenon. I expected tDMR and EWAS regions to present tissue-specific enrichment for DNase I Hotspots in the same way as GWAS regions do, thus highlighting relevant tissues and cell types for EWAS. The underlying idea behind this is that a proportion of top EWAS DMPs reflects alterations in cis-regulatory elements in relevant tissues (some EWAS arrays, such as the 850k array, have been purposefully designed to capture this phenomenon²³⁷). If this is the case, in a similar way to GWAS I expect a higher number of overlaps between EWAS DMPs and DNase I hotspots in relevant tissues when compared with tissues with no disease relevance. In a similar way, the rationale behind expecting tDMPs to be enriched for tissue-specific DNase I hotspots is based on the notion that a proportion of tDMPs are formed due to tissue-specific cis-regulatory processes, and are therefore expected to overlap tissue-specific DNase I hotspots (a proxy for active regions in a given tissue).

Early in the project it became clear that one of the main differences in eFORGE (in relation to FORGE) would be the background used in the statistical test. The background in FORGE consists of SNPs from two different types of arrays (GWAS typing arrays and the Illumina_HumanOmni2.5 array). The GWAS typing arrays include a variety of arrays listed in the Methods section, including, among others, the Affymetrix GeneChip 500K array and the Illumina Cardio-MetaboChip array. In the case of EWAS there are a variety of different technologies that can be applied⁸. When choosing the background for eFORGE I had to choose among the following approaches:

- Illumina GoldenGate assay
- Illumina 27k array
- Illumina 450k array
- RRBS (Reduced Representation Bisulphite Sequencing)
- MeDIP-chip
- MeDIP-seq

Among these choices I selected the Illumina 450k array, due to several reasons: it is a widely used method⁸, which allows for a high level of reproducibility between studies¹⁴. In addition, DMPs are limited to array probes, which simplifies background generation. Using an array for eFORGE background design also makes the approach conceptually more similar to FORGE analysis. In addition, as 94% of 27k array probes

are present on the 450k array, a 27k array background can be generated as a subset of a 450k array background, allowing for the analysis of a higher number of EWAS.

Going forward, the eFORGE approach could be extended to analyse EWAS performed using other technologies. One of the most obvious candidates in terms of current numbers would be MeDIP-seq. MeDIP-seq is not array-based but sequencing-based, in a similar way to ChIP-seq²³⁹. This fact means that MeDIP-seq allows for genome-wide discovery of differentially methylated regions (DMRs). However, MeDIP-seq presents some disadvantages, including a bias for regions with a high CG content²⁴⁰. Considering the limited nature of Illumina DNAm arrays, and the CG-skewed data obtained with MeDIP-seq, another alternative would be Whole-Genome Bisulphite Sequencing (WGBS). With the current decrease in the costs of sequencing it is not unlikely that WGBS will be the method of choice for EWAS in the future.

6.1.2 eFORGE findings and applications

Given the implementation of FORGE and FORGE2, I sought to address the following question: does the field of epigenome-wide association studies (EWAS) present this same phenomenon of tissue-specific enrichment for histone marks and DNase I hotspots? The answer obtained after analysing many EWAS is yes, the field of EWAS

presents the same phenomenon of tissue-specific enrichment for DNase I hotspots and histone marks (see Results).

It is important to highlight that the correct interpretation of this tissue-specific enrichment can vary, as EWAS are subject to many confounders, including cell composition effects as mentioned previously¹⁸. Given that these cell composition effects are driven by tissue-specific DNAm, and that tissue-specific DNAm and tissue-specific DNase I hotspots tend to co-localise, eFORGE has also demonstrated the capability of detecting cell composition effects. This adds a novel approach for the detection of cell composition effects, an area of vital importance to the field of EWAS.

eFORGE has been applied across a range of diseases and traits^{146,241,242,243}. Specifically for three autoimmune diseases (rheumatoid arthritis, systemic lupus erythematosus and Sjögren's syndrome) eFORGE has identified target immune cell types. eFORGE has also detected an immune component in a multiple sclerosis EWAS performed on non-pathological brain tissue. Despite a non-pathological macroscopic tissue aspect, it may well be the case that subtle changes in immune activation (reflected in DNAm differences) were an important component detected by the study. "Priming" or "pre-activation" changes in immune cells could be present in a tissue with a non-pathological macroscopic aspect. These subtle changes could however be detected by the EWAS array and be reflected in the change in DNAm status for a specific genomic site. In this analysis eFORGE supports the notion of immune activation underlying EWAS signal.

eFORGE has also been applied to detect cell composition effects, as is demonstrated by the study of granulocyte/lymphocyte ratio changes in the immune response to ovarian cancer. From a theoretical perspective, I consider an unexpectedly cell type-specific signal for a whole blood study as a sign of cell composition effects at play. However, it is important to also consider disease aetiology and factors such as probe filtration. I have also analysed cell composition corrected whole blood studies, such as the Liu et al. EWAS on rheumatoid arthritis¹⁷². For this analysis I have found less striking cell type-specific enrichment patterns, which are more consistent with the expected aetiology of this complex disease. My analyses show that eFORGE results for EWAS driven by cell composition effects differ from those for studies showing true cell type-specific effects, and I report these analyses as examples for future applications of eFORGE.

Cancer EWAS are characterised by radical changes in DNAm. Some of these changes may be involved in the silencing of tumour suppressor genes and the activation of oncogenes, for example by altered CpG island DNAm patterns. From a genome-wide perspective many of these changes form part of cancer epigenetic reprogramming that transforms cells to a state in which self-renewal and other cancer hallmarks are phenotypically present. Previous observations report cases in which the cancer epigenomic signature leads cells to a more “stem cell-like” state²⁴⁴. In cancer, to achieve stem cell-like phenotypes such as pluripotency or dedifferentiation a cell may retrace its position in the developmental gradient of Waddington’s landscape (see Introduction) and reprogram itself to a “higher”, more *primaeval* state. As I have

noted in the chapter on DNase I hypersensitivity, for differentiated cells around 1/3 of DHSs are from a primaeval state, 1/3 are from a state of intermediate differentiation and 1/3 are from a fully differentiated state. In cancer epigenetic reprogramming a cell may reactivate many of the “primaeval state” DHSs, in a similar way to epigenetic reprogramming in iPSCs. This reprogramming will have a reflection on both DHSs and DNAm levels. Prior studies on DHSs and cancer have shown a messy trend, in which cells do not neatly regress to a primaeval state but “highjack” DHSs and cellular machinery to acquire primaeval state-associated traits among other phenotypes⁶³. In contrast to this, I report a clear stem cell signal across 5 cancer EWAS (see Results).

Due to the design of cancer EWAS, DNAm differences found are a reflection of common features between the many different cancer samples studied. Each individual cancer sample may show a mixture of functional and non-functional epigenetic alterations. However, when studying events across cancer samples, there may be a selection for the purely functional changes that ensure cell survival amidst the “noise” of arbitrary genomic events. Therefore, it is perhaps not surprising that I observe a clear stem cell signal across cancer EWAS, because this epigenetic reprogramming to an earlier stage is a key element to ensure cell survival (and is associated with the key feature of self-renewal, as also observed in iPSCs). I thus observe a more “purified” signal in my meta-analysis of these studies, which can be typical for an analysis of common trends originating as a consensus from many samples.

In eFORGE analyses of cancer EWAS I can therefore observe a trend that supports notions of cancer epigenetic reprogramming and the reactivation of stem cell DHSs, in agreement with previous research⁶³. Further research is needed to uncover how this is achieved mechanistically. For now, I can state that this finding, surprising in its clarity, merits a closer look in regards to the underlying mechanisms.

Certain analyses have observed high probe signal across all tissues when analysing certain probe sets with eFORGE²⁴¹. This is likely to indicate that those sites show high constitutive activity (indicated by DNase I hotspot overlap) compared to background probes that are similar for MAF, distance to TSS and GC content.

eFORGE has shown many applications. The first application includes analysis of EWAS data to identify disease-relevant cell types from a heterogeneous tissue (e.g. whole blood). In addition, eFORGE can detect cell composition effects in EWAS data. In this context, eFORGE provides an approach that is complementary to DNAm-based tools. eFORGE detection of cell composition effects is based on the assumption that tissue or cell type-specific differentially methylated positions (tDMPs or cDMPs) are driving the confounding signal. Because eFORGE can detect any strong influence of tDMPs in an EWAS probe list, eFORGE is capable of detecting cell composition effects, as has been shown in the aforementioned study on ovarian cancer²⁹. It is important to add that eFORGE is provided as a web tool, facilitating analysis in a way not currently offered by competing methods^{49,48,52,51}.

eFORGE signal can also be used as a measure of quality control in tissue- and cell type-specific DNAm studies (as mentioned above, tDMP and cDMP sets can be traced independently back to their original tissue or cell type using eFORGE).

eFORGE also suggests that DNase I hotspot data can be used inversely to predict tissue-specific differentially methylated positions, supporting other approaches that focus on tDMP prediction using other layers of epigenetic data⁸⁶. eFORGE has also revealed insights into mechanistic understanding of disease- and trait-associated regions (e.g. stem cell targets identified for cancer EWAS), and serves a key role in prioritising regions overlapping cell type-specific DNase I hotspots/histone mark peaks to pave the way for further experimental analyses (e.g. 4C-seq, or modifications of the epigenome through CRISPR/Cas9-based methods²⁴⁵). The design of eFORGE is geared towards downstream experimental follow-up of EWAS DMPs. For this, eFORGE provides tables detailing cell type-specific DNase I hotspot/histone mark peak overlap results for each DMP analysed. Finally, eFORGE can prioritise histone marks relevant to disease-associated probe sets for further computational analyses (e.g. TFBS analysis following a histone mark-specific signal for a particular EWAS).

Technical improvements introduced by eFORGE (when compared to FORGE) include faster scaling of the binomial test by the use of logarithms instead of decimal digits, the introduction of histone mark datasets, faster scaling of bitstring-generating code, the redesign of the SQLite database to facilitate the addition of new datasets as tables and the open design of the database, which can now be generated from

scratch simply by the execution of a single script. These changes, and many other minor code additions, have been placed on an open repository as a resource for the bioinformatics community. In addition, a study has been performed on software reproducibility taking eFORGE as an example²⁴⁶. Other advances made possible by eFORGE design include the development of the first EWAS catalogue, the eFORGE catalogue (as, unlike GWAS, EWAS are not yet listed on a centralised registry), and contributions to the design of FORGE2. The development and methods of eFORGE prepared the stage for the development of FORGE2, which in turn has provided further insights into the genomics of diseases. In an example of reproducibility and code reutilisation, eFORGE code improvements (such as the faster scaling of the binomial test) have been passed on to the FORGE2 tool.

6.1.3 Considerations and limitations of eFORGE

As DNA hypermethylation and hypomethylation are both present in EWAS, two different top probe sets may be analysed by eFORGE. I recommend that DNA hypo- and hypermethylated probes be analysed separately when possible in order to improve functional interpretation of EWAS results. In addition, in order to allow for a complementary analysis to enrichment I developed depletion analysis for eFORGE.

It is important to add that eFORGE can only provide analysis for cell types for which there are data available. These data may be processed open chromatin data (e.g. from DNase-seq or ATAC-seq), or processed histone mark data in broadPeak or

similar formats. Guidelines and code have been provided to add data in these formats to a local eFORGE database. eFORGE currently contains 454 samples in its online database. However, many cell types are still not present in the database and to analyse these I have had to turn to complementary datasets, such as in the aforementioned analysis of microglial enhancers on sites from the MS EWAS by Huynh et al.¹⁷⁴.

6.1.4 Outlook on eFORGE

Despite the fact that DNase I hotspot-based tools have been implemented for the field of GWAS^{139,137,138,247}, eFORGE represents a dramatic conceptual shift, focusing on a different field, the field of EWAS, which is noted for radical differences (such as cell composition effects)¹⁸. Before eFORGE it had not been demonstrated in a systematic way that EWAS DMPs showed enrichment for tissue-specific DNase I hotspots. However, I have shown that tissue-specific DNase I hotspot enrichment is a systematic phenomenon across the EWAS field. After demonstrating this for an unbiased selection of EWAS, I have examined particular examples in more depth in order to unveil functional mechanistic insights, leading to differing biological interpretation for a published study on multiple sclerosis and the characterisation of a cell type-specific signal for ovarian cancer, in addition to demonstrating the detection of cell composition effects, and the observation of a stem cell signature across a range of cancer EWAS. Furthermore, analyses of blood-based EWAS for

autoimmune diseases have highlighted immune-specific regulatory elements for further study.

With eFORGE I have developed a tool that does not only highlight the EWAS probes that are likely to be functional in a cell type- or tissue-specific context, but also shows particular trends for sets of probes associated with different traits and diseases. The development of this tool makes the automated cell type-specific enrichment analysis of any array-based EWAS possible. This constitutes a useful addition to the current limited toolbox for the interpretation and analysis of EWAS data. From now onwards, any scientist performing an EWAS can obtain cell type-specific enrichment results in minutes, from an online tool that has been rigorously tested for performance. As mentioned previously, false positives are highly improbable, eFORGE results are highly reproducible and the execution time has been dramatically reduced from previous code. The lack of requirement of a local installation (due to the availability of a web tool) is an even stronger support for the already high reproducibility standard of this particular software. In addition, by providing eFORGE results as part of a first-in-field catalogue, the eFORGE catalogue, I have made study-specific graphs and tables publicly available for further functional exploration.

As a long-term product of this research I expect an increase in the understanding of the epigenetics of complex traits and diseases, and more specifically in the understanding and interpretation of EWAS results. eFORGE analyses press the case for the generation of additional epigenetic data for all main cell types. These datasets,

coupled with improved computational analysis of EWAS regions, will drive understanding of the mechanisms underlying the observed epigenetic changes, potentially leading to an improvement in the diagnosis and treatment of complex diseases.

6.2 FORGE2

6.2.1 FORGE2 aims and context

GWAS represent one of the biggest advances in the last decade for the genetic study of disease, highlighting scores of SNPs associated with a variety of disorders, including type 2 diabetes and schizophrenia^{123,248}. However, the interpretation of these associations remains challenging, especially in view of uncovering the functional genomics underlying disease mechanism^{249,250}. Recent breakthroughs in genomic methods (including CRISPR/Cas9²⁵¹ and 3C-related technologies¹¹¹) have added momentum to this field of research, and aid the process of linking genetics and disease¹³⁰. However, the community has not yet completely explored the possibilities offered by large-scale epigenomics data gathered through the ENCODE¹⁸³, BLUEPRINT¹⁸⁴ and Epigenomics Roadmap initiatives⁸¹. Mapping GWAS-associated potential regulatory elements to the relevant tissues through functional overlap analysis has been approached using DNase I hotspot data^{139,198}, but not recent genome-wide histone mark data obtained through ChIP-seq⁸¹. To further the insights obtained through DNase I hotspot data it is necessary to separate the

different classes of genomic elements (e.g. promoters, enhancers) driving the observed GWAS-DNase I hotspot enrichment. This problem can be approached using histone mark data. I have developed FORGE2 to explore this rich dataset.

6.2.2 FORGE2 findings

Applying FORGE2 to disease-associated SNPs as described in the GWAS catalogue¹²³ yielded potential insights into disease aetiology. FORGE2 analysis includes five histone marks: H3K4me1, H3K4me3, H3K27me3, H3K36me3 and H3K9me3. I found the first four marks to be informative in the study of GWAS results. FORGE2 histone mark analysis proved an important complement to FORGE DNase I hotspot analysis for a variety of diseases, revealing previously undetected associations of particular cell types with specific diseases. In addition, many histone mark analysis results confirmed previous FORGE findings. This is true especially for certain autoimmune diseases, for which blood association was detected in H3K4me1, H3K4me3 and H3K36me3. Heart phenotypes such as QT interval and PR interval, which in FORGE show enrichment for heart tissue, also show enrichment for heart tissue in H3K4me1 and H3K4me3 analysis. These repeated findings add support to the FORGE2 histone mark analysis approach.

In addition to confirming previous findings, FORGE2 sheds light on novel disease-cell type associations. Novel associations of cell types with specific diseases include, for H3K4me1, the association of T cells with Graves' disease and with vein graft stenosis in coronary artery bypass grafting, the association of haematopoietic stem cells with

mean platelet volume and F cell distribution, and the association of stem cells with attention deficit hyperactivity disorder. In addition, I observe a CD14+/haematopoietic stem cell signature for Alzheimer's disease-associated SNPs. I also report improvements on cell type-specific enrichment patterns previously observed in FORGE. Using data for H3K4me1 (the most informative mark I have tested) I can find a stronger cell type-specific signal for shared systemic lupus erythematosus/systemic sclerosis SNPs. For this GWAS SNP set I observe an enrichment that is specific to natural killer cells. These findings support the relevance of studying the enhancer-associated mark H3K4me1 in the analysis of GWAS findings.

For H3K4me3, a promoter-associated mark¹⁸³, I find a novel enrichment in stomach tissue for SNPs associated with food allergy. In addition I also confirm findings for mean platelet volume also discovered in the H3K4me1 analysis. In total I can confirm 9 FORGE findings through H3K4me3 analysis. H3K4me3 enrichments allow for a more straightforward interpretation than DNase I hotspot enrichments, as many individual overlaps can be linked to promoter elements and a network can be built with the corresponding genes.

In my analysis of H3K27me3 regions, I find some remarkable associations including, among others, a B cell signal for SNPs associated with C reactive protein levels and a haematopoietic stem cell enrichment for thyroid cancer-associated SNPs. These findings may have an association with Polycomb- repressed regions, and a study of these may yield further insight into the reported associations. For H3K36me3, in

addition to confirming several FORGE findings, I detect a high general enrichment for SNPs associated with blood metabolite levels, and a skin- and blood-specific enrichment for SNPs associated with psoriasis. These findings highlight the relevance of transcription-associated regions and could also provide a link between alternative splicing and GWAS SNP mechanism.

For many GWAS datasets I observe an enrichment signature in relevant tissues for H3K4me1 broadPeaks, in agreement with previous findings that propose that many GWAS SNPs act through enhancer disruption ²⁵². However, for some diseases I detect an enrichment for other histone mark broadPeaks. It is possible that different complex diseases act through different classes of regulatory elements, potentially meriting a distinction between diseases that have a genetic component mainly mediated through enhancers, or “enhancer diseases”, and diseases that are mainly mediated through other genomic elements (such as promoters, transcription-related regions, and Polycomb-repressed regions). FORGE2 allows for the separate analysis of 5 histone mark datasets enriched for different classes of regulatory elements, focusing analysis of cell type-specific enrichment signal to this finer level. A potential next step would be to generate a FORGE-like tool using chromatin state annotations derived from ChromHMM or ChromImpute to explore these enrichments further^{86,253}.

6.2.3 Considerations on FORGE and FORGE2

Several factors must be taken into account when comparing FORGE and FORGE2. These tools use different datasets, the first focusing on DNase I hotspot data and the latter focusing on histone mark broadPeaks. Although in many cases DNase I hotspots and histone marks can represent the same elements (such as potential enhancers in the case of the NK cell-specific signal for SNPs associated with SLE and Systemic Sclerosis), DNase I hotspots can represent many different regulatory element classes, including promoters, enhancers, and insulators. A tissue-specific DNase I hotspot enrichment signal may be indicative of an overlap with any or all of these regulatory element classes, whereas the histone marks in FORGE2 each present enrichment for a subgroup of regulatory elements, thus presenting a more interpretable analysis and in some ways deconvoluting FORGE signal. FORGE and FORGE2 also use different frameworks for dealing with multiple testing correction. FORGE uses tissue-level Bonferroni correction and FORGE2 uses an FDR-based approach (Benjamini-Yekutieli). In addition, FORGE and FORGE2 contain data for different tissues although in the cases for which I observe completely new enrichments (see Methods) all the FORGE2 tissues described are also present in FORGE.

6.2.4 Outlook on FORGE2

FORGE2 is a powerful new tool for cell type-specific enrichment analysis of complex disease-associated variation, highlighting new results not detected by FORGE. In

addition, FORGE2 facilitates the prioritisation of candidate variants, aiding experimental approaches. I provide this tool for use by the community, in order to aid the functional characterisation of GWAS SNPs in a cell type-specific manner, a pressing issue in view of the current lack of functional understanding of GWAS SNPs²⁴⁹. I also report a list of novel tissue-disease associations in order to aid the characterisation of relevant GWAS-associated mechanisms through functional genomics approaches. In conclusion, I anticipate that FORGE2 will be a valuable tool for current research on disease genomics, detecting novel cell type-specific enrichments for GWAS SNP lists in a histone mark-specific context.

6.3 4C-seq analysis

6.3.1 4C-seq analysis aims and context

A large proportion of variants associated with diseases by EWAS and GWAS are either intergenic or intronic, and thus present challenges for functional interpretation^{17,136}. It has been proposed that a proportion of these variants may affect phenotype by influencing regulatory elements that control gene expression²⁵⁴. This is supported by the fact that many EWAS and GWAS variants are preferentially located in open chromatin regions from tissues that have a known involvement in disease^{198,139,146}. Open chromatin regions, and particularly DHSs, are representative of all the main classes of cis-regulatory elements¹⁹⁹, including enhancers. Histone

mark enrichment analyses shown here (FORGE2, eFORGE) and also by other groups⁸¹ indicate that many tissue-specific open chromatin enrichments are replicated by analyses using data for H3K4me1, a classic enhancer mark, suggesting that many GWAS and EWAS variants do not only act through cis-regulatory elements but specifically enhancers, thus potentially affecting gene expression through enhancer modulation. This is also supported by complementary analyses that use chromatin state enhancer annotations²⁵². It is known that enhancers loop in three-dimensional space to regulate their target genes^{254,255}. Thus the identification of enhancer-located variants seems to point a mechanism of action in which variants affect enhancer activity and thus the expression of target genes. It is, however, hard to establish target genes for many enhancers, with some enhancer-promoter contacts spanning megabases²⁵⁶. It has also been shown that the strategy to link variants to the closest gene is a poor predictor of enhancer-promoter contacts²⁵². In this context the application of 3D genomic techniques has proved effective, as in the study of the obesity-associated FTO locus^{130,135} and the QRS duration-associated SCN10A locus²⁰⁰. In both of these cases a distal enhancer was linked to a gene involved in a key disease-related process. Through these examples it has become clear that knowledge of the 3D-genomic context is of great help in the characterisation of certain disease-associated loci. Furthering this effort, I proposed to study the three-dimensional context of selected GWAS loci in mNSCs (homologous regions) and human immune cell types, in order to increase knowledge of the 3D structure of these regions and establish candidate target genes for GWAS and EWAS variant action.

When studying genomic interactions there are many techniques to choose from, including FISH and 3C techniques such as 4C, 5C, Hi-C, capture Hi-C, capture-C, and ChIA-PET^{104,111} (discussed in the Introduction). Each one of these techniques shows specific advantages optimised for a particular application, and thus no one technique fits all 3D genomic studies. Of all these techniques the choice depends, as always, on the specific scientific question posed. In my study of the interactomes of GWAS and EWAS regions I also have had to balance cost with efficiency to select the most adequate technique for the purpose. Specifically, selection of FISH would have been inadequate considering the resources available and the scalability offered by current 3C-based technologies. Within 3C-based technologies, simple 3C (which measures one vs one interactions) would have been inadequate due to the low throughput it presents. In my study I sought to fine-map the proximal interactome for each of my candidate regions. Therefore a one vs all approach such as 4C-seq or capture-C was the most suitable, as I could repeat the technique for each of my candidate regions and I could analyse these with a high resolution that large-scale methods such as 5C, Hi-C or ChIA-PET cannot provide. Given the 4C-seq in-house expertise at the UCL Cancer Institute (Hadjur group) I decided to favour 4C-seq as my method of choice. To summarise, 4C-seq and capture-C both provide the ability to map fine details of local 3D-genomic structure, and I decided to use 4C-seq based on in-house expertise (facilitating the sharing of specific protocols, reagents and experimental material with comparable data).

In order to prioritise variants for 4C-seq analysis I have favoured a multidimensional data analysis approach using eFORGE and FORGE2, thus integrating the computational and experimental sides of this project (see Methods). In addition I have also used several other computational tools including FORGE¹³⁹, the Braineac database¹⁵⁴, the GTEx database¹⁵⁵, UCSC LiftOver¹⁵⁷ and the ChIP tool¹⁵². Multiple datasets were used including DNase I hotspots, expression Quantitative Trait Loci (eQTL), FANTOM5 enhancer annotations and promoter Capture Hi-C contacts. It is important to highlight that for all these datasets the cell type-specific context of the data was taken into account, as each one of these epigenomic features varies between cell types. In addition, information regarding sequence conservation was taken into account for selecting mouse regions homologous to human GWAS SNP loci. Genomic context and disease association data were taken into account for all loci analysed.

Complementing efforts to characterise the disparity in 3D genomic structure among different types of blood cells¹⁴⁷ I prioritised the study of cell type-specific interactions among immune cell types (rather than studying one cell type under several conditions). This was done in order to gain an understanding of the cell type-specific nature of candidate interactions, complementing my bioinformatics approaches that seek to uncover the candidate cell types involved in disease mechanism.

Given the availability of sorted human immune cell types from the BLUEPRINT project, of which I was a participant, and my objective to study the cell type-specific

nature of selected loci, I favoured a study design involving 5 immune cell types (CD14+, CD4+, CD8+, B cells and NK cells). Future approaches, using these data as a reference dataset for the proximal interactomes of selected loci, may choose to further research on these regions by analysing a single cell type from this study design under different conditions (e.g. activation with LPS) or in a case-control design, obtaining the same sorted cell type from healthy and diseased subjects.

6.3.2 4C-seq analysis findings

I have found a range of cell type-specific interactions for selected disease-associated loci. The most notable of the cell type-specific interactions detected was for the IKZF3 locus, linking intergenic variant rs12946510 to different isoforms of IKZF3 in CD14+, CD4+, CD8+, NK and B cells. This shows relevance for the aetiology of IBD, linking a GWAS variant to a target pathway of lenalidomide, a candidate drug projected to have efficiency in the treatment of IBD subtypes²³⁴. Importantly, interactions between rs12946510 and IKZF3 cannot be observed at the resolution of BLUEPRINT promoter Capture Hi-C data¹⁵⁰. The uncovering of this interaction supports the implementation of a targeted high-resolution 4C-seq approach to study GWAS loci, providing evidence for enhancer contacts not observable in current state-of-the-art data.

Additionally, several eQTL associations were studied. Notably, a homologous region to that containing rs548181 (indicated by the Braineac database to be an eQTL

for 13 proximal genes) was shown to interact only with Fez1 in mNSCs. eQTL associations can be mediated through cis or trans effects, and my research on the locus homologous to rs548181 in mNSCs suggests cis effects for Fez1 regulation. Crucially, FEZ1 has been identified as a schizophrenia-associated gene in humans for independent reasons²⁰⁷. Given that rs548181 is an intronic GWAS SNP it is important to characterise its gene targets and potential regulatory effects. Knowledge of its direct interaction with Fez1 is of utility in establishing a functional role for this schizophrenia-associated variant. This is important as previous family-based studies have been unable to put forward any links between rs548181 and downstream mechanisms²⁰⁶. It is also important to highlight, however, that further research in human cells would be required to confirm these findings.

In addition to uncovering target pathways for disease-associated loci I have also discovered potential new enhancers and shown a mosaic of cell type-specific interactions at high resolution, in some cases on related genes in the same donors (as is the case of CSF2 and CSF2RB), providing a resource for the immunology community to explore potential mechanisms of GWAS SNP action. The discovery of potential new enhancers is an advantage resulting from the “one vs all” approach of 4C-seq, which has allowed me to evaluate potential “other” interactions not involving known promoters or enhancers to investigate potentially novel mechanisms underlying disease aetiology.

Last, but not least, I have generated what are, to my knowledge, the first 4C-seq maps for NK cells ever made, validating the use of 4C-seq in one of the most important immune cell types.

6.3.3 Outlook on 4C-seq analysis

In line with previous research that has established *Mus musculus* as one of the main model organisms in mammalian biology¹⁹⁴, I applied 4C-seq in mouse neural stem cells as a test system to study regions homologous to human GWAS loci. Resulting 4C-seq analyses of these mouse regions suggest potential targets for further analysis. Despite positive findings in mouse NSCs, there is still an open question as to whether these findings hold for equivalent human neural progenitors. In order for these findings to have further impact on our understanding of disease mechanism it would be necessary to perform one of two actions: either to demonstrate the same 4C-seq interaction in human cells comparing healthy and diseased subjects, or to perform CRISPR/Cas9 genome editing of the disease-associated variant in human cells, followed by elucidating the influence of CRISPR/Cas9-related genome editing on gene expression. A subsequent step would be to characterise the link between gene expression and phenotype. Similarly, CRISPR/Cas9 genome editing in sorted blood cell types will serve to clarify the phenotypic effect of immune disease-associated variants, in addition to aiding the identification of druggable targets. The immune cell type-specific interactomes I have generated will aid the prioritisation of candidate cell types for genome editing efforts on selected loci. In line with previous

studies on variants in the FTO locus^{130,135} and the SCN10A locus²⁰⁰, I have generated 3D genomic data for a range of disease-associated loci, an important milestone on the way to uncovering disease-relevant mechanisms for selected variants.

Several additional caveats must be highlighted. For instance, disease-relevant interactions may not be present for the cell type studied. Cell type selection is critical when studying enhancer-promoter contacts and it is possible that more advanced stages of neural differentiation will show interactions not present in mNSCs for the neural disease-associated regions I have analysed. It is also possible that other immune cell types not studied here may reveal immune disease-relevant interactions for selected loci. I have, however, taken steps to optimise target region selection in view of the cell types available. When selecting candidate loci (see Methods) I have used cell type-specific data for a range of epigenomic datasets. I have fine-tuned this analysis to the level of distinguishing related immune cell types (e.g. CD4+ and CD8+ T cell epigenomic datasets were separated for analysis). I have also used tissue-level data in certain cases. For example, I have prioritised GWAS SNPs that may exert a regulatory effect in NSCs by selecting variants that are eQTL across different brain regions.

Another caveat involves analysing the correct target region within the locus studied. Given the requirement for unique 20 bp sequences within each target, due to the need for effective PCR primers in 4C, some of the pre-designed PCR primers from van de Werken et al. (2012)¹⁴⁹ (http://compgenomics.weizmann.ac.il/tanay/?page_id=367) were proximal (>5kb)

but not exactly overlapping target positions. Despite the fact that for such small genomic distances I do not expect a significant loss of evidence for genomic contacts, the fact that some primers do not exactly overlap target positions (due to the aforementioned constraint for 4C primer design) adds a consideration when interpreting the resulting data that must be taken into account.

It is also important to note that the absence of evidence for genomic interaction does not constitute evidence for the absence of regulatory links between GWAS SNPs and eQTL-affected genes. The way eQTL associations are identified does not necessarily imply three-dimensional genomic contact between participating regions, as there are other ways a GWAS SNP may affect the expression of a particular gene (including regulation involving downstream pathways).

Given the dynamic nature of the epigenome of immune cells, it is possible that certain enhancer contacts only take place in certain physiological conditions, such as LPS-induced activation. Here I have presented analyses from healthy donors in a normal physiological state. Further approaches may apply immune activation protocols to study cells in an activated state.

In conclusion, I do not only provide detailed 4C-seq results for a list of genomic loci but also a method for ranking candidate GWAS SNPs (and EWAS DMPs) for interactome analysis. I thus provide analytical and experimental resources to face one of the main challenges in the study of disease-associated intergenic regions: the identification of target genes.

7 References

1. Berger, S. L., Kouzarides, T., Shiekhataar, R. & Shilatifard, A. An operational definition of epigenetics. *Genes Dev.* **23**, 781–783 (2009).
2. Shlyueva, D., Stampfel, G. & Stark, A. Transcriptional enhancers: from properties to genome-wide predictions. *Nat. Rev. Genet.* **15**, 272–286 (2014).
3. Varley, K. E. *et al.* Dynamic DNA methylation across diverse human cell lines and tissues. *Genome Res.* **23**, 555 (2013).
4. Jones, P. A. Functions of DNA methylation: islands, start sites, gene bodies and beyond. *Nat. Rev. Genet.* **13**, 484–492 (2012).
5. Libertini, E. *et al.* Saturation analysis for whole-genome bisulfite sequencing data. *Nat. Biotechnol.* **34**, 691–693 (2016).
6. Meissner, A. *et al.* Reduced representation bisulfite sequencing for comparative high-resolution DNA methylation analysis. *Nucleic Acids Res.* **33**, 5868 (2005).
7. Lee, Y. K. *et al.* Improved reduced representation bisulfite sequencing for epigenomic profiling of clinical samples. *Biol. Proced. Online* **16**, 1 (2014).
8. Michels, K. B. *et al.* Recommendations for the design and analysis of epigenome-wide association studies. *Nat. Methods* **10**, 949–955 (2013).
9. Sinoquet, C. & Mourad, R. *Probabilistic Graphical Models for Genetics, Genomics, and Postgenomics*. (OUP Oxford, 2014).
10. Morris, T. J. *et al.* ChAMP: 450k Chip Analysis Methylation Pipeline. *Bioinformatics* **30**, 428–430 (2014).
11. Aryee, M. J. *et al.* Minfi: a flexible and comprehensive Bioconductor package for the analysis of Infinium DNA methylation microarrays. *Bioinforma. Oxf. Engl.* **30**, 1363–1369 (2014).

12. Assenov, Y. *et al.* Comprehensive analysis of DNA methylation data with RnBeads. *Nat. Methods* **11**, 1138–1140 (2014).
13. Pidsley, R. *et al.* A data-driven approach to preprocessing Illumina 450K methylation array data. *BMC Genomics* **14**, 293 (2013).
14. Bibikova, M. *et al.* High density DNA methylation array with single CpG site resolution. *Genomics* **98**, 288–295 (2011).
15. Titus, A. J., Houseman, E. A., Johnson, K. C. & Christensen, B. C. methyLiftOver: cross-platform DNA methylation data integration. *Bioinformatics* **32**, 2517–2519 (2016).
16. Do, C. *et al.* Genetic–epigenetic interactions in cis: a major focus in the post-GWAS era. *Genome Biol.* **18**, 120 (2017).
17. Rakyan, V. K., Down, T. A., Balding, D. J. & Beck, S. Epigenome-wide association studies for common human diseases. *Nat. Rev. Genet.* **12**, 529–541 (2011).
18. Birney, E., Smith, G. D. & Greally, J. M. Epigenome-wide Association Studies and the Interpretation of Disease -Omics. *PLOS Genet* **12**, e1006105 (2016).
19. The BLUEPRINT Consortium. Quantitative comparison of DNA methylation assays for biomarker development and clinical applications. *Nat. Biotechnol.* **34**, 726–737 (2016).
20. Davis, S., Du, P., Bilke, S., Triche, T. & Bootwalla, M. methylumi: Handle Illumina methylation data. *R Package Version 2*, (2012).
21. Morris, T. J. & Beck, S. Analysis pipelines and packages for Infinium HumanMethylation450 BeadChip (450k) data. *Methods San Diego Calif* **72**, 3–8 (2015).

22. Chen, Y. *et al.* Discovery of cross-reactive probes and polymorphic CpGs in the Illumina Infinium HumanMethylation450 microarray. *Epigenetics* **8**, 203–209 (2013).
23. Patil, V., Ward, R. L. & Hesson, L. B. The evidence for functional non-CpG methylation in mammalian cells. *Epigenetics* **9**, 823–828 (2014).
24. Butcher, L. M. *et al.* Non-CG DNA methylation is a biomarker for assessing endodermal differentiation capacity in pluripotent stem cells. *Nat. Commun.* **7**, 10458 (2016).
25. Maksimovic, J., Gordon, L. & Oshlack, A. SWAN: Subset-quantile Within Array Normalization for Illumina Infinium HumanMethylation450 BeadChips. *Genome Biol.* **13**, R44 (2012).
26. Teschendorff, A. E. *et al.* A beta-mixture quantile normalization method for correcting probe design bias in Illumina Infinium 450 k DNA methylation data. *Bioinformatics* **29**, 189–196 (2013).
27. Dedeurwaerder, S. *et al.* Evaluation of the Infinium Methylation 450K technology. *Epigenomics* **3**, 771–784 (2011).
28. Fortin, J.-P. *et al.* Functional normalization of 450k methylation array data improves replication in large cancer studies. *Genome Biol.* **15**, 503 (2014).
29. Teschendorff, A. E. *et al.* An epigenetic signature in peripheral blood predicts active ovarian cancer. *PloS One* **4**, e8274 (2009).
30. Johnson, W. E., Li, C. & Rabinovic, A. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostat. Oxf. Engl.* **8**, 118–127 (2007).

31. Chen, C. *et al.* Removing Batch Effects in Analysis of Expression Microarray Data: An Evaluation of Six Batch Adjustment Methods. *PLOS ONE* **6**, e17238 (2011).
32. Butcher, L. M. & Beck, S. Probe Lasso: a novel method to rope in differentially methylated regions with 450K DNA methylation data. *Methods San Diego Calif* **72**, 21–28 (2015).
33. Jaffe, A. E. *et al.* Bump hunting to identify differentially methylated regions in epigenetic epidemiology studies. *Int. J. Epidemiol.* **41**, 200–209 (2012).
34. Pedersen, B. S., Schwartz, D. A., Yang, I. V. & Kechris, K. J. Comb-p: software for combining, analyzing, grouping and correcting spatially correlated P-values. *Bioinformatics* **28**, 2986–2988 (2012).
35. Peters, T. J. *et al.* De novo identification of differentially methylated regions in the human genome. *Epigenetics Chromatin* **8**, 6 (2015).
36. Libertini, E. *et al.* Information recovery from low coverage whole-genome bisulfite sequencing. *Nat. Commun.* **7**, 11306 (2016).
37. Eckhardt, F. *et al.* DNA methylation profiling of human chromosomes 6, 20 and 22. *Nat. Genet.* **38**, 1378–1385 (2006).
38. Li, Y. *et al.* The DNA Methylome of Human Peripheral Blood Mononuclear Cells. *PLoS Biol* **8**, e1000533 (2010).
39. Ashburner, M. *et al.* Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.* **25**, 25–29 (2000).
40. Gene Ontology Consortium: going forward. *Nucleic Acids Res.* **43**, D1049–D1056 (2015).
41. Kanehisa, M. & Goto, S. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res.* **28**, 27–30 (2000).

42. Subramanian, A. *et al.* Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proc. Natl. Acad. Sci.* **102**, 15545–15550 (2005).
43. Huang, D. W., Sherman, B. T. & Lempicki, R. A. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nat. Protoc.* **4**, 44–57 (2009).
44. Huang, D. W., Sherman, B. T. & Lempicki, R. A. Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucleic Acids Res.* **37**, 1–13 (2009).
45. Carbon, S. *et al.* AmiGO: online access to ontology and annotation data. *Bioinforma. Oxf. Engl.* **25**, 288–289 (2009).
46. Jiao, Y., Widschwendter, M. & Teschendorff, A. E. A systems-level integrative framework for genome-wide DNA methylation and gene expression data identifies differential gene expression modules under epigenetic control. *Bioinformatics* **30**, 2360–2366 (2014).
47. Teschendorff, A. E. *et al.* DNA methylation outliers in normal breast tissue identify field defects that are enriched in cancer. *Nat. Commun.* **7**, (2016).
48. Houseman, E. A. *et al.* DNA methylation arrays as surrogate measures of cell mixture distribution. *BMC Bioinformatics* **13**, 86 (2012).
49. Zou, J., Lippert, C., Heckerman, D., Aryee, M. & Listgarten, J. Epigenome-wide association studies without the need for cell-type composition. *Nat. Methods* **11**, 309–311 (2014).
50. McGregor, K. *et al.* An evaluation of methods correcting for cell-type heterogeneity in DNA methylation studies. *Genome Biol.* **17**, 84 (2016).

51. Houseman, E. A. *et al.* Reference-free deconvolution of DNA methylation data and mediation by cell composition effects. *BMC Bioinformatics* **17**, (2016).
52. Houseman, E. A., Molitor, J. & Marsit, C. J. Reference-free cell mixture adjustments in analysis of DNA methylation data. *Bioinformatics* **30**, 1431–1439 (2014).
53. Liang, L. & Cookson, W. O. C. Grasping nettles: cellular heterogeneity and other confounders in epigenome-wide association studies. *Hum. Mol. Genet.* **23**, R83–R88 (2014).
54. Wilhelm-Benartzi, C. S. *et al.* Review of processing and analysis methods for DNA methylation array data. *Br. J. Cancer* **109**, 1394–1402 (2013).
55. Song, L. & Crawford, G. E. DNase-seq: a high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells. *Cold Spring Harb. Protoc.* **2010**, pdb.prot5384 (2010).
56. Buenrostro, J. D., Giresi, P. G., Zaba, L. C., Chang, H. Y. & Greenleaf, W. J. Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, DNA-binding proteins and nucleosome position. *Nat. Methods* **10**, 1213–1218 (2013).
57. Giresi, P. G., Kim, J., McDaniell, R. M., Iyer, V. R. & Lieb, J. D. FAIRE (Formaldehyde-Assisted Isolation of Regulatory Elements) isolates active regulatory elements from human chromatin. *Genome Res.* **17**, 877 (2007).
58. Simon, J. M., Giresi, P. G., Davis, I. J. & Lieb, J. D. Using formaldehyde-assisted isolation of regulatory elements (FAIRE) to isolate active regulatory DNA. *Nat. Protoc.* **7**, 256–267 (2012).

59. Furey, T. S. ChIP-seq and beyond: new and improved methodologies to detect and characterize protein-DNA interactions. *Nat. Rev. Genet.* **13**, 840–852 (2012).
60. Muñoz-López, M. & García-Pérez, J. L. DNA Transposons: Nature and Applications in Genomics. *Curr. Genomics* **11**, 115–128 (2010).
61. Chénais, B. Transposable elements and human cancer: A causal relationship? *Biochim. Biophys. Acta BBA - Rev. Cancer* **1835**, 28–35 (2013).
62. Merckenschlager, M. & Odom, D. T. CTCF and cohesin: linking gene regulatory elements with their targets. *Cell* **152**, 1285–1297 (2013).
63. Stergachis, A. B. *et al.* Developmental Fate and Cellular Maturity Encoded in Human Regulatory DNA Landscapes. *Cell* **154**, 888–903 (2013).
64. Waddington, C. H. *The Strategy of the Genes*. (Routledge, 2014).
65. Bunting, K. L. *et al.* Multi-tiered Reorganization of the Genome during B Cell Affinity Maturation Anchored by a Germinal Center-Specific Locus Control Region. *Immunity* **45**, 497–512 (2016).
66. Whalen, S., Truty, R. M. & Pollard, K. S. Enhancer-promoter interactions are encoded by complex genomic signatures on looping chromatin. *Nat. Genet.* **48**, 488–496 (2016).
67. Zhu, Y. *et al.* Constructing 3D interaction maps from 1D epigenomes. *Nat. Commun.* **7**, 10812 (2016).
68. Zentner, G. E. & Henikoff, S. Regulation of nucleosome dynamics by histone modifications. *Nat. Struct. Mol. Biol.* **20**, 259–266 (2013).
69. Huang, H., Sabari, B. R., Garcia, B. A., Allis, C. D. & Zhao, Y. SnapShot: Histone Modifications. *Cell* **159**, 458–458.e1 (2014).

70. Jenuwein, T. & Allis, C. D. Translating the histone code. *Science* **293**, 1074–1080 (2001).
71. Barski, A. *et al.* High-Resolution Profiling of Histone Methylations in the Human Genome. *Cell* **129**, 823–837 (2007).
72. Allis, C. D. & Jenuwein, T. The molecular hallmarks of epigenetic control. *Nat. Rev. Genet.* **17**, 487–500 (2016).
73. Zhou, V. W., Goren, A. & Bernstein, B. E. Charting histone modifications and the functional organization of mammalian genomes. *Nat. Rev. Genet.* **12**, 7–18 (2011).
74. Martens, J. H. A., Stunnenberg, H. G. & Logie, C. The Decade of the Epigenomes? *Genes Cancer* **2**, 680–687 (2011).
75. Robert, V. J. *et al.* The SET-2/SET1 Histone H3K4 Methyltransferase Maintains Pluripotency in the *Caenorhabditis elegans* Germline. *Cell Rep.* **9**, 443–450 (2014).
76. Bannister, A. J. & Kouzarides, T. Regulation of chromatin by histone modifications. *Cell Res.* **21**, 381–395 (2011).
77. Vermeulen, M. *et al.* Selective anchoring of TFIID to nucleosomes by trimethylation of histone H3 lysine 4. *Cell* **131**, 58–69 (2007).
78. Voigt, P., Tee, W.-W. & Reinberg, D. A double take on bivalent promoters. *Genes Dev.* **27**, 1318–1338 (2013).
79. Calo, E. & Wysocka, J. Modification of Enhancer Chromatin: What, How, and Why? *Mol. Cell* **49**, 825–837 (2013).
80. Dahl, J. A. *et al.* Broad histone H3K4me3 domains in mouse oocytes modulate maternal-to-zygotic transition. *Nature* **537**, 548–552 (2016).

81. Roadmap Epigenomics Consortium *et al.* Integrative analysis of 111 reference human epigenomes. *Nature* **518**, 317–330 (2015).
82. Hu, D. *et al.* The MLL3/MLL4 branches of the COMPASS family function as major histone H3K4 monomethylases at enhancers. *Mol. Cell. Biol.* **33**, 4745–4754 (2013).
83. Jeong, K. W. *et al.* Recognition of enhancer element-specific histone methylation by TIP60 in transcriptional activation. *Nat. Struct. Mol. Biol.* **18**, 1358–1365 (2011).
84. Creyghton, M. P. *et al.* Histone H3K27ac separates active from poised enhancers and predicts developmental state. *Proc. Natl. Acad. Sci. U. S. A.* **107**, 21931–21936 (2010).
85. Tie, F. *et al.* CBP-mediated acetylation of histone H3 lysine 27 antagonizes *Drosophila* Polycomb silencing. *Dev. Camb. Engl.* **136**, 3131–3141 (2009).
86. Ernst, J. & Kellis, M. Large-scale imputation of epigenomic datasets for systematic annotation of diverse human tissues. *Nat. Biotechnol.* **33**, 364–376 (2015).
87. Filippakopoulos, P. *et al.* Histone Recognition and Large-Scale Structural Analysis of the Human Bromodomain Family. *Cell* **149**, 214–231 (2012).
88. Morey, L. & Helin, K. Polycomb group protein-mediated repression of transcription. *Trends Biochem. Sci.* **35**, 323–332 (2010).
89. Thornton, S. R., Butty, V. L., Levine, S. S. & Boyer, L. A. Polycomb Repressive Complex 2 Regulates Lineage Fidelity during Embryonic Stem Cell Differentiation. *PLOS ONE* **9**, e110498 (2014).
90. Wagner, E. J. & Carpenter, P. B. Understanding the language of Lys36 methylation at histone H3. *Nat. Rev. Mol. Cell Biol.* **13**, 115–126 (2012).

91. Wagner, E. J. & Carpenter, P. B. Understanding the language of Lys36 methylation at histone H3. *Nat. Rev. Mol. Cell Biol.* **13**, 115–126 (2012).
92. Wen, H. *et al.* ZMYND11 links histone H3.3K36me3 to transcription elongation and tumour suppression. *Nature* **508**, 263–268 (2014).
93. Schwartz, S., Meshorer, E. & Ast, G. Chromatin organization marks exon-intron structure. *Nat. Struct. Mol. Biol.* **16**, 990–995 (2009).
94. Ehrlich, M. *et al.* ICF, An Immunodeficiency Syndrome: DNA Methyltransferase 3B Involvement, Chromosome Anomalies, and Gene Dysregulation. *Autoimmunity* **41**, 253–271 (2008).
95. Yao, T. P. *et al.* Gene dosage-dependent embryonic development and proliferation defects in mice lacking the transcriptional integrator p300. *Cell* **93**, 361–372 (1998).
96. Tanaka, Y. *et al.* Extensive brain hemorrhage and embryonic lethality in a mouse null mutant of CREB-binding protein. *Mech. Dev.* **95**, 133–145 (2000).
97. Aucott, R. *et al.* HP1-beta is required for development of the cerebral neocortex and neuromuscular junctions. *J. Cell Biol.* **183**, 597–606 (2008).
98. Bledau, A. S. *et al.* The H3K4 methyltransferase Setd1a is first required at the epiblast stage, whereas Setd1b becomes essential after gastrulation. *Development* **141**, 1022–1035 (2014).
99. Lee, S. C. W. *et al.* Polycomb repressive complex 2 component Suz12 is required for hematopoietic stem cell function and lymphopoiesis. *Blood* **126**, 167–175 (2015).

100. Hu, M. *et al.* Histone H3 lysine 36 methyltransferase Hypb/Setd2 is required for embryonic vascular remodeling. *Proc. Natl. Acad. Sci.* **107**, 2956–2961 (2010).
101. Hu, Y. *et al.* Homozygous Disruption of the Tip60 Gene Causes Early Embryonic Lethality. *Dev. Dyn. Off. Publ. Am. Assoc. Anat.* **238**, 2912–2921 (2009).
102. Alberts, B. *et al.* Chromosomal DNA and Its Packaging in the Chromatin Fiber. (2002).
103. Maeshima, K. & Eltsov, M. Packaging the Genome: the Structure of Mitotic Chromosomes. *J. Biochem. (Tokyo)* **143**, 145–153 (2008).
104. Hughes, J. R. *et al.* Analysis of hundreds of cis-regulatory landscapes at high resolution in a single, high-throughput experiment. *Nat. Genet.* **46**, 205–212 (2014).
105. Ramón Cajal, S. Un sencillo metodo de coloración selectiva del reticulo protoplásmico. *Trab Lab Invest Biol* **2**, 129–221 (1903).
106. Barr, M. L. & Bertram, E. G. A morphological distinction between neurones of the male and female, and the behaviour of the nucleolar satellite during accelerated nucleoprotein synthesis. in *Problems of Birth Defects* 101–102 (Springer, 1949).
107. Bauman, J. G. J., Wiegant, J., Borst, P. & Van Duijn, P. A new method for fluorescence microscopical localization of specific DNA sequences by in situ hybridization of fluorochrome-labelled RNA. *Exp. Cell Res.* **128**, 485–490 (1980).

108. Williamson, I. *et al.* Spatial genome organization: contrasting views from chromosome conformation capture and fluorescence in situ hybridization. *Genes Dev.* **28**, 2778–2791 (2014).
109. Dekker, J., Rippe, K., Dekker, M. & Kleckner, N. Capturing chromosome conformation. *Science* **295**, 1306–1311 (2002).
110. Davies, J. O. J., Oudelaar, A. M., Higgs, D. R. & Hughes, J. R. How best to identify chromosomal interactions: a comparison of approaches. *Nat. Methods* **14**, 125–134 (2017).
111. Wit, E. de & Laat, W. de. A decade of 3C technologies: insights into nuclear organization. *Genes Dev.* **26**, 11–24 (2012).
112. Mifsud, B. *et al.* Mapping long-range promoter contacts in human cells with high-resolution capture Hi-C. *Nat. Genet.* **47**, 598–606 (2015).
113. Zhao, Z. *et al.* Circular chromosome conformation capture (4C) uncovers extensive networks of epigenetically regulated intra- and interchromosomal interactions. *Nat. Genet.* **38**, 1341–1347 (2006).
114. Dostie, J. *et al.* Chromosome Conformation Capture Carbon Copy (5C): A massively parallel solution for mapping interactions between genomic elements. *Genome Res.* **16**, 1299–1309 (2006).
115. Belton, J.-M. *et al.* Hi-C: a comprehensive technique to capture the conformation of genomes. *Methods San Diego Calif* **58**, 268–276 (2012).
116. Horike, S., Cai, S., Miyano, M., Cheng, J.-F. & Kohwi-Shigematsu, T. Loss of silent-chromatin looping and impaired imprinting of DLX5 in Rett syndrome. *Nat. Genet.* **37**, 31–40 (2005).
117. Fullwood, M. J. *et al.* An oestrogen-receptor- α -bound human chromatin interactome. *Nature* **462**, 58–64 (2009).

118. Garrod, A. E. & Harris, H. *Inborn errors of metabolism*. (Oxford University Press, 1909).
119. Thomsen, S. F. Genetics of asthma: an introduction for the clinician. *Eur. Clin. Respir. J.* **2**, (2015).
120. Lander, E. S. *et al.* Initial sequencing and analysis of the human genome. *Nature* **409**, 860–921 (2001).
121. Burton, P. R. *et al.* Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature* **447**, 661–678 (2007).
122. Iotchkova, V. *et al.* Discovery and refinement of genetic loci associated with cardiometabolic risk using dense imputation maps. *Nat. Genet.* **48**, 1303–1312 (2016).
123. Welter, D. *et al.* The NHGRI GWAS Catalog, a curated resource of SNP-trait associations. *Nucleic Acids Res.* **42**, D1001–D1006 (2014).
124. Slatkin, M. Linkage disequilibrium — understanding the evolutionary past and mapping the medical future. *Nat. Rev. Genet.* **9**, 477 (2008).
125. Cortes, A. & Brown, M. A. Promise and pitfalls of the Immunochip. *Arthritis Res. Ther.* **13**, 101 (2011).
126. Voight, B. F. *et al.* The metabochip, a custom genotyping array for genetic studies of metabolic, cardiovascular, and anthropometric traits. *PLoS Genet.* **8**, e1002793 (2012).
127. Farh, K. K.-H. *et al.* Genetic and epigenetic fine mapping of causal autoimmune disease variants. *Nature* **518**, 337–343 (2015).
128. Dahl, A. *et al.* A multiple-phenotype imputation method for genetic studies. *Nat. Genet.* **48**, 466–472 (2016).

129. Fusi, N., Lippert, C., Lawrence, N. D. & Stegle, O. Warped linear mixed models for the genetic analysis of transformed phenotypes. *Nat. Commun.* **5**, 4890 (2014).
130. Claussnitzer, M. *et al.* FTO Obesity Variant Circuitry and Adipocyte Browning in Humans. *N. Engl. J. Med.* **373**, 895–907 (2015).
131. Wang, X. *et al.* Discovery and validation of sub-threshold genome-wide association study loci using epigenomic signatures. *eLife* **5**, e10557 (2016).
132. Hall, D. M.-H. & Smoller, D. J. W. A New Role for Endophenotypes in the GWAS Era: Functional Characterization of Risk Variants. *Harv. Rev. Psychiatry* **18**, 67 (2010).
133. Gibson, G. Rare and common variants: twenty arguments. *Nat. Rev. Genet.* **13**, 135–145 (2012).
134. Lek, M. *et al.* Analysis of protein-coding genetic variation in 60,706 humans. *BioRxiv* 030338 (2016).
135. Smemo, S. *et al.* Obesity-associated variants within FTO form long-range functional connections with IRX3. *Nature* **advance online publication**, (2014).
136. Hindorff, L. A. *et al.* Potential etiologic and functional implications of genome-wide association loci for human diseases and traits. *Proc. Natl. Acad. Sci.* **106**, 9362–9367 (2009).
137. Ward, L. D. & Kellis, M. HaploReg v4: systematic mining of putative causal variants, cell types, regulators and target genes for human complex traits and disease. *Nucleic Acids Res.* **44**, D877–881 (2016).
138. Boyle, A. P. *et al.* Annotation of functional variation in personal genomes using RegulomeDB. *Genome Res.* **22**, 1790–1797 (2012).

139. Dunham, I., Kulesha, E., Iotchkova, V., Morganella, S. & Birney, E. FORGE : A tool to discover cell specific enrichments of GWAS associated SNPs in regulatory regions. *bioRxiv* 013045 (2014). doi:10.1101/013045
140. Trynka, G. *et al.* Disentangling the Effects of Colocalizing Genomic Annotations to Functionally Prioritize Non-coding Variants within Complex-Trait Loci. *Am. J. Hum. Genet.* **97**, 139–152 (2015).
141. Tak, Y. G. & Farnham, P. J. Making sense of GWAS: using epigenomics and genome engineering to understand the functional relevance of SNPs in non-coding regions of the human genome. *Epigenetics Chromatin* **8**, 57 (2015).
142. Relton, C. L. & Davey Smith, G. Two-step epigenetic Mendelian randomization: a strategy for establishing the causal role of epigenetic processes in pathways to disease. *Int. J. Epidemiol.* **41**, 161–176 (2012).
143. Paul, D. S. & Beck, S. Advances in epigenome-wide association studies for common diseases. *Trends Mol. Med.* **20**, 541–543 (2014).
144. Philibert, R. *et al.* A quantitative epigenetic approach for the assessment of cigarette consumption. *Psychol. Clin. Settings* 656 (2015).
doi:10.3389/fpsyg.2015.00656
145. Lord, J. & Cruchaga, C. The epigenetic landscape of Alzheimer's disease. *Nat. Neurosci.* **17**, 1138–1140 (2014).
146. Breeze, C. E. *et al.* eFORGE: A Tool for Identifying Cell Type-Specific Signal in Epigenomic Data. *Cell Rep.* **17**, 2137–2150 (2016).
147. Javierre, B. M. *et al.* Lineage-Specific Genome Architecture Links Enhancers and Non-coding Disease Variants to Target Gene Promoters. *Cell* **167**, 1369–1384.e19 (2016).

148. van de Werken, H. J. G. *et al.* Chapter Four - 4C Technology: Protocols and Data Analysis. in *Methods in Enzymology* (ed. Allis, C. W. and C. D.) **513**, 89–112 (Academic Press, 2012).
149. van de Werken, H. J. G. *et al.* Robust 4C-seq data analysis to screen for regulatory DNA interactions. *Nat. Methods* **9**, 969–972 (2012).
150. Javierre, B. M. *et al.* Lineage-specific genome architecture links enhancers and non-coding disease variants to target gene promoters. *Cell* **167**, 1369–1384 (2016).
151. Gutierrez-Arcelus, M. *et al.* Passive and active DNA methylation and the interplay with genetic variation in gene regulation. *eLife* **2**, e00523 (2013).
152. Schofield, E. C. *et al.* CHiCP: a web-based tool for the integrative and interactive visualization of promoter capture Hi-C datasets. *Bioinformatics* **32**, 2511–2513 (2016).
153. Marzi, S. J. *et al.* Tissue-specific patterns of allelically-skewed DNA methylation. *Epigenetics* **11**, 24–35 (2016).
154. Ramasamy, A. *et al.* Genetic variability in the regulation of gene expression in ten regions of the human brain. *Nat. Neurosci.* **17**, 1418–1428 (2014).
155. Lonsdale, J. *et al.* The Genotype-Tissue Expression (GTEx) project. *Nat. Genet.* **45**, 580–585 (2013).
156. Walsh, A. M. *et al.* Integrative genomic deconvolution of rheumatoid arthritis GWAS loci into gene and cell type associations. *Genome Biol.* **17**, 79 (2016).
157. Hinrichs, A. S. *et al.* The UCSC Genome Browser Database: update 2006. *Nucleic Acids Res.* **34**, D590-598 (2006).

158. John, S. *et al.* Chromatin accessibility pre-determines glucocorticoid receptor binding patterns. *Nat. Genet.* **43**, 264–268 (2011).
159. Sabo, P. J. *et al.* Discovery of functional noncoding elements by digital analysis of chromatin structure. *Proc. Natl. Acad. Sci. U. S. A.* **101**, 16837–16842 (2004).
160. Quinlan, A. R. & Hall, I. M. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* **26**, 841–842 (2010).
161. Eckhardt, F. *et al.* DNA methylation profiling of human chromosomes 6, 20 and 22. *Nat. Genet.* **38**, 1378–1385 (2006).
162. Benjamini, Y. & Yekutieli, D. The control of the false discovery rate in multiple testing under dependency. *Ann. Stat.* **29**, 1165–1188 (2001).
163. Lowe, R. & Rakyan, V. K. Marmal-aid – a database for Infinium HumanMethylation450. *BMC Bioinformatics* **14**, 359 (2013).
164. Aryee, M. J. *et al.* Minfi: a flexible and comprehensive Bioconductor package for the analysis of Infinium DNA methylation microarrays. *Bioinforma. Oxf. Engl.* **30**, 1363–1369 (2014).
165. Jaffe, A. E. & Irizarry, R. A. Accounting for cellular heterogeneity is critical in epigenome-wide association studies. *Genome Biol.* **15**, R31 (2014).
166. Benjamini, Y. & Hochberg, Y. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *J. R. Stat. Soc. Ser. B Methodol.* **57**, 289–300 (1995).
167. Lowe, R., Slodkiewicz, G., Goldman, N. & Rakyan, V. K. The human blood DNA methylome displays a highly distinctive profile compared with other somatic tissues. *Epigenetics Off. J. DNA Methylation Soc.* **0** (2015).
doi:10.1080/15592294.2014.1003744

168. Hirohata, S. *et al.* Accelerated generation of CD14+ monocyte-lineage cells from the bone marrow of rheumatoid arthritis patients. *Arthritis Rheum.* **39**, 836–843 (1996).
169. Yin, Y. *et al.* Normalization of CD4+ T cell metabolism reverses lupus. *Sci. Transl. Med.* **7**, 274ra18-274ra18 (2015).
170. Singh, N. & Cohen, P. L. The T cell in Sjogren's syndrome: Force majeure, not spectateur. *J. Autoimmun.* **39**, 229–233 (2012).
171. Coit, P. *et al.* Genome-wide DNA methylation study suggests epigenetic accessibility and transcriptional poising of interferon-regulated genes in naïve CD4+ T cells from lupus patients. *J. Autoimmun.* **43**, 78–84 (2013).
172. Liu, Y. *et al.* Epigenome-wide association data implicate DNA methylation as an intermediary of genetic risk in rheumatoid arthritis. *Nat. Biotechnol.* **31**, 142–147 (2013).
173. Altorok, N. *et al.* Genome-wide DNA methylation patterns in naive CD4+ T cells from patients with primary Sjögren's syndrome. *Arthritis Rheumatol. Hoboken NJ* **66**, 731–739 (2014).
174. Huynh, J. L. *et al.* Epigenome-wide differences in pathology-free regions of multiple sclerosis-affected brains. *Nat. Neurosci.* **17**, 121–130 (2014).
175. Xavier, A. L., Menezes, J. R. L., Goldman, S. A. & Nedergaard, M. Fine-tuning the central nervous system: microglial modelling of cells and synapses. *Philos. Trans. R. Soc. Lond. B. Biol. Sci.* **369**, 20130593 (2014).
176. Lowe, R. & Rakyan, V. K. Correcting for cell-type composition bias in epigenome-wide association studies. *Genome Med.* **6**, 23 (2014).
177. Charles A Janeway, J., Travers, P., Walport, M. & Shlomchik, M. J. B-cell activation by armed helper T cells. (2001).

178. Fang, F. *et al.* Breast cancer methylomes establish an epigenomic foundation for metastasis. *Sci. Transl. Med.* **3**, 75ra25 (2011).
179. Kibriya, M. G. *et al.* A genome-wide DNA methylation study in colorectal carcinoma. *BMC Med. Genomics* **4**, 50 (2011).
180. Laczmanska, I. *et al.* Protein tyrosine phosphatase receptor-like genes are frequently hypermethylated in sporadic colorectal cancer. *J. Hum. Genet.* **58**, 11–15 (2013).
181. Arai, E. *et al.* Single-CpG-resolution methylome analysis identifies clinicopathologically aggressive CpG island methylator phenotype clear cell renal cell carcinomas. *Carcinogenesis* **33**, 1487–1493 (2012).
182. Barreau, O. *et al.* Identification of a CpG island methylator phenotype in adrenocortical carcinomas. *J. Clin. Endocrinol. Metab.* **98**, E174–184 (2013).
183. ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature* **489**, 57–74 (2012).
184. Adams, D. *et al.* BLUEPRINT to decode the epigenetic signature written in blood. *Nat. Biotechnol.* **30**, 224–226 (2012).
185. Schleinitz, N., Vély, F., Harlé, J.-R. & Vivier, E. Natural killer cells in human autoimmune diseases. *Immunology* **131**, 451 (2010).
186. Kinsella, R. J. *et al.* Ensembl BioMarts: a hub for data retrieval across taxonomic space. *Database J. Biol. Databases Curation* **2011**, bar030 (2011).
187. Dörner, T. & Radbruch, A. Antibodies and B cell memory in viral immunity. *Immunity* **27**, 384–392 (2007).
188. Li, J. *et al.* B lymphocytes from early vertebrates have potent phagocytic and microbicidal abilities. *Nat. Immunol.* **7**, 1116–1124 (2006).

189. Parra, D. *et al.* Pivotal advance: peritoneal cavity B-1 B cells have phagocytic and microbicidal capacities and present phagocytosed antigen to CD4+ T cells. *J. Leukoc. Biol.* **91**, 525–536 (2012).
190. Shaheen, R. *et al.* A novel syndrome of hypohidrosis and intellectual disability is linked to COG6 deficiency. *J. Med. Genet.* **50**, 431–436 (2013).
191. Chen, Y. *et al.* Downregulation of TNIP1 Expression Leads to Increased Proliferation of Human Keratinocytes and Severer Psoriasis-Like Conditions in an Imiquimod-Induced Mouse Model of Dermatitis. *PLOS ONE* **10**, e0127957 (2015).
192. Cocks, B. G., de Waal Malefyt, R., Galizzi, J. P., de Vries, J. E. & Aversa, G. IL-13 induces proliferation and differentiation of human B cells activated by the CD40 ligand. *Int. Immunol.* **5**, 657–663 (1993).
193. Goldminz, A. M., Au, S. C., Kim, N., Gottlieb, A. B. & Lizzul, P. F. NF- κ B: an essential transcription factor in psoriasis. *J. Dermatol. Sci.* **69**, 89–94 (2013).
194. Chinwalla, A. T. *et al.* Initial sequencing and comparative analysis of the mouse genome. *Nature* **420**, 520–562 (2002).
195. Hardouin, S. N. & Nagy, A. Mouse models for human disease. *Clin. Genet.* **57**, 237–244 (2000).
196. Vietri Rudan, M. *et al.* Comparative Hi-C Reveals that CTCF Underlies Evolution of Chromosomal Domain Architecture. *Cell Rep.* **10**, 1297–1309 (2015).
197. Sofueva, S. *et al.* Cohesin-mediated interactions organize chromosomal domain architecture. *EMBO J.* **32**, 3119 (2013).
198. Maurano, M. T. *et al.* Systematic localization of common disease-associated variation in regulatory DNA. *Science* **337**, 1190–1195 (2012).

199. Vierstra, J. *et al.* Mouse regulatory DNA landscapes reveal global principles of cis-regulatory evolution. *Science* **346**, 1007–1012 (2014).
200. van den Boogaard, M. *et al.* A common genetic variant within SCN10A modulates cardiac SCN5A expression. *J. Clin. Invest.* **124**, 1844–1852 (2014).
201. Odom, D. T. *et al.* Tissue-specific transcriptional regulation has diverged significantly between human and mouse. *Nat. Genet.* **39**, 730–732 (2007).
202. Lunnon, K. *et al.* Methylomic profiling implicates cortical deregulation of ANK1 in Alzheimer's disease. *Nat. Neurosci.* **17**, 1164–1170 (2014).
203. De Jager, P. L. *et al.* Alzheimer's disease: early alterations in brain DNA methylation at ANK1, BIN1, RHBDF2 and other loci. *Nat. Neurosci.* **17**, 1156–1163 (2014).
204. Fujibuchi, T. *et al.* AIP1/WDR1 supports mitotic cell rounding. *Biochem. Biophys. Res. Commun.* **327**, 268–275 (2005).
205. Bamburg, J. R. & Bernstein, B. W. Actin dynamics and cofilin-actin rods in alzheimer disease. *Cytoskelet. Hoboken NJ* **73**, 477–497 (2016).
206. Derks, E. M., Ophoff, R. A. & Genetic Risk Outcome of Psychosis (GROUP). Replication and refinement of the role of rs548181 in schizophrenia: Results from a family based study. *Am. J. Med. Genet. B Neuropsychiatr. Genet.* **162**, 75–77 (2013).
207. Kang, E. *et al.* Interaction between FEZ1 and DISC1 in regulation of neuronal development and risk for schizophrenia. *Neuron* **72**, 559–571 (2011).
208. Bando, S. Y. *et al.* Complex Network Analysis of CA3 Transcriptome Reveals Pathogenic and Compensatory Pathways in Refractory Temporal Lobe Epilepsy. *PLOS ONE* **8**, e79913 (2013).

209. López-Laso, E. *et al.* Classic and late-onset neurological disease in two siblings with glutaryl-CoA dehydrogenase deficiency. *J. Inherit. Metab. Dis.* **30**, 979 (2007).
210. Yin, K.-J., Hamblin, M., Fan, Y., Zhang, J. & Chen, Y. E. Krüppel-like factors in the central nervous system: novel mediators in Stroke. *Metab. Brain Dis.* **30**, 401 (2015).
211. Nemoto, Y. *et al.* Identification and characterization of a synaptojanin 2 splice isoform predominantly expressed in nerve terminals. *J. Biol. Chem.* **276**, 41133–41142 (2001).
212. Wortmann, S. B. *et al.* Mutations in the phospholipid remodeling gene SERAC1 impair mitochondrial function and intracellular cholesterol trafficking and cause dystonia and deafness. *Nat. Genet.* **44**, 797–802 (2012).
213. Guo, J.-H. *et al.* Isolation and characterization of the human D-glyceric acidemia related glycerate kinase gene GLYCK1 and its alternatively splicing variant GLYCK2. *DNA Seq. J. DNA Seq. Mapp.* **17**, 1–7 (2006).
214. Okun, E., Griffioen, K. J. & Mattson, M. P. Toll-like receptor Signaling in Neural Plasticity and Disease. *Trends Neurosci.* **34**, 269 (2011).
215. Planells-Cases, R. *et al.* Subunit composition of VRAC channels determines substrate specificity and cellular resistance to Pt-based anti-cancer drugs. *EMBO J.* **34**, 2993–3008 (2015).
216. Kauraniemi, P. & Kallioniemi, A. Activation of multiple cancer-associated genes at the ERBB2 amplicon in breast cancer. *Endocr. Relat. Cancer* **13**, 39–49 (2006).

217. Söderman, J., Berglind, L. & Almer, S. Gene Expression-Genotype Analysis Implicates GSDMA, GSDMB, and LRRC3C as Contributors to Inflammatory Bowel Disease Susceptibility. *BioMed Res. Int.* **2015**, (2015).
218. Däbritz, J. *et al.* Reprogramming of monocytes by GM-CSF contributes to regulatory immune functions during intestinal inflammation. *J. Immunol. Baltim. Md 1950* **194**, 2424–2438 (2015).
219. Roberts, R. L. *et al.* Confirmation of association of IRGM and NCF4 with ileal Crohn's disease in a population-based cohort. *Genes Immun.* **9**, 561–565 (2008).
220. Trentham, D. E. *et al.* Effects of oral administration of type II collagen on rheumatoid arthritis. *Science* **261**, 1727–1730 (1993).
221. Barnett, M. L. *et al.* Treatment of rheumatoid arthritis with oral type II collagen. Results of a multicenter, double-blind, placebo-controlled trial. *Arthritis Rheum.* **41**, 290–297 (1998).
222. Carmona, F. D. *et al.* A Genome-wide Association Study Identifies Risk Alleles in Plasminogen and P4HA2 Associated with Giant Cell Arteritis. *Am. J. Hum. Genet.* **100**, 64–74 (2017).
223. Shiomi, A., Usui, T. & Mimori, T. GM-CSF as a therapeutic target in autoimmune diseases. *Inflamm. Regen.* **36**, 8 (2016).
224. Deiß, A., Brecht, I., Haarmann, A. & Buttmann, M. Treating multiple sclerosis with monoclonal antibodies: a 2013 update. *Expert Rev. Neurother.* **13**, 313–335 (2013).
225. Bowes, J. *et al.* Dense genotyping of immune-related susceptibility loci reveals new insights into the genetics of psoriatic arthritis. *Nat. Commun.* **6**,

226. Chen, X. *et al.* RNASET2 tag SNP but not CCR6 polymorphisms is associated with autoimmune thyroid diseases in the Chinese Han population. *BMC Med. Genet.* **16**, 11 (2015).
227. Kochi, Y. *et al.* A regulatory variant in CCR6 is associated with rheumatoid arthritis susceptibility. *Nat. Genet.* **42**, 515–519 (2010).
228. Plenge, R. M. *et al.* TRAF1–C5 as a Risk Locus for Rheumatoid Arthritis — A Genomewide Study. *N. Engl. J. Med.* **357**, 1199–1209 (2007).
229. Abdul-Sater, A. A. *et al.* The signaling adaptor TRAF1 negatively regulates Toll-like receptor signaling and this underlies its role in rheumatic disease. *Nat. Immunol.* **18**, 26–35 (2017).
230. Chen, M., Daha, M. R. & Kallenberg, C. G. M. The complement system in systemic autoimmune disease. *J. Autoimmun.* **34**, J276–J286 (2010).
231. Miura, G. Drug repurposing: Down goes Ikaros. *Nat. Chem. Biol.* **10**, 86–86 (2014).
232. Lu, G. *et al.* The Myeloma Drug Lenalidomide Promotes the Cereblon-Dependent Destruction of Ikaros Proteins. *Science* **343**, 305–309 (2014).
233. Krönke, J. *et al.* Lenalidomide Causes Selective Degradation of IKZF1 and IKZF3 in Multiple Myeloma Cells. *Science* **343**, 301–305 (2014).
234. Diamanti, A. *et al.* The clinical implications of thalidomide in inflammatory bowel diseases. *Expert Rev. Clin. Immunol.* **11**, 699–708 (2015).
235. Wang, L., Jia, P., Wolfinger, R. D., Chen, X. & Zhao, Z. Gene set analysis of genome-wide association studies: methodological issues and perspectives. *Genomics* **98**, (2011).
236. Homuth, G. *et al.* Extensive alterations of the whole-blood transcriptome are associated with body mass index: results of an mRNA profiling study

- involving two large population-based cohorts. *BMC Med. Genomics* **8**, (2015).
237. Moran, S., Arribas, C. & Esteller, M. Validation of a DNA methylation microarray for 850,000 CpG sites of the human genome enriched in enhancer sequences. *Epigenomics* **8**, 389–399 (2016).
238. Wagner, J. R. *et al.* The relationship between DNA methylation, genetic and expression inter-individual variation in untransformed human fibroblasts. *Genome Biol.* **15**, R37 (2014).
239. Clark, C. *et al.* A Comparison of the Whole Genome Approach of MeDIP-Seq to the Targeted Approach of the Infinium HumanMethylation450 BeadChip® for Methylome Profiling. *PLoS ONE* **7**, e50233 (2012).
240. Staunstrup, N. H. *et al.* Genome-wide DNA methylation profiling with MeDIP-seq using archived dried blood spots. *Clin. Epigenetics* **8**, 81 (2016).
241. van Dongen, J. *et al.* Genetic and environmental influences interact with age and sex in shaping the human methylome. *Nat. Commun.* **7**, 11115 (2016).
242. Mendelson, M. M. *et al.* Association of Body Mass Index with DNA Methylation and Gene Expression in Blood Cells and Relations to Cardiometabolic Disease: A Mendelian Randomization Approach. *PLoS Med.* **14**, e1002215 (2017).
243. Ligthart, S. *et al.* DNA methylation signatures of chronic low-grade inflammation are associated with complex diseases. *Genome Biol.* **17**, 255 (2016).
244. Widschwendter, M. *et al.* Epigenetic stem cell signature in cancer. *Nat. Genet.* **39**, 157–158 (2007).

245. Köeferle, A., Stricker, S. H. & Beck, S. Brave new epigenomes: the dawn of epigenetic engineering. *Genome Med.* **7**, 59 (2015).
246. Lewis, J., Breeze, C. E., Charlesworth, J., Maclaren, O. J. & Cooper, J. Where next for the reproducibility agenda in computational biology? *BMC Syst. Biol.* **10**, 52 (2016).
247. Li, M. J., Wang, L. Y., Xia, Z., Sham, P. C. & Wang, J. GWAS3D: Detecting human regulatory variants by integrative analysis of genome-wide associations, chromosome interactions and histone modifications. *Nucleic Acids Res.* **41**, W150-158 (2013).
248. Visscher, P. M. *et al.* 10 Years of GWAS Discovery: Biology, Function, and Translation. *Am. J. Hum. Genet.* **101**, 5–22 (2017).
249. Manolio, T. A. Genomewide Association Studies and Assessment of the Risk of Disease. *N. Engl. J. Med.* **363**, 166–176 (2010).
250. McCarthy, M. I. & MacArthur, D. G. Human disease genomics: from variants to biology. *Genome Biol.* **18**, (2017).
251. Mali, P. *et al.* RNA-Guided Human Genome Engineering via Cas9. *Science* **339**, 823–826 (2013).
252. Corradin, O. & Scacheri, P. C. Enhancer variants: evaluating functions in common disease. *Genome Med.* **6**, (2014).
253. Ernst, J. & Kellis, M. ChromHMM: automating chromatin state discovery and characterization. *Nat. Methods* **9**, 215
254. Edwards, S. L., Beesley, J., French, J. D. & Dunning, A. M. Beyond GWASs: Illuminating the Dark Road from Association to Function. *Am. J. Hum. Genet.* **93**, 779–797 (2013).

255. Kulaeva, O. I., Nizovtseva, E. V., Polikanov, Y. S., Ulianov, S. V. & Studitsky, V. M. Distant activation of transcription: mechanisms of enhancer action. *Mol. Cell. Biol.* **32**, 4892–4897 (2012).
256. Williamson, I., Hill, R. E. & Bickmore, W. A. Enhancers: from developmental genetics to the genetics of common human disease. *Dev. Cell* **21**, 17–19 (2011).

6 Appendices

UCSC Genome Browser on Mouse Dec. 2011 (GRCm38/mm10) Assembly

move <<< << < > >> >>> zoom in 1.5x 3x 10x base zoom out 1.5x 3x 10x 100x

chr8:84,499,999-85,500,001 1,000,003 bp.

go

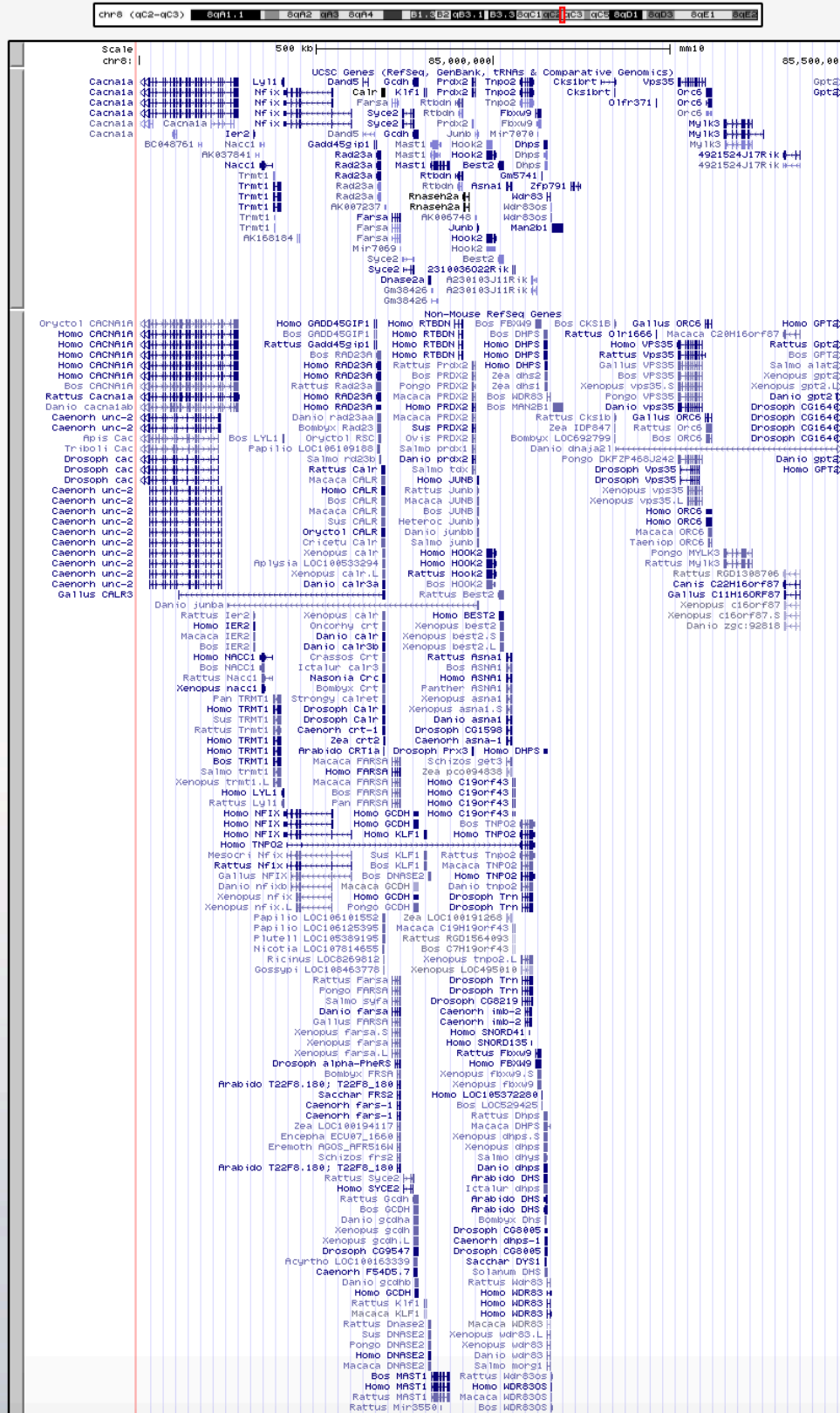


Figure S1: Gcdh locus homology. Human gene positions are shown for the Gcdh locus in mouse (supporting figure for mNSC 4C-seq **figure 5.4**). Gene positions for other species are also included.

[illegible]

Figure S2: Ppm1m locus homology. Human gene positions are shown for the Ppm1m locus in mouse (supporting figure for mNSC 4C-seq **figure 5.6**). Gene positions for other species are also included.

mNSC 4C-seq primers	
Csp6l side primers	Sequence (Illumina adaptor-4C PCR primer).
chr5:3869961 6- 38699311_Cs p6l	CAAGCAGAAGACGGCATACGAGATCGGTCTCGGCATTCTGCTGAACCG CTCTCCGATCT-GTCATTCTTACTTTGCAGGC
chr8:2425350 7- 24253114_Cs p6l	CAAGCAGAAGACGGCATACGAGATCGGTCTCGGCATTCTGCTGAACCG CTCTCCGATCT-TAACCCAGTTAGTGCTGTGC
chr8:8741133 4- 87410950_Cs p6l	CAAGCAGAAGACGGCATACGAGATCGGTCTCGGCATTCTGCTGAACCG CTCTCCGATCT-TTGTAATTCTGTTCTTCGCA
chr9:3657673 3- 36576228_Cs p6l	CAAGCAGAAGACGGCATACGAGATCGGTCTCGGCATTCTGCTGAACCG CTCTCCGATCT-AAAGAGAAGAGGTATTGAGTTAGG
chr9:1061012 43- 106101777_C sp6l	CAAGCAGAAGACGGCATACGAGATCGGTCTCGGCATTCTGCTGAACCG CTCTCCGATCT-TTTCTTTGGCCTTCTGTAAG
chr17:601423 5- 6013909_Csp 6l	CAAGCAGAAGACGGCATACGAGATCGGTCTCGGCATTCTGCTGAACCG CTCTCCGATCT-CATCTCCTGTAGGCCATAAC
DpnII side primers	Sequence (Illumina adaptor-4C PCR primer).
chr5:3869961 6- 38699311_Dp nII	AATGATACGGCGACCACCGAGATCTACACTCTTCCCTACACGACGCTCT TCCGATCT-GGGGCAGTTAGTGTGTGATC
chr8:2425350 7- 24253114_Dp nII	AATGATACGGCGACCACCGAGATCTACACTCTTCCCTACACGACGCTCT TCCGATCT-GCCAAGACTCGCTCTTGATC
chr8:8741133	AATGATACGGCGACCACCGAGATCTACACTCTTCCCTACACGACGCTCT

4-87410950_Dp nll	TCCGATCT-AATTGGAAAGACTGATGATC
chr9:3657673 3-36576228_Dp nll	AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGCTCT TCCGATCT-GCATTGATTACCCAGGATC
chr9:1061012 43-106101777_D pnll	AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGCTCT TCCGATCT-TCCAGACCCTCTGCCTGATC
chr17:601423 5-6013909_Dpn ll	AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGCTCT TCCGATCT-AGAATCTTCCACCTCTGATC

Table S1: sequences of PCR primers targeting mouse regions homologous to human GWAS/EWAS loci. Illumina adapters are also included in the sequence. Indexes are not necessary as the subsequent data analysis pipeline performs *in silico* deconvolution using the 20bp primer sequence. Pre-designed primers were selected from the 4C database by van de Werken et al. (2012)¹⁴⁹

BLUEPRINT 4C-seq primers	
Csp6l side primer	Sequence (Illumina adaptor segment 1-index-Illumina adaptor segment 2-4C PCR primer).
chr1:90022562-90022945Csp_NK r	CAAGCAGAAGACGGCATACGAGAT-AAGCTA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- GAGTGGTGAGGATTGAGAAG
chr17:37912171-37912498Csp_NK r	CAAGCAGAAGACGGCATACGAGAT-AAGCTA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- CTAGTTCCCATCCACCAC
chr22:37258335-37258758Csp_NK r	CAAGCAGAAGACGGCATACGAGAT-AAGCTA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- ATTTACTGCCTGTTTCATCCA
chr5:131430055-131430830Csp_N Kr	CAAGCAGAAGACGGCATACGAGAT-AAGCTA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- TCTAGTATCTCAGCCCTCCA
chr6:167534229-167534357Csp_N	CAAGCAGAAGACGGCATACGAGAT-AAGCTA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-

Kr	CAAACGTTCCCAAATCTTC
chr9:123652531-123653088Csp_N Kr	CAAGCAGAAGACGGCATAACGAGAT-AAGCTA-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GCCCAGAGAGCTAATTACTG
chr1:90022562-90022945Csp_CD 14	CAAGCAGAAGACGGCATAACGAGAT-CGTGAT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GAGTGGTGAGGATTGAGAAG
chr17:37912171-37912498Csp_CD 14	CAAGCAGAAGACGGCATAACGAGAT-CGTGAT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CTAGTTCCCATCCACCAC
chr22:37258335-37258758Csp_CD 14	CAAGCAGAAGACGGCATAACGAGAT-CGTGAT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-ATTTACTGCCTGTTTCATCCA
chr5:131430055-131430830Csp_C D14	CAAGCAGAAGACGGCATAACGAGAT-CGTGAT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-TCTAGTATCTCAGCCCTCCA
chr6:167534229-167534357Csp_C D14	CAAGCAGAAGACGGCATAACGAGAT-CGTGAT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CAAACGTTCCCAAATCTTC
chr9:123652531-123653088Csp_C D14	CAAGCAGAAGACGGCATAACGAGAT-CGTGAT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GCCCAGAGAGCTAATTACTG
chr1:90022562-90022945Csp_CD 4	CAAGCAGAAGACGGCATAACGAGAT-ACATCG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GAGTGGTGAGGATTGAGAAG
chr17:37912171-37912498Csp_CD 4	CAAGCAGAAGACGGCATAACGAGAT-ACATCG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CTAGTTCCCATCCACCAC
chr22:37258335-37258758Csp_CD 4	CAAGCAGAAGACGGCATAACGAGAT-ACATCG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-ATTTACTGCCTGTTTCATCCA
chr5:131430055-131430830Csp_C D4	CAAGCAGAAGACGGCATAACGAGAT-ACATCG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-TCTAGTATCTCAGCCCTCCA
chr6:167534229-167534357Csp_C D4	CAAGCAGAAGACGGCATAACGAGAT-ACATCG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CAAACGTTCCCAAATCTTC
chr9:123652531-123653088Csp_C D4	CAAGCAGAAGACGGCATAACGAGAT-ACATCG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GCCCAGAGAGCTAATTACTG
chr1:90022562-90022945Csp_CD 8	CAAGCAGAAGACGGCATAACGAGAT-GCCTAA-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GAGTGGTGAGGATTGAGAAG
chr17:37912171-37912498Csp_CD 8	CAAGCAGAAGACGGCATAACGAGAT-GCCTAA-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CTAGTTCCCATCCACCAC
chr22:37258335-	CAAGCAGAAGACGGCATAACGAGAT-GCCTAA-

37258758Csp_CD8	GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- ATTTACTGCCTGTTTCATCCA
chr5:131430055-131430830Csp_CD8	CAAGCAGAAGACGGCATACGAGAT-GCCTAA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- TCTAGTATCTCAGCCCTCCA
chr6:167534229-167534357Csp_CD8	CAAGCAGAAGACGGCATACGAGAT-GCCTAA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- CAAACGTTCCCAAATCTTC
chr9:123652531-123653088Csp_CD8	CAAGCAGAAGACGGCATACGAGAT-GCCTAA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- GCCCAGAGAGCTAATTACTG
chr1:90022562-90022945Csp_Bcel	CAAGCAGAAGACGGCATACGAGAT-TGGTCA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- GAGTGGTGAGGATTGAGAAG
chr17:37912171-37912498Csp_Bcel	CAAGCAGAAGACGGCATACGAGAT-TGGTCA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- CTAGTTCCCATCCACCAC
chr22:37258335-37258758Csp_Bcel	CAAGCAGAAGACGGCATACGAGAT-TGGTCA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- ATTTACTGCCTGTTTCATCCA
chr5:131430055-131430830Csp_Bcel	CAAGCAGAAGACGGCATACGAGAT-TGGTCA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- TCTAGTATCTCAGCCCTCCA
chr6:167534229-167534357Csp_Bcel	CAAGCAGAAGACGGCATACGAGAT-TGGTCA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- CAAACGTTCCCAAATCTTC
chr9:123652531-123653088Csp_Bcel	CAAGCAGAAGACGGCATACGAGAT-TGGTCA- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- GCCCAGAGAGCTAATTACTG
chr1:90022562-90022945Csp_NK	CAAGCAGAAGACGGCATACGAGAT-CACTGT- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- GAGTGGTGAGGATTGAGAAG
chr17:37912171-37912498Csp_NK	CAAGCAGAAGACGGCATACGAGAT-CACTGT- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- CTAGTTCCCATCCACCAC
chr22:37258335-37258758Csp_NK	CAAGCAGAAGACGGCATACGAGAT-CACTGT- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- ATTTACTGCCTGTTTCATCCA
chr5:131430055-131430830Csp_NK	CAAGCAGAAGACGGCATACGAGAT-CACTGT- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- TCTAGTATCTCAGCCCTCCA
chr6:167534229-167534357Csp_NK	CAAGCAGAAGACGGCATACGAGAT-CACTGT- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- CAAACGTTCCCAAATCTTC
chr9:123652531-123653088Csp_NK	CAAGCAGAAGACGGCATACGAGAT-CACTGT- GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC- GCCCAGAGAGCTAATTACTG

chr1:90022562-90022945Csp_CD14r	CAAGCAGAAGACGGCATACGAGAT-ATTGGC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GAGTGGTGAGGATTGAGAAG
chr17:37912171-37912498Csp_CD14r	CAAGCAGAAGACGGCATACGAGAT-ATTGGC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CTAGTTCCCATCCACCAC
chr22:37258335-37258758Csp_CD14r	CAAGCAGAAGACGGCATACGAGAT-ATTGGC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-ATTTACTGCCTGTTTCATCCA
chr5:131430055-131430830Csp_CD14r	CAAGCAGAAGACGGCATACGAGAT-ATTGGC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-TCTAGTATCTCAGCCCTCCA
chr6:167534229-167534357Csp_CD14r	CAAGCAGAAGACGGCATACGAGAT-ATTGGC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CAAACGTTCCCAAATCTTC
chr9:123652531-123653088Csp_CD14r	CAAGCAGAAGACGGCATACGAGAT-ATTGGC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GCCCAGAGAGCTAATTACTG
chr1:90022562-90022945Csp_CD4r	CAAGCAGAAGACGGCATACGAGAT-GATCTG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GAGTGGTGAGGATTGAGAAG
chr17:37912171-37912498Csp_CD4r	CAAGCAGAAGACGGCATACGAGAT-GATCTG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CTAGTTCCCATCCACCAC
chr22:37258335-37258758Csp_CD4r	CAAGCAGAAGACGGCATACGAGAT-GATCTG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-ATTTACTGCCTGTTTCATCCA
chr5:131430055-131430830Csp_CD4r	CAAGCAGAAGACGGCATACGAGAT-GATCTG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-TCTAGTATCTCAGCCCTCCA
chr6:167534229-167534357Csp_CD4r	CAAGCAGAAGACGGCATACGAGAT-GATCTG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CAAACGTTCCCAAATCTTC
chr9:123652531-123653088Csp_CD4r	CAAGCAGAAGACGGCATACGAGAT-GATCTG-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GCCCAGAGAGCTAATTACTG
chr1:90022562-90022945Csp_CD8r	CAAGCAGAAGACGGCATACGAGAT-TCAAGT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GAGTGGTGAGGATTGAGAAG
chr17:37912171-37912498Csp_CD8r	CAAGCAGAAGACGGCATACGAGAT-TCAAGT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CTAGTTCCCATCCACCAC
chr22:37258335-37258758Csp_CD8r	CAAGCAGAAGACGGCATACGAGAT-TCAAGT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-ATTTACTGCCTGTTTCATCCA
chr5:131430055-131430830Csp_CD	CAAGCAGAAGACGGCATACGAGAT-TCAAGT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-

D8r	TCTAGTATCTCAGCCCTCCA
chr6:167534229-167534357Csp_C D8r	CAAGCAGAAGACGGCATACGAGAT-TCAAGT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CAAACGTTCCCAAATCTTC
chr9:123652531-123653088Csp_C D8r	CAAGCAGAAGACGGCATACGAGAT-TCAAGT-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GCCCAGAGAGCTAATTACTG
chr1:90022562-90022945Csp_Bc elr	CAAGCAGAAGACGGCATACGAGAT-CTGATC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GAGTGGTGAGGATTGAGAAG
chr17:37912171-37912498Csp_Bc elr	CAAGCAGAAGACGGCATACGAGAT-CTGATC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CTAGTTCCCATCCACCAC
chr22:37258335-37258758Csp_Bc elr	CAAGCAGAAGACGGCATACGAGAT-CTGATC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-ATTTACTGCCTGTTTCATCCA
chr5:131430055-131430830Csp_B celr	CAAGCAGAAGACGGCATACGAGAT-CTGATC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-TCTAGTATCTCAGCCCTCCA
chr6:167534229-167534357Csp_B celr	CAAGCAGAAGACGGCATACGAGAT-CTGATC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-CAAACGTTCCCAAATCTTC
chr9:123652531-123653088Csp_B celr	CAAGCAGAAGACGGCATACGAGAT-CTGATC-GTGACTGGAGTTCAGACGTGTGCTCTTCCGATC-GCCCAGAGAGCTAATTACT
DpnII side primer	Sequence (Illumina adaptor-4C PCR primer).
chr1:90022562-90022945_DpnII	AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGC TCTTCCGATCT-AGTTGGGTATTCAATTTGATC
chr17:37912171-37912498_DpnII	AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGC TCTTCCGATCT-TTTCCCCAGAGTCAGGGATC
chr22:37258335-37258758_DpnII	AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGC TCTTCCGATCT-TGGGTGTGAAACAGGAGATC
chr5:131430055-131430830_DpnII	AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGC TCTTCCGATCT-GTTCTCTCTCTCACATGATC
chr6:167534229-167534357_DpnII	AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGC TCTTCCGATCT-CAGACAATAGGTTTCGTGATC
chr9:123652531-123653088_DpnII	AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGC TCTTCCGATCT-ATACAAGGCCCTTTGTGATC

Table S2: BLUEPRINT 4C-seq primers. Shown are sequences of PCR primers targeting human immune cell regions for 4C-seq analysis across CD14+, CD4+, CD8+, B cell and NK cell samples from the BLUEPRINT consortium. Illumina adapters are also included in the sequence. Due to the fact that more than one

sample had to be sequenced in a single HiSeq run, Illumina indexes had to be included in the primers for demultiplexing sample-specific data. The DpnII primer set was the same across all cell types. Indexes were included for each sample and replicate in the Csp6I primer.

ID	Tissue
GSM822626	Blood
GSM990067	Blood
GSM989984	Blood
GSM990028	Blood
GSM1013189	Blood
GSM1079483	Blood
GSM1052018	Blood
GSM989848	Blood
GSM1023120	Blood
GSM1079482	Blood
GSM1052191	Blood
GSM989844	Blood
GSM1051536	Blood
GSM990360	Blood
GSM990352	Blood
GSM1023121	Blood
GSM1051549	Blood
GSM990061	Blood
GSM1051605	Blood
GSM989979	Blood
GSM796665	Blood
GSM990054	Blood
GSM989927	Blood
GSM796667	Blood
GSM990273	Blood
GSM880157	Blood
GSM1051907	Blood
GSM931103	Blood
GSM989998	Blood
GSM990064	Blood
GSM990355	Blood
GSM989842	Blood

GSM990321	Blood
GSM861680	Blood
GSM1013198	Blood
GSM1051802	Blood
GSM1051813	Blood
GSM990471	Blood
GSM989837	Blood
GSM990318	Blood
GSM989990	Blood
GSM990425	Blood
GSM1051691	Blood
GSM1052131	Blood
GSM990618	Blood
GSM990101	Blood
GSM1009686	Blood
GSM989948	Blood
GSM989853	Blood
GSM1075928	Blood
TCGA-BH-A1EN-11A-23D-A13T-05	Breast
TCGA-BH-A0E1-11A-13D-A10Q-05	Breast
TCGA-BH-A0DG-11A-43D-A12R-05	Breast
TCGA-BH-A0DH-11A-31D-A10Q-05	Breast
TCGA-E9-A1NF-11A-73D-A14H-05	Breast
TCGA-E9-A1RD-11A-33D-A161-05	Breast
TCGA-A7-A0DB-11A-33D-A093-05	Breast
TCGA-BH-A1F2-11A-32D-A13T-05	Breast
GSM815144	Breast
TCGA-BH-A0HA-11A-31D-A12R-05	Breast
TCGA-BH-A0BS-11A-11D-A12R-05	Breast
TCGA-E2-A1IO-11A-21D-A145-05	Breast
TCGA-BH-A1FM-11B-23D-A13T-05	Breast
TCGA-A7-A13F-11A-42D-A12R-05	Breast
TCGA-A7-A0D9-11A-53D-A10Q-05	Breast

GSM815145	Breast
TCGA-BH-A0B8-11A-41D-A093-05	Breast
TCGA-E9-A1N6-11A-32D-A145-05	Breast
TCGA-BH-A0BM-11A-12D-A093-05	Breast
TCGA-E9-A1N5-11A-41D-A14H-05	Breast
TCGA-BH-A1F0-11B-23D-A138-05	Breast
TCGA-BH-A1EY-11B-21D-A13T-05	Breast
TCGA-BH-A0DV-11A-22D-A12R-05	Breast
TCGA-E2-A1LI-11A-23D-A161-05	Breast
GSM815146	Breast
TCGA-BH-A1F6-11B-94D-A13T-05	Breast
TCGA-BH-A0E0-11A-13D-A10Q-05	Breast
TCGA-BH-A1FR-11B-42D-A13T-05	Breast
TCGA-BH-A0DI-11A-32D-A12R-05	Breast
TCGA-E2-A1LS-11A-32D-A161-05	Breast
TCGA-BH-A0BC-11A-22D-A093-05	Breast
GSM815149	Breast
GSM849975	Breast
TCGA-E9-A1RC-11A-33D-A161-05	Breast
TCGA-E9-A1NA-11A-33D-A145-05	Breast
TCGA-E9-A1RI-11A-41D-A16A-05	Breast
TCGA-BH-A0BT-11A-21D-A12R-05	Breast
TCGA-AC-A23H-11A-12D-A161-05	Breast
GSM815142	Breast
TCGA-E9-A1ND-11A-43D-A145-05	Breast
GSM999359	Breast
TCGA-BH-A0AU-11A-11D-A12R-05	Breast

TCGA-E9-A1NE-11A-43D-A14N-05	Breast
TCGA-BH-A0BZ-11A-61D-A12R-05	Breast
GSM815148	Breast
TCGA-AC-A2FG-11A-22D-A17F-05	Breast
TCGA-E9-A1RH-11A-34D-A16A-05	Breast
TCGA-E9-A1N8-11A-42D-A145-05	Breast
TCGA-AC-A2FF-11A-13D-A17F-05	Breast
TCGA-A7-A13G-11A-51D-A13T-05	Breast
TCGA-AA-3502-11A-01D-1407-05	Colon
GSM1049405	Colon
TCGA-AA-3660-11A-01D-1721-05	Colon
GSM1049400	Colon
TCGA-A6-2682-11A-01D-1551-05	Colon
GSM1049388	Colon
GSM1049384	Colon
TCGA-AA-3509-11A-01D-1407-05	Colon
GSM1049404	Colon
GSM1049369	Colon
GSM1049387	Colon
TCGA-AA-3713-11A-01D-1721-05	Colon
TCGA-AZ-6601-11A-01D-1772-05	Colon
TCGA-A6-2675-11A-01D-1721-05	Colon
TCGA-AA-3488-11A-01D-1407-05	Colon
GSM1049402	Colon
TCGA-A6-2679-11A-01D-1551-05	Colon
GSM1049381	Colon
TCGA-G4-6295-11A-01D-1721-05	Colon
TCGA-G4-6297-11A-01D-1721-05	Colon
TCGA-A6-4107-11A-01D-	Colon

1551-05	
GSM1049374	Colon
TCGA-AZ-6600-11A-01D-1772-05	Colon
TCGA-A6-2671-11A-01D-1551-05	Colon
TCGA-G4-6314-11A-01D-1721-05	Colon
GSM1049398	Colon
TCGA-AA-3492-11A-01D-1407-05	Colon
GSM1049377	Colon
GSM1049399	Colon
GSM1049376	Colon
GSM1049379	Colon
TCGA-AZ-6599-11A-01D-1772-05	Colon
GSM1049383	Colon
TCGA-A6-2686-11A-01D-1551-05	Colon
GSM1049403	Colon
TCGA-AA-3510-11A-01D-1407-05	Colon
GSM1049393	Colon
GSM1049394	Colon
TCGA-A6-2680-11A-01D-1551-05	Colon
GSM1049386	Colon
GSM1049378	Colon
GSM1049385	Colon
TCGA-AA-3495-11A-01D-1407-05	Colon
GSM1049406	Colon
GSM1049391	Colon
GSM1049373	Colon
GSM1049370	Colon
TCGA-G4-6320-11A-01D-1721-05	Colon
TCGA-G4-6322-11A-01D-1721-05	Colon
TCGA-AA-3712-11A-01D-1721-05	Colon
TCGA-B0-5092-11A-01D-1418-05	Kidney
TCGA-BP-5173-11A-01D-1424-05	Kidney
TCGA-CJ-4902-11A-01D-1424-	Kidney

05	
TCGA-B0-5710-11A-01D-1670-05	Kidney
TCGA-B0-4713-11A-01D-1275-05	Kidney
TCGA-CJ-4918-11A-01D-1424-05	Kidney
TCGA-BQ-7053-11A-01D-1963-05	Kidney
TCGA-CJ-4882-11A-01D-1424-05	Kidney
TCGA-B0-4714-11A-01D-1275-05	Kidney
TCGA-B0-4823-11A-01D-1418-05	Kidney
TCGA-CZ-5465-11A-01D-1500-05	Kidney
TCGA-BP-5184-11A-01D-1424-05	Kidney
TCGA-B0-4707-11A-01D-1275-05	Kidney
TCGA-BP-4795-11A-01D-1418-05	Kidney
TCGA-BQ-7049-11A-01D-1963-05	Kidney
TCGA-B0-5098-11A-01D-1418-05	Kidney
TCGA-BP-5186-11A-01D-1424-05	Kidney
GSM868049	Kidney
TCGA-BQ-5886-11A-01D-1590-05	Kidney
TCGA-CZ-5456-11A-01D-1500-05	Kidney
TCGA-BQ-7059-11A-01D-1963-05	Kidney
TCGA-B0-4852-11A-01D-1500-05	Kidney
TCGA-B0-4848-11A-01D-1275-05	Kidney
TCGA-B0-5121-11A-01D-1418-05	Kidney
GSM999343	Kidney
TCGA-B0-4828-11A-01D-1275-05	Kidney
TCGA-B0-4688-11A-01D-1275-05	Kidney
TCGA-BP-5195-11A-01D-	Kidney

1424-05	
GSM868051	Kidney
TCGA-BQ-5892-11A-01D-1590-05	Kidney
TCGA-BQ-7050-11A-01D-1963-05	Kidney
TCGA-BP-5190-11A-01D-1424-05	Kidney
TCGA-B0-4710-11A-02D-1500-05	Kidney
TCGA-BP-5189-11A-01D-1424-05	Kidney
TCGA-CZ-5451-11A-01D-1500-05	Kidney
TCGA-B0-5107-11A-01D-1418-05	Kidney
TCGA-BP-5177-11A-01D-1424-05	Kidney
TCGA-BP-4993-11A-01D-1418-05	Kidney
TCGA-CJ-4897-11A-01D-1424-05	Kidney
TCGA-CJ-4912-11A-01D-1424-05	Kidney
TCGA-CZ-4866-11A-01D-1500-05	Kidney
TCGA-CZ-5460-11A-01D-1500-05	Kidney
TCGA-B0-4844-11A-01D-1275-05	Kidney
TCGA-BP-4770-11A-01D-1500-05	Kidney
TCGA-BQ-5877-11A-01D-1590-05	Kidney
TCGA-B0-5083-11A-01D-1418-05	Kidney
TCGA-A4-7288-11A-01D-2137-05	Kidney
TCGA-B0-5400-11A-01D-1500-05	Kidney
TCGA-A3-3370-11A-01D-1418-05	Kidney
TCGA-B0-4816-11A-02D-1500-05	Kidney
GSM999339	Liver
TCGA-DD-A116-11A-11D-A132-05	Liver
TCGA-DD-A11D-11A-12D-	Liver

A132-05	
TCGA-DD-A3A3-11A-11D-A22H-05	Liver
GSM868054	Liver
TCGA-G3-A25W-11A-12D-A16X-05	Liver
TCGA-BC-A10S-11A-11D-A132-05	Liver
TCGA-DD-A3A1-11A-11D-A20Z-05	Liver
TCGA-DD-A113-11A-12D-A132-05	Liver
GSM868052	Liver
TCGA-DD-A1EG-11A-11D-A20Z-05	Liver
TCGA-BC-A10R-11A-11D-A132-05	Liver
TCGA-BD-A2L6-11A-21D-A20Z-05	Liver
TCGA-DD-A11C-11A-11D-A132-05	Liver
TCGA-DD-A119-11A-11D-A132-05	Liver
TCGA-DD-A1E9-11A-11D-A153-05	Liver
TCGA-DD-A39W-11A-11D-A20Z-05	Liver
TCGA-ES-A2HT-11A-11D-A17Z-05	Liver
TCGA-DD-A1EJ-11A-11D-A153-05	Liver
TCGA-EP-A12J-11A-11D-A132-05	Liver
TCGA-DD-A115-11A-12D-A132-05	Liver
TCGA-DD-A39Z-11A-21D-A20Z-05	Liver
TCGA-EP-A26S-11A-12D-A16X-05	Liver
TCGA-DD-A1EF-11A-11D-A132-05	Liver
TCGA-DD-A11A-11A-11D-A132-05	Liver
TCGA-BD-A3EP-11A-12D-A22H-05	Liver
TCGA-BC-A10W-11A-11D-A132-05	Liver
TCGA-FV-A2QR-11A-11D-	Liver

A20Z-05	
TCGA-BC-A10U-11A-11D-A132-05	Liver
TCGA-DD-A1EB-11A-11D-A132-05	Liver
TCGA-BC-A10X-11A-11D-A132-05	Liver
TCGA-FV-A23B-11A-11D-A16X-05	Liver
TCGA-BC-A110-11A-11D-A132-05	Liver
TCGA-DD-A1EL-11A-11D-A153-05	Liver
TCGA-ES-A2HS-11A-11D-A17Z-05	Liver
TCGA-DD-A3A2-11A-11D-A20Z-05	Liver
TCGA-G3-A25X-11A-11D-A16X-05	Liver
TCGA-DD-A114-11A-12D-A132-05	Liver
TCGA-DD-A1EI-11A-11D-A132-05	Liver
TCGA-DD-A1ED-11A-11D-A153-05	Liver
TCGA-BC-A10Z-11A-11D-A132-05	Liver
TCGA-BC-A10Q-11A-11D-A132-05	Liver
TCGA-DD-A118-11A-11D-A132-05	Liver
TCGA-BC-A10T-11A-11D-A132-05	Liver
TCGA-DD-A1EE-11A-11D-A132-05	Liver
TCGA-DD-A1EH-11A-11D-A132-05	Liver
TCGA-DD-A11B-11A-11D-A132-05	Liver
TCGA-DD-A39X-11A-11D-A20Z-05	Liver
GSM868055	Liver
TCGA-BC-A112-11A-11D-A132-05	Liver
TCGA-44-6146-11A-01D-1756-05	Lung
TCGA-18-5595-11A-01D-1633-05	Lung

TCGA-22-5485-11A-01D-1633-05	Lung
TCGA-50-6594-11A-01D-1756-05	Lung
TCGA-39-5028-11A-01D-1440-05	Lung
TCGA-44-6145-11A-01D-1756-05	Lung
TCGA-44-6778-11A-01D-1856-05	Lung
TCGA-22-5471-11A-01D-1633-05	Lung
TCGA-39-5036-11A-01D-1440-05	Lung
TCGA-50-5932-11A-01D-1756-05	Lung
TCGA-43-6771-11A-01D-1818-05	Lung
GSM999345	Lung
TCGA-22-5474-11A-01D-1633-05	Lung
TCGA-39-5037-11A-01D-1440-05	Lung
TCGA-49-4488-11A-01D-1756-05	Lung
GSM868074	Lung
TCGA-39-5011-11A-01D-1440-05	Lung
TCGA-22-5473-11A-11D-1633-05	Lung
TCGA-22-5482-11A-01D-1633-05	Lung
TCGA-33-4589-11A-01D-1440-05	Lung
GSM999358	Lung
GSM868072	Lung
TCGA-39-5034-11A-01D-1440-05	Lung
GSM868008	Lung
TCGA-33-4583-11A-01D-1440-05	Lung
TCGA-50-5930-11A-01D-1756-05	Lung
TCGA-33-4582-11A-01D-1440-05	Lung
TCGA-50-6593-11A-01D-1756-05	Lung
TCGA-22-5478-11A-11D-1633-	Lung

05	
TCGA-73-4676-11A-01D-1756-05	Lung
TCGA-44-2662-11A-01D-1551-05	Lung
TCGA-44-5643-11A-01D-1626-05	Lung
TCGA-44-2659-11A-01D-1551-05	Lung
TCGA-49-6745-11A-01D-1856-05	Lung
TCGA-39-5035-11A-01D-1440-05	Lung
TCGA-50-6592-11A-01D-1756-05	Lung
TCGA-05-5420-11A-01D-1626-05	Lung
TCGA-39-5039-11A-01D-1440-05	Lung
TCGA-43-3394-11A-01D-1551-05	Lung
TCGA-44-6148-11A-01D-1756-05	Lung
TCGA-50-5935-11A-01D-1756-05	Lung
TCGA-39-5019-11A-01D-1818-05	Lung
GSM999391	Lung
TCGA-22-5472-11A-11D-1633-05	Lung
TCGA-50-5931-11A-01D-1756-05	Lung
TCGA-44-2668-11A-01D-1551-05	Lung
TCGA-38-4632-11A-01D-1756-05	Lung
TCGA-50-5936-11A-01D-1626-05	Lung
TCGA-22-5480-11A-01D-1633-05	Lung
TCGA-44-2656-11A-01D-1551-05	Lung
TCGA-HC-7745-11A-01D-2116-05	Prostate
TCGA-G9-6367-11A-01D-1787-05	Prostate
TCGA-G9-6353-11A-02D-1963-05	Prostate

TCGA-CH-5768-11A-01D-1578-05	Prostate
TCGA-EJ-7785-11A-01D-2116-05	Prostate
TCGA-EJ-7328-11A-01D-2116-05	Prostate
TCGA-EJ-7784-11A-01D-2116-05	Prostate
TCGA-G9-6384-11A-01D-1787-05	Prostate
GSM847574	Prostate
TCGA-G9-6373-11A-01D-1787-05	Prostate
TCGA-G9-6348-11A-01D-1787-05	Prostate
TCGA-G9-6332-11A-01D-1787-05	Prostate
GSM847571	Prostate
GSM847573	Prostate
GSM937263	Prostate
TCGA-HC-7211-11A-01D-2116-05	Prostate
TCGA-EJ-7327-11A-01D-2116-05	Prostate
TCGA-CH-5763-11A-01D-1578-05	Prostate
TCGA-EJ-7125-11A-01D-1963-05	Prostate
TCGA-CH-5762-11A-01D-1578-05	Prostate
TCGA-CH-5767-11B-01D-1787-05	Prostate
GSM847569	Prostate
TCGA-CH-5766-11A-01D-1578-05	Prostate
TCGA-G9-6333-11A-01D-1963-05	Prostate
TCGA-G9-6496-11A-01D-1787-05	Prostate
TCGA-G9-6494-11A-01D-1787-05	Prostate
TCGA-G9-6356-11A-01D-1787-05	Prostate
TCGA-EJ-7782-11A-01D-2116-05	Prostate
TCGA-EJ-7786-11A-01D-2116-05	Prostate
TCGA-EJ-7794-11A-01D-2116-	Prostate

05	
TCGA-G9-6385-11A-01D-1787-05	Prostate
TCGA-HC-7737-11A-02D-2116-05	Prostate
TCGA-EJ-7781-11A-01D-2116-05	Prostate
TCGA-HC-7819-11A-01D-2116-05	Prostate
TCGA-G9-6363-11A-01D-1787-05	Prostate
TCGA-G9-6351-11A-01D-1963-05	Prostate
GSM937267	Prostate
GSM999369	Prostate
TCGA-CH-5771-11A-01D-1578-05	Prostate
TCGA-EJ-7789-11A-01D-2116-05	Prostate
TCGA-CH-5765-11A-01D-1578-05	Prostate
TCGA-G9-6362-11A-01D-1787-05	Prostate
TCGA-EJ-7123-11A-01D-1963-05	Prostate
TCGA-EJ-7317-11A-01D-2116-05	Prostate
TCGA-HC-7820-11A-01D-2116-05	Prostate
GSM937269	Prostate
TCGA-HC-7742-11A-01D-2116-05	Prostate
GSM847572	Prostate
TCGA-EJ-7792-11A-01D-2116-05	Prostate
TCGA-HC-7752-11A-01D-2116-05	Prostate
TCGA-ET-A25N-11A-01D-A16P-05	Thyroid Gland
TCGA-EM-A1CW-11A-12D-A13Z-05	Thyroid Gland
TCGA-BJ-A28X-11A-11D-A22G-05	Thyroid Gland
TCGA-ET-A3DP-11A-22D-A21B-05	Thyroid Gland
TCGA-BJ-A2N8-11A-11D-A18G-05	Thyroid Gland
TCGA-EM-A1YD-11A-11D-	Thyroid

A14Z-05	Gland
TCGA-EL-A3TB-11A-11D-A22G-05	Thyroid Gland
TCGA-EM-A1CV-11A-11D-A13Z-05	Thyroid Gland
TCGA-EL-A3ZO-11A-12D-A23O-05	Thyroid Gland
TCGA-EL-A3T6-11A-11D-A223-05	Thyroid Gland
TCGA-BJ-A2NA-11A-11D-A19K-05	Thyroid Gland
TCGA-BJ-A290-11A-11D-A17Y-05	Thyroid Gland
TCGA-E8-A2JQ-11A-11D-A19K-05	Thyroid Gland
TCGA-EL-A3T2-11A-11D-A22G-05	Thyroid Gland
TCGA-BJ-A3PU-11A-11D-A223-05	Thyroid Gland
TCGA-EM-A1CT-11A-11D-A13Z-05	Thyroid Gland
TCGA-EL-A3ZL-11A-11D-A23O-05	Thyroid Gland
TCGA-ET-A3DW-11A-11D-A19K-05	Thyroid Gland
TCGA-H2-A3RI-11A-11D-A223-05	Thyroid Gland
TCGA-EL-A3N3-11A-11D-A211-05	Thyroid Gland
TCGA-EL-A3T0-11A-12D-A22G-05	Thyroid Gland
TCGA-ET-A2MY-11B-11D-A18G-05	Thyroid Gland
TCGA-EL-A3ZP-11A-11D-A23O-05	Thyroid Gland
TCGA-EL-A3MW-11A-11D-A211-05	Thyroid Gland
TCGA-EM-A1CS-11A-21D-A13Z-05	Thyroid Gland
TCGA-EL-A3T8-11A-11D-A22G-05	Thyroid Gland
TCGA-BJ-A2N7-11A-11D-A18G-05	Thyroid Gland
TCGA-DO-A1JZ-11A-11D-A13Z-05	Thyroid Gland
TCGA-BJ-A28W-11A-11D-A16P-05	Thyroid Gland
TCGA-EL-A3H1-11A-11D-	Thyroid

A21B-05	Gland
TCGA-GE-A2C6-11A-11D-A16P-05	Thyroid Gland
TCGA-EL-A3ZK-11A-11D-A230-05	Thyroid Gland
TCGA-EL-A3N2-11A-11D-A211-05	Thyroid Gland
TCGA-EL-A3T1-11A-11D-A22G-05	Thyroid Gland
TCGA-BJ-A28T-11A-11D-A16P-05	Thyroid Gland
TCGA-EL-A3H7-11A-11D-A21B-05	Thyroid Gland
TCGA-FY-A3TY-11A-12D-A231-05	Thyroid Gland
TCGA-EL-A3MX-11A-11D-A21B-05	Thyroid Gland
TCGA-EM-A1YE-11A-11D-A14Z-05	Thyroid Gland
TCGA-EL-A3H2-11A-11D-A211-05	Thyroid Gland
TCGA-EM-A1YC-11A-11D-A14Z-05	Thyroid Gland
TCGA-EL-A3TA-11A-12D-A22G-05	Thyroid Gland
TCGA-ET-A2N5-11B-11D-A18G-05	Thyroid Gland
TCGA-EL-A3ZH-11A-11D-A230-05	Thyroid Gland
TCGA-EL-A3MY-11A-12D-A21B-05	Thyroid Gland
TCGA-BJ-A28R-11A-11D-A16P-05	Thyroid Gland
TCGA-ET-A25J-11A-01D-A16P-05	Thyroid Gland
TCGA-BJ-A2N9-11A-11D-A18G-05	Thyroid Gland
TCGA-EL-A3T3-11A-11D-A22G-05	Thyroid Gland
TCGA-EL-A3T7-11A-21D-A22G-05	Thyroid Gland

Table S3: tDMP sample IDs. tDMP samples were analysed to identify tissue-specific differentially methylated positions using dmpFinder. GEO/TCGA IDs are listed for all tDMP samples.

PUBLICATIONS

eFORGE: A TOOL FOR IDENTIFYING CELL TYPE-SPECIFIC SIGNAL IN
EPIGENOMIC DATA, BREEZE ET AL., CELL REPORTS (2016)



RESOURCE

eFORGE: A Tool for Identifying Cell Type-Specific Signal in Epigenomic Data

Charles E. Breeze²¹ , Dirk S. Paul, Jenny van Dongen, Lee M. Butcher, John C. Ambrose, James E. Barrett, Robert Lowe, Vardhman K. Rakyan, Valentina Iotchkova, Mattia Frontini, Kate Downes, Willem H. Ouwehand, Jonathan Laperle, Pierre-Étienne Jacques, Guillaume Bourque, Anke K. Bergmann, Reiner Siebert, Edo Vellenga, Sadia Saeed, Filomena Matarese, Joost H.A. Martens, Hendrik G. Stunnenberg, Andrew E. Teschendorff, Javier Herrero, Ewan Birney, Ian Dunham, Stephan Beck 

²¹ Lead Contact

Open Access DOI: <http://dx.doi.org/10.1016/j.celrep.2016.10.059> |  CrossMark

Open access funded by Wellcome Trust

Epigenome-wide association studies (EWAS) provide an alternative approach for studying human disease through consideration of non-genetic variants such as altered DNA methylation. To advance the complex interpretation of EWAS, we developed eFORGE (<http://eforge.cs.ucl.ac.uk/>), a new standalone and web-based tool for the analysis and interpretation of EWAS data. eFORGE determines the cell type-specific regulatory component of a set of EWAS-identified differentially methylated positions. This is achieved by detecting enrichment of overlap with DNase I hypersensitive sites across 454 samples (tissues, primary cell types, and cell lines) from the ENCODE, Roadmap Epigenomics, and BLUEPRINT projects. Application of eFORGE to 20 publicly available EWAS datasets identified disease-relevant cell types for several common diseases, a stem cell-like signature in cancer, and demonstrated the ability to detect cell-composition effects for EWAS performed on heterogeneous tissues. Our approach bridges the gap between large-scale epigenomics data and EWAS-derived target selection to yield insight into disease etiology.

Cell Reports

Search

All Content

Advanced Search

Cell Reports

All Journals

Explore

Online Now

Current Issue

Archive

Journal Information

For Authors

< Previous

Volume 17, Issue 8
November 15, 2016
[Open Access](#)

Select All

Export Citations

Email a Colleague

Add to Reading List

Reports

☐

Structural Basis for the Activation of IKK1/α



On the cover: Different immune cell types contribute to the etiology of complex diseases. However, the degree to which specific cell types play a role in complex diseases such as rheumatoid arthritis is unknown. To identify the cell types underlying complex diseases, Breeze et al. developed the eFORGE software tool. This image shows erythrocytes and leukocytes, highlighting CD4+ T cells, CD8+ T cells, and CD19+ B cells.

Enlarge Cover

252



GENETIC AND ENVIRONMENTAL INFLUENCES INTERACT WITH AGE AND SEX
IN SHAPING THE HUMAN METHYLOME, VAN DONGEN ET AL.,
NATURE COMMUNICATIONS (2016)



ARTICLE

Received 10 Jun 2015 | Accepted 23 Feb 2016 | Published 7 Apr 2016

DOI: 10.1038/ncomms11115

OPEN

Genetic and environmental influences interact with age and sex in shaping the human methylome

Jenny van Dongen^{1,*}, Michel G. Nivard^{1,*}, Gonneke Willemsen¹, Jouke-Jan Hottenga¹, Quinta Helmer¹, Conor V. Dolan¹, Erik A. Ehli², Gareth E. Davies², Maarten van Iterson³, Charles E. Breeze⁴, Stephan Beck⁴, BIOS Consortium[†], H. Eka Suchiman³, Rick Jansen⁵, Joyce B. van Meurs⁶, Bastiaan T. Heijmans^{3,**}, P. Eline Slagboom^{3,**} & Dorret I. Boomsma^{1,**}

The methylome is subject to genetic and environmental effects. Their impact may depend on sex and age, resulting in sex- and age-related physiological variation and disease susceptibility. Here we estimate the total heritability of DNA methylation levels in whole blood and estimate the variance explained by common single nucleotide polymorphisms at 411,169 sites in 2,603 individuals from twin families, to establish a catalogue of between-individual variation in DNA methylation. Heritability estimates vary across the genome (mean = 19%) and interaction analyses reveal thousands of sites with sex-specific heritability as well as sites where the environmental variance increases with age. Integration with previously published data illustrates the impact of genome and environment across the lifespan at methylation sites associated with metabolic traits, smoking and ageing. These findings demonstrate that our catalogue holds valuable information on locations in the genome where methylation variation between people may reflect disease-relevant environmental exposures or genetic variation.

THE INTERNATIONAL HUMAN EPIGENOME CONSORTIUM: A BLUEPRINT FOR SCIENTIFIC COLLABORATION AND DISCOVERY, STUNNENBERG ET AL., CELL (2016)



The International Human Epigenome Consortium: A Blueprint for Scientific Collaboration and Discovery

Hendrik G. Stunnenberg,^{1,*} The International Human Epigenome Consortium,⁴ and Martin Hirst^{2,3,*}

¹Department of Molecular Biology, Faculties of Science and Medicine, Radboud University, Nijmegen, 6525AG, the Netherlands

²Department of Microbiology and Immunology, Michael Smith Laboratories, University of British Columbia, Vancouver, BC V6T 1Z4, Canada

³Canada's Michael Smith Genome Sciences Centre, BC Cancer Agency, Vancouver, BC V5Z 4S6, Canada

⁴<http://ihc-epigenomes.org/>

*Correspondence: h.stunnenberg@ncmls.ru.nl (H.G.S.), mhirst@bcgsc.ca (M.H.)

<http://dx.doi.org/10.1016/j.cell.2016.11.007>

The International Human Epigenome Consortium (IHEC) coordinates the generation of a catalog of high-resolution reference epigenomes of major primary human cell types. The studies now presented (see the Cell Press IHEC web portal at <http://www.cell.com/consortium/IHEC>) highlight the coordinated achievements of IHEC teams to gather and interpret comprehensive epigenomic datasets to gain insights in the epigenetic control of cell states relevant for human health and disease.

TISSUE-INDEPENDENT AND TISSUE-SPECIFIC PATTERNS OF DNA METHYLATION ALTERATION IN CANCER, CHEN ET AL., EPIGENETICS AND CHROMATIN (2016)



Epigenetics & Chromatin

RESEARCH

Open Access



Tissue-independent and tissue-specific patterns of DNA methylation alteration in cancer

Yuting Chen^{1,2}, Charles E. Breeze³, Shao Zhen¹, Stephan Beck³ and Andrew E. Teschendorff^{1,4,5*}

Abstract

Background: There is growing evidence that DNA methylation alterations contribute to carcinogenesis. While cancer tissue exhibits widespread DNA methylation changes, the proportion of tissue-specific versus tissue-independent DNA methylation alterations in cancer is unclear. In addition, it is unknown which factors determine the patterns of aberrant DNA methylation in cancer.

Results: Using HumanMethylation450 BeadChips (450k), we here analyze genome-wide DNA methylation patterns of ten types of fetal tissue, in addition to matched normal-cancer data for corresponding tissue types, encompassing over 3000 samples. We demonstrate that the level of aberrant cancer DNA methylation in gene promoters and gene bodies is highly correlated between cancer types. We estimate that up to 60 % of the DNA methylation variation in a cancer genome of a given tissue type is explained by the corresponding variation in a cancer genome of another type, implying that much of the cancer DNA methylation landscape is tissue independent. We further show that histone marks in normal cells are better predictors of aberrant cancer DNA methylation than the corresponding signals in human embryonic stem cells. We build predictors of cancer DNA methylation patterns and show that although inclusion of three histone marks (H3K4me3, H3K27me3 and H3K36me3) improves model accuracy, the bivalent marks are the most predictive. Finally, we show that chromatin accessibility of gene promoters in normal tissue dictates the promoter's propensity to acquire aberrant DNA methylation in cancer in so far as it determines its level of DNA methylation in normal tissue.

Conclusions: Our data show that a considerable fraction of the aberrant cancer DNA methylation landscape results from a mechanism that is largely tissue specific. Histone marks as specified in the normal cell of origin provide highly predictive models of aberrant cancer DNA methylation and outperform those derived from the same marks in hESCs.

EPIGENETIC REPROGRAMMING OF FALLOPIAN TUBE FIMBRIAE IN BRCA
MUTATION CARRIERS DEFINES EARLY OVARIAN CANCER EVOLUTION,
BARTLETT ET AL., NATURE COMMUNICATIONS (2016)



ARTICLE

Received 24 Nov 2015 | Accepted 14 Apr 2016 | Published 24 May 2016

DOI: 10.1038/ncomms11620

OPEN

Epigenetic reprogramming of fallopian tube fimbriae in *BRCA* mutation carriers defines early ovarian cancer evolution

Thomas E. Bartlett^{1,2,*}, Kantaraja Chindera^{1,*}, Jacqueline McDermott¹, Charles E. Breeze³, William R. Cooke¹, Allison Jones¹, Daniel Reisel¹, Smita T. Karegodar¹, Rupali Arora⁴, Stephan Beck³, Usha Menon¹, Louis Dubeau⁵ & Martin Widschwendter¹

The exact timing and contribution of epigenetic reprogramming to carcinogenesis are unclear. Women harbouring *BRCA1/2* mutations demonstrate a 30–40-fold increased risk of high-grade serous extra-uterine Müllerian cancers (HGSEMC), otherwise referred to as ‘ovarian carcinomas’, which frequently develop from fimbrial cells but not from the proximal portion of the fallopian tube. Here we compare the DNA methylome of the fimbrial and proximal ends of the fallopian tube in *BRCA1/2* mutation carriers and non-carriers. We show that the number of CpGs displaying significant differences in methylation levels between fimbrial and proximal fallopian tube segments are threefold higher in *BRCA* mutation carriers than in controls, correlating with overexpression of activation-induced deaminase in their fimbrial epithelium. The differentially methylated CpGs accurately discriminate HGSEMCs from non-serous subtypes. Epigenetic reprogramming is an early pre-malignant event integral to *BRCA1/2* mutation-driven carcinogenesis. Our findings may provide a basis for cancer-preventative strategies.

A COMPARISON OF REFERENCE-BASED ALGORITHMS FOR CORRECTING CELL-TYPE HETEROGENEITY IN EPIGENOME-WIDE ASSOCIATION STUDIES, TESCHENDORFF ET AL., BMC BIOINFORMATICS 2017

BMC Bioinformatics

METHODOLOGY ARTICLE

Open Access



A comparison of reference-based algorithms for correcting cell-type heterogeneity in Epigenome-Wide Association Studies

Andrew E. Teschendorff^{1,2,3*}, Charles E. Breeze⁴, Shijie C. Zheng^{1,5} and Stephan Beck⁴

Abstract

Background: Intra-sample cellular heterogeneity presents numerous challenges to the identification of biomarkers in large Epigenome-Wide Association Studies (EWAS). While a number of reference-based deconvolution algorithms have emerged, their potential remains underexplored and a comparative evaluation of these algorithms beyond tissues such as blood is still lacking.

Results: Here we present a novel framework for reference-based inference, which leverages cell-type specific DNase Hypersensitive Site (DHS) information from the NIH Epigenomics Roadmap to construct an improved reference DNA methylation database. We show that this leads to a marginal but statistically significant improvement of cell-count estimates in whole blood as well as in mixtures involving epithelial cell-types. Using this framework we compare a widely used state-of-the-art reference-based algorithm (called constrained projection) to two non-constrained approaches including CIBERSORT and a method based on robust partial correlations. We conclude that the widely-used constrained projection technique may not always be optimal. Instead, we find that the method based on robust partial correlations is generally more robust across a range of different tissue types and for realistic noise levels. We call the combined algorithm which uses DHS data and robust partial correlations for inference, EpiDISH (*Epigenetic Dissection of Intra-Sample Heterogeneity*). Finally, we demonstrate the added value of EpiDISH in an EWAS of smoking.

Conclusions: Estimating cell-type fractions and subsequent inference in EWAS may benefit from the use of non-constrained reference-based cell-type deconvolution methods.

WHERE NEXT FOR THE REPRODUCIBILITY AGENDA IN COMPUTATIONAL BIOLOGY?, LEWIS ET AL., BMC SYSTEMS BIOLOGY (2016)

BMC Systems Biology

CORRESPONDENCE

Open Access

Where next for the reproducibility agenda in computational biology?



Joanna Lewis^{1,2*†}, Charles E. Breeze^{3†}, Jane Charlesworth⁴, Oliver J. Maclaren^{5,6} and Jonathan Cooper⁷

[†]Equal contributors

Abstract

Background: The concept of reproducibility is a foundation of the scientific method. With the arrival of fast and powerful computers over the last few decades, there has been an explosion of results based on complex computational analyses and simulations. The reproducibility of these results has been addressed mainly in terms of exact *replicability* or numerical equivalence, ignoring the wider issue of the reproducibility of conclusions through equivalent, extended or alternative methods.

Results: We use case studies from our own research experience to illustrate how concepts of reproducibility might be applied in computational biology. Several fields have developed 'minimum information' checklists to support the full reporting of computational simulations, analyses and results, and standardised data formats and model description languages can facilitate the use of multiple systems to address the same research question. We note the importance of defining the key features of a result to be reproduced, and the expected agreement between original and subsequent results. Dynamic, updatable tools for publishing methods and results are becoming increasingly common, but sometimes come at the cost of clear communication. In general, the reproducibility of computational research is improving but would benefit from additional resources and incentives.

Conclusions: We conclude with a series of linked recommendations for improving reproducibility in computational biology through communication, policy, education and research practice. More reproducible research will lead to higher quality conclusions, deeper understanding and more valuable knowledge.

DNA METHYLOME ANALYSIS REVEALS DISTINCT EPIGENETIC PATTERNS OF ASCENDING AORTIC DISSECTION AND BICUSPID AORTIC VALVE, PAN ET AL., CARDIOVASCULAR RESEARCH (2017)

Cardiovascular Research

DNA methylome analysis reveals distinct epigenetic patterns of ascending aortic dissection and bicuspid aortic valve

Sun Pan^{1†}, Hao Lai^{1†}, Yiru Shen², Charles Breeze³, Stephan Beck³, Tao Hong¹, Chunsheng Wang^{1*}, and Andrew E. Teschendorff^{2,4,5*}

Aims

Epigenetics may mediate the effects of environmental risk factors on disease, including heart disease. Thus, measuring the DNA methylome offers the opportunity to identify novel disease biomarkers and novel insights into disease mechanisms. The DNA methylation landscape of ascending aortic dissection (AD) and bicuspid aortic valve (BAV) with aortic aneurysmal dilatation remain uncharacterized. The present study aimed to explore the genome-wide DNA methylation landscape underpinning these two diseases.

Methods and results

We used Illumina 450k DNA methylation beadarrays to analyse 21 ascending aorta samples, including 10 cases with AD, 5 with BAV and 6 healthy controls. We adjusted for intra-sample cellular heterogeneity, providing the first unbiased genome-wide exploration of the DNA methylation landscape underpinning these two diseases. We discover that both diseases are characterized by loss of DNA methylation at non-CpG sites. We validate this non-CpG hypomethylation signature with pyrosequencing. In contrast to non-CpGs, AD and BAV exhibit distinct DNA methylation landscapes at CpG sites, with BAV characterized mainly by hypermethylation of EZH2 targets. In the case of AD, integrative DNA methylation gene expression analysis reveals that AD is characterized by a dedifferentiated smooth muscle cell phenotype. Our integrative analysis further reveals hypomethylation associated overexpression of RARA in AD, a pattern which is also seen in cells exposed to smoke toxins.

Conclusion

Our data supports a model in which increased cellular proliferation in AD and BAV underpins loss of methylation at non-CpG sites. Our data further supports a model, in which AD is associated with an inflammatory vascular remodeling process, possibly mediated by the epigenome and linked to environmental risk factors such as smoking.

eFORGE ANALYSIS CODE

eFORGE is the epigenetic equivalent of the FORGE tool. For details of the FORGE analysis approach see documentation in the FORGE web version at

<http://www.1000genomes.org/forgo-analysis>

eFORGE is also available as a web tool at

<http://eforge.cs.ucl.ac.uk/>

1. The script itself is currently called eforge.pl written in Perl. It has the following Perl dependencies.

use 5.012;

use warnings;

use DBI;

use Sort::Naturally;

use Cwd;

use Storable;

use Getopt::Long;

2. The sqlite3 db file that stores the bitstrings. This file is called eforge_1.1.db currently.

3. A stored hash containing the parameters for the background selection. Currently two files, one with the 27k and one with the 450k annotation data (mvp_bins).

The database and the hashes are downloadable from:

<http://eforge.cs.ucl.ac.uk/?download>

4. An eforge.ini file in the same directory as the script. Edit this to provide the directory in which the database and hash are stored.

5. An R 3.0 installation with the "devtools" and "rCharts" packages installed. See

<https://github.com/ramnathv/rCharts>. You will need to install the latest version e.g.

```
require(devtools)
install_github('rCharts', 'ramnathv', ref = "dev")
```

The input data is one of several options.

a. A list of 450k probeids (DMPs)

b. BED format (for 450k probes)

The analysis requires a minimum of 5 DMPs (this is not a strict limit but

operationally is best).

To work DMPs currently have to be on either the 27k or the the 450k array. The script gives warnings on DMPs not found.

It also warns for background sets that do not have the right number of probes chosen, but this is really for information only.

It takes a series of command line options as follows

-f : the file to run on

-data : whether to analyse ENCODE (encode) or Epigenome Roadmap (erc) data

-label : a name for the files that are generated and for the plot titles where there is a title.

-format : for the input data format. If this is location data e.g. bed format, the probeid is obtained from the sqlite3 database.

Some of these default as described in the perldoc. Minimally the command line is:

```
eForge.pl -f probeidfile -label Some_label
```

which will by default run on Epigenome Roadmap data

OUTPUT

=====

there are several outputs generated

1. A pdf static chart, that would be good for download.
2. A d3 interactive chart.
3. A Datatables table.
4. R code files for generating the charts and the table.
5. There is also a tsv file of the results.

WEBSERVER

=====

To install the web interface, please refer to the INSTALL document in the webserver folder.#!/usr/bin/env perl

=head1 NAME

eForge.pl - Experimentally derived Functional element Overlap analysis of ReGions from EWAS.

=head1 SYNOPSIS

eForge.pl options (-f file) (-mvp mvplist)

=head1 DESCRIPTION

Analyse a set of MVPs for their overlap with DNase 1 hotspots compared to matched background MVPs.

Identifies enrichment in DHS by tissue and plots graphs and table to display. Arbitrarily a minimum of 5* MVPs is required.

Note that if no MVPs are given the script will run on A DEFAULT EWAS* as an example output.

Several outputs are made.

A straight base R graphics pdf chart of the data.

A polychart (<https://github.com/Polychart/polychart2>) interactive javascript graphic using rCharts (<http://ramnathv.github.io/rCharts/>).

A dimple (<http://dimplejs.org>) d3 interactive graphic using rCharts.

A table using the Datatables (<https://datatables.net>) plug-in for the jQuery Javascript library, again accessed through rCharts.

In each of the graphics the colouring should be consistent. Blue (p value > 0.05), light red or pink (0.05 => p value > 0.01), red or dark red (p value <= 0.01) for the 95% and 99% CIs.

Or whatever other thresholds are specified.

eForge functions, plotting options and stats are provided by eForge::eForge, eForge::ePlot and eForge::eStats modules.

=head1 OPTIONS

=over

=item B<--dataset TAG>

Set of functional data to look for enrichment. Either ENCODE data ('encode'), unconsolidated Roadmap

Epigenome data ('erc'), consolidated Roadmap Epigenome data ('erc2'), or Blueprint data ('blueprint').

erc by default.

Use --dataset ? to get a list of available datasets on your local install.

=item B<--array TAG>

Array (FKA background) is set at default to 450k array ('450k'), the Illumina Infinium HumanMethylation450 BeadChip.

For the time being, it is sufficient for MVPs to be on the 450k array. Probes within 1kb of each other will undergo filtering.

Use `--array ?` to get a list of available backgrounds on your local install.

`=item B<--label STRING>`

Supply a label that you want to use for the plotting titles, and filenames.

`=item B<--f FILENAME>`

Supply the name of a file containing a list of MVPs.

Format must be given by the `-format` flag.

If not supplied the analysis is performed either on mvps provided as probeids (cg or ch probes) in a

comma separated list through the mvps option or on a set of data from a default ewas study, namely a

set of monocyte tDMPs from Jaffe AE and Irizarry RA, Genome Biol 2014.

Note that at least 5 MVPs are required at a minimum by default.

`=item B<--mvps probe_id,probe_id...>`

Can provide the mvps as probeids in a comma separated list.

`=item B<--min_mvps INT>`

Specify the minimum number of MVPs to be allowed. Default is 5 now we are using binomial test.

`=item B<--thresh FLOAT,FLOAT>`

Alter the default binomial p value thresholds. Give a comma separate list of three e.g. 0.05,0.01 for the defaults

`=item B<--format STRING>`

If `f` is specified, specify the file format as follow:

probeid = list of mvps as probeids each on a separate line. Optionally can add other fields after the probeid which are ignored, unless the pvalue filter is specified, in which case eForge assumes that the second field is the minus log₁₀ pvalue

bed = File given is a bed file of locations (chr\tbeg\tend). bed format should be 0 based and the chromosome should be given as chrN.

However we will also accept chromosomes as just N (ensembl) and 1-based format where beg and end are the same*.

tabix = File contains MVPs in tabix format.

`=item B<--filter FLOAT>`

Set a filter on the MVPs based on the -log₁₀ pvalue. This works for files in the probeid' format.

Give a value as the lower threshold and only MVPs with $-\log_{10}$ pvalues \geq to the threshold will be analysed. Default is no filtering.

=item B<--save_stats>

Output annotation stats for the original and the random picks.

=item B<--reps INT>

The number of background matching sets to pick and analyse. Default 1000.

=item B<--proxy TAG>

Apply filter for MVPs in proximity (within 1 kb of another test MVP). With proximity filter specified, eForge will report MVPs removed due to proximity with another MVP in the list and will randomly pick one of the probes among the set of probes that are in proximity (within 1 kb of each other).

At the moment, this is a dummy flag as only one proximity filter is available for each array. It will become useful if the database and code support more than one. At the moment to turn off proximity filtering, simply specify -noproxy

=item B<--noproxy>

Turn off proximity filtering.

=item B<--depletion>

Analyse for depletion pattern instead of the default enrichment analysis. Use when dealing with datasets suspected not to overlap with DHS (or the relevant functional assay). Specifying depletion will be indicated on the label (the text "Depletion Analysis" will be added to the file label).

=item B<--noplot>

Just make the data file, don't plot.

=item B<--help|-h|-?>

Print a brief help message and exits.

=item B<--man|-m>

Print this perldoc and exit.

=back

=head1 LICENCE AND COPYRIGHT

eForge.pl Functional analysis of EWAS MVPs

Copyright (C) [2014-2015] EMBL - European Bioinformatics Institute and University College London

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; version 2 dated June, 1991 or at your option any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

A copy of the GNU General Public License is available in the source tree; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

=head1 CONTACT

Charles Breeze, C<< <c.breeze at ucl.ac.uk> >>

Javier Herrero, C<< <javier.herrero at ucl.ac.uk> >>

=head1 ACKNOWLEDGEMENTS

This software is based on the FORGE tool developed by Ian Dunham at the EMBL-EBI

Javier Herrero <javier.herrero@ucl.ac.uk>

=cut

```
use strict;
use 5.010;
use warnings;
use DBI; #database link to sqlite database
use Sort::Naturally;
use Cwd;
use Getopt::Long; #check this module
use File::Basename;
use Config::IniFiles;
use Pod::Usage;
use Scalar::Util qw(looks_like_number);
use eForge::eStats;
use eForge::ePlot;
use eForge::eForge;
use Data::UUID;
use Statistics::Multtest qw(BY);
```

```
my $cwd = getcwd;
```

```

my $dbname = "eforge_1.2.db";

my $array; # Default value
my $array_label;
my $format = 'probeid'; # Input format
my $label = 'Unnamed'; # Label for plots
my $reps = 1000;
# set binomial p values, multiple test correction is used
my $thresh; # string for command line option
my $t_marginal = 0.05; # default marginal p-value threshold
my $t_strict = 0.01; # default strict p-value threshold

my $min_num_probes = 5; # the minimum number of probes allowed for test. Set to 5 as
we have binomial p

my ($dataset, $filename, $save_probe_annotation_stats, $noplots,
    $help, $man, $proxy, $noproxy, $depletion, $filter, $out_dir, $probe_list,
    $web, $autoopen);

GetOptions (
    'dataset=s' => \$dataset,
    'save_stats|bkgrd' => \$save_probe_annotation_stats,
    'array|bkgd=s' => \$array,
    'label=s' => \$label,
    'f=s' => \$filename,
    'format=s' => \$format,
    'probes|mvps=s@' => \$probe_list,
    'min_num_probes|min_mvps=i' => \$min_num_probes,
    'noplots' => \$noplots,
    'reps=i' => \$reps,
    'thresh=s' => \$thresh,
    'proxy=s' => \$proxy,
    'noproxy' => \$noproxy,
    'depletion' => \$depletion,
    'filter=f' => \$filter,
    'out_dir=s' => \$out_dir,
    'web=s' => \$web,
    'autoopen' => \$autoopen,
    'help|h|?' => \$help,
    'man|m' => \$man,
);

pod2usage(1) if ($help);
pod2usage(-verbose => 2) if ($man);

if (!$out_dir) {
    my $ug = new Data::UUID;
    $out_dir = $ug->to_hexstring($ug->create());
}
mkdir $out_dir;

```

```
# Define the thresholds to use.
if ($thresh) {
    ($t_marginal, $t_strict) = parse_pvalue_thresholds($thresh);
}

##
=====
=====
## Connect to the DB
##
=====
=====
# This reads the config file and sets up the $datadir variable
my $dirname = dirname(__FILE__);
my $cfg = Config::IniFiles->new( -file => "$dirname/eforge.ini" );
my $datadir = $cfg->val('Files', 'datadir');

unless (-s "$datadir/$dbname") {
    die "Database $dbname not found or empty";
}
my $dsn = "dbi:SQLite:dbname=$datadir/$dbname";
my $dbh = DBI->connect($dsn, "", "") or die $DBI::errstr;
##
=====
=====

##
=====
=====
## Check the dataset against the info on the DB
##
=====
=====
my $all_datasets = get_all_datasets($dbh);
if (!defined($all_datasets)) {
    die "Empty database: no dataset loaded!\n";
} elsif (!defined($dataset)) {
    $dataset = $all_datasets->[0]->{tag};
    print "Using default dataset: [$dataset] ".$all_datasets->[0]->{name}."\n";
} elsif ($dataset eq "?") {
    print "Available datasets:\n - [" .join("\n - [", map {$_->{tag}. " " $_->{name}}
    @$all_datasets)."\n";
    exit();
} elsif (!grep {$_ eq $dataset} map {$_->{tag}} @$all_datasets) {
    die "Dataset $dataset unknown\nAvailable datasets:\n - [" .join("\n - [", map {$_-
    >{tag}. " " $_->{name}} @$all_datasets)."\n";
}
##
=====
=====
```



```

##
=====
#####
## Check the array name (A.K.A. background) against DB
##
=====
#####
my $all_arrays = get_all_arrays($dbh);
if (!defined($all_arrays)) {
    die "Empty database: no background loaded!\n";
} elsif (!defined($array)) {
    $array = $all_arrays->[0]->{tag};
    print "Using default background: [$array] ".$all_arrays->[0]->{name}."\n";
    $array_label = $all_arrays->[0]->{name};
} elsif ($array eq "?") {
    print "Available arrays:\n - [" .join("\n - [", map {$_->{tag}.} ".$_->{name}}
    @$all_arrays)."\n";
    exit();
} elsif (!grep {$_ eq $array} map {$_->{tag}} @$all_arrays) {
    die "Array $array unknown\nAvailable arrays:\n - [" .join("\n - [", map {$_->{tag}.}
    ".$_->{name}} @$all_arrays)."\n";
} else {
    foreach my $this_array (@$all_arrays) {
        if ($this_array->{tag} eq $array) {
            $array_label = $this_array->{name};
            last;
        }
    }
}
##
=====
=====

##
=====
=====
## Check the proxy_filter (A.K.A. filter) against DB
##
=====
=====
# Set proximity filter
if (defined $noproxy) {
    $proxy = undef;
} else {
    my $all_proxy_filters = get_all_proximity_filters($dbh);
    if ($all_proxy_filters->{$array}) {
        $proxy = $all_proxy_filters->{$array};
    }
}
##
=====
=====

```

```
##
=====
#####
## Append main options (depletion on/off; array; dataset) to $label
##
=====
#####
if (defined $depletion) {
    $label = "$label.depletion";
}
(my $lab = $label) =~ s/\s/_/g; # Avoid whitespaces on the label
$lab = "$lab.$array.$dataset";
##
=====
=====

##
=====
#####
## Read and process the input MVPs
##
=====
#####
warn "[".scalar(localtime())."] Processing input...\n";
# This will read the probes from the file if provided, from the probe list otherwise or use
the
# example data set as a last resort.
my $mvps = get_input_probes($filename, $probe_list);
my $original_mvps = [@mvps];
my $num_of_input_mvps = scalar(@mvps);

# Apply the proximity filter if requested
my ($proximity_excluded);
if(defined $proxy) {
    ($proximity_excluded, $mvps) = proximity_filter($dbh, $array, $mvps);
    while (my ($excluded_mvp, $mvp) = each %$proximity_excluded) {
        warn "$excluded_mvp excluded for $proxy proximity filter with $mvp\n";
    }
}

# $annotated_probes is an arrayref with probe_id, sum, bit, gene_group, cgi_group for
each input probe
my $annotated_probes = get_probe_annotations_and_overlap_for_dataset($dbh,
$dataset, $array, $mvps);
my $existing_probes = {map {$_->[0] => 1} @$annotated_probes};
$mvps = [keys %$existing_probes];

## Detect and remove the missing probes.
my $num_missing_probes = find_missing_probes($original_mvps, $existing_probes,
$proximity_excluded);
```

```
# Print summary of filtering and checks:
my $msg = "For $label, $num_of_input_mvps MVPs provided, ". scalar @$mvps.
    " retained: $num_missing_probes were not found";
if (defined $proxy) {
    $msg .= " and " . scalar(keys %$proximity_excluded) . " excluded using $proxy
proximity filter";
}
warn $msg, ".\n";

# Check we have enough MVPs left
my $num_of_valid_probes = scalar @$mvps;
if ($num_of_valid_probes < $min_num_probes) {
    die "Fewer than $min_num_probes MVPs. Analysis not run\n";
}
##
=====

# get the cell list array and the hash that connects the cells and tissues
# $samples is a hash whose keys are the $cells (short name for the cell type/lines) and
value is
# another hash with 'tissue', 'datatype', 'file' and 'acc' keys.
# IMPORTANT: $cells contains the list of cells in the order defined in the DB. This is
critical
# to correctly assign each bit to the right sample.
my ($cells, $samples) = get_samples_from_dataset($dbh, $dataset);

# unpack the bitstrings and store the overlaps by cell.
# $overlaps is a complex hash like:
# $overlaps->{'MVPS'}->{$probe_id}->{'SUM'} (total number of overlaps of this probe
with features in this dataset)
# $overlaps->{'MVPS'}->{$probe_id}->{'PARAMS'} (gene and CGI annotations for this
probe)
# $overlaps->{'CELLS'}->{$cell}->{'COUNT'} (number of input MVPs that overlap with
the signal on this sample)
# $overlaps->{'CELLS'}->{$cell}->{'MVPS'} (list of input MVPs that overlap with the
signal on this sample)
my $overlaps = process_overlaps($annotated_probes, $cells, $dataset);

# generate stats on the background selection
if (defined $save_probe_annotation_stats) {
    save_probe_annotation_stats($overlaps, $out_dir, $lab, "test");
}

# only pick background mvps matching mvps that had bitstrings originally.
#reference to hash key 'MVPS' is due to use of eforge.pm module from eForge tool
#(in subroutines process_overlaps, etc)

# Identify the feature and cpg island relationship, and then make random picks
warn "[".scalar(localtime())."] Loading the $array background...\n";
```

```

my $random_picks = get_random_matching_picks($overlaps, $array, $datadir, $reps);

#####check below lines:

# for bkgrd set need to get distribution of counts instead
# make a hash of data -> cell -> bkgrd-Set -> overlap counts
my %overlaps_per_cell; #this hash is going to store the overlaps for the random picks,
per cell

# Get the bits for the background sets and process
my $total_num_probes_in_random_picks;

warn "[".scalar(localtime())."] Running the analysis with $num_of_valid_probes
MVPs...\n";
my $count = 0;
foreach my $this_random_pick (@$random_picks) {
    warn "[".scalar(localtime())."] Repetition $count out of ".$reps."\n" if (++$count%100
== 0);
    $annotated_probes = get_probe_annotations_and_overlap_for_dataset($dbh, $dataset,
$array, $this_random_pick);

    $total_num_probes_in_random_picks += scalar @$annotated_probes;

    unless (scalar @$annotated_probes == $num_of_valid_probes) {
        warn "Random pick #$count only has " . scalar @$annotated_probes . " probes
compared to $num_of_valid_probes in the input set.\n";
    }

    my $this_pick_overlaps = process_overlaps($annotated_probes, $cells, $dataset);

    # accumulate the overlap counts by cell
    foreach my $cell (keys %{$this_pick_overlaps->{'CELLS'}}) {
        push @{$overlaps_per_cell{$cell}}, $this_pick_overlaps->{'CELLS'}->{$cell}-
>{'COUNT'};
    }

    if (defined $save_probe_annotation_stats) {
        save_probe_annotation_stats($this_pick_overlaps, $out_dir, $lab, $count);
    }
}

$dbh->disconnect();
warn "[".scalar(localtime())."] All repetitions done.\n";

warn "[".scalar(localtime())."] Calculating p-values...\n";
#Having got the test overlaps and the bkgrd overlaps now calculate p values and output
#the table to be read into R for plotting.

if (!$web) {
    open(BACKGROUND, "| gzip -9 > $out_dir/background.tsv.gz") or die "Cannot open
background.tsv";
}

```

```

my @results;
my @pvalues;
####ncmp is a function from Sort::Naturally
foreach my $cell (sort {ncmp($$samples{$a}{'tissue'}, $$samples{$b}{'tissue'}) ||
ncmp($a,$b)} @$cells){
    # above line sorts by the tissues alphabetically (from $samples hash values)

    # ultimately want a data frame of names(results)<-c("Zscore", "Cell", "Tissue", "File",
"MVPs")
    if (!$web) {
        print BACKGROUND join("\t", @{$overlaps_per_cell{$cell}}), "\n";
    }
    my $teststat = ($overlaps->{'CELLS'}->{$cell}->{'COUNT'} or 0); #number of overlaps
for the test MVPs

    # binomial pvalue, probability of success is derived from the background overlaps
over the tests for this cell
    # $backmvps is the total number of background mvps analysed
    # $tests is the number of overlaps found over all the background tests
    my $total_num_overlaps_in_random_picks;
    foreach (@{$overlaps_per_cell{$cell}}) {
        $total_num_overlaps_in_random_picks += $_;
    }
    my $p = sprintf("%.6f", $total_num_overlaps_in_random_picks /
$total_num_probes_in_random_picks);

    # binomial probability for $teststat or more hits out of $mvpcount mvps
    # sum the binomial for each k out of n above $teststat
    my $pbinom;
    if (defined $depletion) {
        foreach my $k (0 .. $teststat) {
            $pbinom += binomial($k, $num_of_valid_probes, $p);
        }
    } else {
        foreach my $k ($teststat .. $num_of_valid_probes) {
            $pbinom += binomial($k, $num_of_valid_probes, $p);
        }
    }
    if ($pbinom > 1) {
        $pbinom = 1;
    }
    # Store the p-values in natural scale (i.e. before log transformation) for FDR correction
    push(@pvalues, $pbinom);
    $pbinom = sprintf("%.2e", $pbinom);

    # Z score calculation (note: this is here only for legacy reasons. Z-scores assume
normal distribution)
    my $zscore = zscore($teststat, $overlaps_per_cell{$cell});

    my $mvp_string = "";
    $mvp_string = join(",", @{$overlaps->{'CELLS'}->{$cell}->{'MVPS'}})
        if defined $overlaps->{'CELLS'}->{$cell}->{'MVPS'};

```

```

# This gives the list of overlapping MVPs for use in the tooltips. If there are a lot of
them this can be a little useless
my ($shortcell, undef) = split('\|', $cell); # undo the concatenation from earlier to deal
with identical cell names.

push(@results, [$zscore, $pbinom, $shortcell, $$samples{$cell}{'tissue'},
$$samples{$cell}{'datatype'}, $$samples{$cell}{'file'}, $mvp_string,
$$samples{$cell}{'acc'}]);
}
if (!$web) {
  close(BACKGROUND);
}

##
=====
#####
## Correct the p-values for multiple testing using the Benjamini-Yekutieli FDR control
method
##
=====
#####
my $qvalues = BY(\@pvalues);
$qvalues = [map {sprintf("%.2e", $_)} @$qvalues];
##
=====
=====

##
=====
=====
## Write the results to a tab-separated file
##
=====
=====
my $results_filename = "$lab.chart.tsv.gz";
open(TSV, "| gzip -9 > $out_dir/$results_filename") or die "Cannot open
$out_dir/$results_filename: $!";
print TSV join("\t", "Zscore", "Pvalue", "Cell", "Tissue", "Datatype", "File", "Probe",
"Accession", "Qvalue"), "\n";
for (my $i = 0; $i < @results; $i++) {
  print TSV join("\t", @{$results[$i]}, $qvalues->[$i]), "\n";
}
close(TSV);
##
=====
=====

##
=====
=====
## Generate plots

```

```
##
=====
=====
warn "[".scalar(localtime())."] Generating plots...\n";
unless (defined $noplot){
  #Plotting and table routines
  Chart($results_filename, $lab, $out_dir, $samples, $cells, $label, $t_marginal, $t_strict,
$dataset); # basic pdf plot
  dChart($results_filename, $lab, $out_dir, $dataset, $label, $t_marginal, $t_strict, $web);
# rCharts Dimple chart
  table($results_filename, $lab, $out_dir, $web); # Datatables chart
}
##
=====
=====

warn "[".scalar(localtime())."] Done.\n";

if ($autoopen) {
  system("open $out_dir/$lab.table.html");
  system("open $out_dir/$lab.dchart.html");
  system("open $out_dir/$lab.chart.pdf");
}

#####
#####
#####
#####
##
## Sub-functions
##
#####
#####
#####
#####

=head2 parse_pvalue_thresholds

Arg[1]      : string $thresholds
Returns     : arrayref of marginal and strict thresholds (floats)
Example     : ($t_marginal, $t_strict) = parse_pvalue_thesholds("0.05,0.01");
Description : This function returns the both marginal and strict p-value thresholds as
read from
    the command line option. The input string should contain both numbers
separated by
    a comma.
Exceptions  : Dies if $thresholds is empty, does not contain numbers or are not defined
between
    0 and 1 and/or the marginal threshold is not larger or equal to the strict one.

=cut
```

```

sub parse_pvalue_thresholds {
  my ($thresh) = @_;
  my ($t_marginal, $t_strict);

  if (!$thresh) {
    die "Cannot read p-value thresholds from an empty string\n";
  }

  ($t_marginal, $t_strict) = split(",", $thresh);
  unless (looks_like_number($t_marginal) && looks_like_number($t_strict)){
    die "You must specify numerical p-value thresholds in a comma separated list\n";
  }
  unless ((1 >= $t_marginal) && ($t_marginal >= $t_strict) && ($t_strict >= 0)) {
    die "The p-value thresholds should be 1 >= T.marginal >= T.strict >= 0\n";
  }
  return ($t_marginal, $t_strict);
}

```

=head2 get_input_probes

Arg[1] : string \$filename
Arg[2] : arrayref \$probe_list
Returns : arrayref of probe IDs (string)
Example : \$mvps = get_input_probes("input.txt", undef);
Example : \$mvps = get_input_probes(undef, ["cg13430807",
"cg10480329,cg06297318,cg19301114"]);
Example : \$mvps = get_input_probes(undef, undef);
Description : This function returns the list of input probe IDs. This can come from
either
 \$filename if defined or from \$probe_list otherwise. Each element in \$probe_list
is a
 string which contains one or more probe IDs separated by commas (see
Examples).
 Falls back to the default data set from Jaffe and Irizarry.
 The set of probe IDs is checked to remove redundant entries.
Exceptions : Dies if the file is not found or cannot be opened for whatever reason.

=cut

```

sub get_input_probes {
  my ($filename, $probe_list) = @_;
  my $probes;

  if (defined $filename) {
    my $fh;
    if ($filename =~ /\.gz$/) {
      open($fh, "gunzip -c $filename |") or die "cannot open file $filename : $!";
    } elsif ($filename =~ /\.bz2$/) {
      open($fh, "bunzip2 -c $filename |") or die "cannot open file $filename : $!";
    } else {
      open($fh, "$filename") or die "cannot open file $filename : $!";
    }
    $probes = process_file($fh, $format, $dbh, $array, $filter);
  }
}

```



```

} elsif ($probe_list and @$probe_list) {
    @$probes = split(/,/ , join(',', @$probe_list));

} else{
    # Test MVPs from Liu Y et al. Nat Biotechnol 2013 Pulmonary_function.snps.bed
    (*put EWAS bedfile here)
    # If no options are given it will run on the default set of MVPs
    warn "No probe input given, so running on default set of probes, a set of monocyte
    tDMPs from Jaffe AE and Irizarry RA, Genome Biol 2014.";
    @$probes = qw(CG00839584 CG02497428 CG02780988 CG03055440 CG05445326
    CG10045881 CG11051139 CG11058932 CG12091331 CG12962778 CG16303562
    CG16501235 CG18589858 CG18712919 CG18854666 CG21792432 CG22081096
    CG25059899 CG26989103 CG27443224);
}

# Remove redundancy in the input
my %probes_hash;
foreach my $probe (@$probes) {
    $probes_hash{$probe}++;
}

while (my ($probe, $num) = each %probes_hash) {
    if ($num > 1) {
        say "$probe is present $num times in the input. Analysing only once."
    }
}

@$probes = keys %probes_hash;

return($probes);
}

```

=head2 find_missing_probes

Arg[1] : arrayref of strings \$original_probe_ids
 Arg[2] : hashref \$existing_probe_ids (keys are probe_ids, values are ignored)
 Arg[3] : hashref \$excluded_probe_ids (keys are probe_ids, values are ignored)
 Returns : int \$num_missing_probes
 Example : my \$num_missing_probes = find_missing_probes(['cg001', 'cg002',
 'cg003', 'cg004'],
 {'cg001' => 1, 'cg003' => 1}, {'cg002' => 'cg001'});
 Description : Detects and prints the list of missing probes if any.
 Exceptions :

=cut

```

sub find_missing_probes {
    my ($original_probes, $existing_probes_hash, $excluded_probes_hash) = @_ ;
    my $num_missing_probes = 0;

    my $missing_probes = [];
    foreach my $probe_id (@$original_probes) {

```

```

        unless ($existing_probes_hash->{$probe_id} or $excluded_probes_hash-
>{$probe_id}) {
            push @$missing_probes, $probe_id;
        }
    }
    $num_missing_probes = scalar @$missing_probes;

    if ($num_missing_probes > 0) {
        warn "The following $num_missing_probes MVPs have not been analysed because
they were not found on the selected array\n";
        warn join("\n", @$missing_probes) . "\n";
    }

    return $num_missing_probes;
}
[Files]
datadir=.
#datadir=/Users/charles/Desktop/EBI_nfs_encode_work_cbreeze/directory_for_eforge
#datadir=/nfs/encode/work/cbreeze/directory_for_eforge
#datadir=/homes/cbreeze/bin/perl/eforge_directory
eFORGE was developed by Charles Breeze while on secondment at the European
Bioinformatics Institute as part of the EpiTrain Initial Training Network.

```

The code, webserver and database are currently maintained by Charles Breeze and Javier Herrero.

```
==> pasted/eForge/eForge.pm <==
```

```
package eForge::eForge;
```

```
=head1 NAME
```

```
eForge::eForge - Interface with the DB and various other common functions for eForge
```

```
=head1 VERSION
```

```
Version 0.01
```

```
=head1 LICENCE AND COPYRIGHT
```

Copyright (C) [2014-2015] EMBL - European Bioinformatics Institute and University College London

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; version 2 dated June, 1991 or at your option any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

A copy of the GNU General Public License is available in the source tree; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

=head1 CONTACT

Charles Breeze, C<< <c.breeze at ucl.ac.uk> >>

Javier Herrero, C<< <javier.herrero at ucl.ac.uk> >>

=head1 ACKNOWLEDGEMENTS

This software is based on the FORGE tool developed by Ian Dunham at the EMBL-EBI

=cut

```
use 5.010;
use strict;
use warnings FATAL => 'all';
use Storable;
```

```
my $MAX_SQL_VARIABLES = 999;
our $VERSION = '0.01';
our (@ISA, @EXPORT);
use Exporter;
@ISA = qw(Exporter);
@EXPORT = qw(get_all_datasets get_all_arrays get_all_proximity_filters process_file
get_random_matching_picks process_overlaps
get_probe_annotations_and_overlap_for_dataset get_samples_from_dataset assign
save_probe_annotation_stats proximity_filter);
```

=head1 SYNOPSIS

Provide functional modules for eForge

=head1 EXPORT

```
get_all_datasets
get_all_arrays
get_all_proximity_filters
process_file
get_random_matching_picks
process_overlaps
get_probe_annotations_and_overlap_for_dataset
get_samples_from_dataset
assign
save_probe_annotation_stats
proximity_filter
```

=head1 SUBROUTINES/METHODS

=head2 save_probe_annotation_stats

```
Arg[1]      : hashref $overlaps (see get_overlaps)
Arg[2]      : string $outdir
```

Arg[3] : string \$label
Arg[4] : string \$set_id
Returns :
Example : save_probe_annotation_stats(\$overlaps, ".", "Unnamed", "test");
Example : save_probe_annotation_stats(\$overlaps, ".", "Unnamed", 23);
Description : Save stats about the probe annotations on a text file. The \$set_id is either "test" (which relates to the input probe_ids, the ones to be tested for enrichment) or the random pick number.
The file will contain for each set the whole list of gene features and CpG islands relationships for the probes in that set.
Exceptions : Dies if file cannot be opened

=cut

```
sub save_probe_annotation_stats {
    my ($overlaps, $out_dir, $lab, $flag) = @_;
```

my \$fh;
my \$file = "\$out_dir/\$lab.overlaps.stats.txt";
open(STATS, ">>\$file") or die "cannot open \$file";
my (@gene_features, @cpg_island_relationships);
foreach my \$probeid (keys %{\$overlaps->{'MVPS'}}){
 my (\$this_gene_feature, \$this_cpg_island_relationship) =
 split("\t", \$overlaps->{'MVPS'}->{\$probeid}->{'PARAMS'});
 push @gene_features, \$this_gene_feature;
 push @cpg_island_relationships, \$this_cpg_island_relationship;
}
say STATS join("\t", \$flag, "gene_features", @gene_features);
say STATS join("\t", \$flag, "cpg_island_relationships", @cpg_island_relationships);
close(STATS);
}

=head2 process_file

Arg[1] : FILEHANDLE \$input_fh
Arg[2] : string \$format
Arg[3] : DB-connection-handle \$dbh
Arg[4] : string \$array
Arg[5] : numeric \$filter
Returns : arrayref of \$probe_ids (string)
Example : my \$probe_ids = process_file(\$fh, "probeid", \$dbh, '450k', undef);
Description : Reads the list of probe IDs or locations from the file. If the file contains locations, these will be translated into probe_ids
Exceptions :

=cut

```
sub process_file {
    my ($fh, $format, $dbh, $array, $filter) = @_;
```

my \$probe_ids = [];

if (\$format =~ /^probe/i) {
 while (<\$fh>) {

```

        next if /^#/;
        chomp;
        my $probe_id;
        if (defined $filter) {
            my $pval;
            ($probe_id, $pval) = split /\s+/, $_;
            next unless $pval >= $filter;
        } else {
            ($probe_id, undef) = split /\s+/, $_; # remove anything that is not supposed to
be there :-)
        }
        push @$probe_ids, $probe_id;
    }

} elsif ($format =~ /^bed/i) {
    if (defined $filter) {
        warn "You have specified p-value filtering, but this isn't implemented for files of
format $format. No filtering will happen."
    }
    my $locations = [];
    while (<$fh>) {
        next if /^track/;
        chomp;
        my ($chr, $from, $to) = split "\t", $_;
        next if (!defined($to));
        unless ($chr =~ /^chr/){
            $chr = "chr". $chr;
        }
        push(@$locations, [$chr, $from]);
    }
    $probe_ids = fetch_all_probe_ids($dbh, $array, $locations);

} elsif ($format =~ /^tabix/i) {
    if (defined $filter) {
        warn "You have specified p-value filtering, but this isn't implemented for files of
format $format. No filtering will happen."
    }
    my $locations = [];
    while (<$fh>) {
        chomp;
        my ($chr, $from, $to) = $_ =~ /(.)\:(\d+)\-(\d+)/;
        push(@$locations, [$chr, $from]);
    }
    $probe_ids = fetch_all_probe_ids($dbh, $array, $locations);
}
return $probe_ids;
}

```

=head2 get_random_matching_picks

```

Arg[1]    : hashref $overlaps
Arg[2]    : string $array_tag
Arg[3]    : string $data_dir

```

Arg[4] : int \$num_random_picks
Returns : arrayref of arrays of \$probe_ids (string)
Example : my \$random_picks = get_random_matching_picks(\$overlaps, "450k", ".", 1000);
Description : Get several random picks of probes matching the criteria defined in the \$overlaps hash. The random picks are selected from a pre-built hash stored in the \$data_dir called mvp_450k_bins (or so).
Exceptions : Cannot find the bins file for the selected array.

=cut

```
sub get_random_matching_picks {
    my ($overlaps, $array, $datadir, $num_random_picks) = @_;
    my $picks = [];

    # load up the stored hashes that contain the bins of mvps by feature and cpig island
    relationship.
    my %bins;
    my $bins_file = $datadir . "/mvp_${array}_bins";
    if (-e $bins_file) {
        %bins = %{ retrieve($bins_file) };
    } else {
        die "Cannot retrieve the file $bins_file\n";
    }

    foreach my $probe_id (keys %{ $overlaps->'MVPS' }) {
        my ($feature, $cpig_island_relationship) = split("\t", join("\t", $overlaps->'MVPS'-
        >{$probe_id}->'PARAMS'));

        #range has to be the number of probes to choose from in that hash subclass
        my $range = scalar @{$bins{$feature}{$cpig_island_relationship}};

        for (my $n = 0; $n < $num_random_picks; $n++) {
            my $picked_probe_id;
            while (1) {
                my $pick = int(rand($range));
                $picked_probe_id = ${$bins{$feature}{$cpig_island_relationship}}[$pick]; #pick
                the $pick'th element in the array as the chosen mvp
                last unless $picked_probe_id eq $probe_id; # must not pick the test mvp itself.
            }
            push(@{$picks->[$n]}, $picked_probe_id);
        }
    }

    return $picks;
}
```

=head2 process_overlaps

Arg[1] : arrayref of arrays \$annotated_probes (\$probe_id, \$sum, \$bit_string, \$feature, \$CGI_context)

```

Arg[2]      : arrayref of strings $cells (shortname for the cells in the dataset)
Arg[3]      : string $dataset
Returns     : hashref $stats
Example     : my $result = process_overlaps($rows, $cells, 'erc');
Description : Returns a reference to a complex hash with stats from the rows. These
are split
            into 'MVPS' and 'CELLS'. The former contains 'SUM' and 'PARAMS' for each
probe ID
            while the latter contains 'COUNT' and 'MVPS' for each cell.
Exceptions  : Dies if the number of cells does not match the length of the bit string.

=cut

sub process_overlaps {
    my ($rows, $cells, $data) = @_;
    my $overlaps;
    my @overlapping_probes_per_cell;
    my @indexes = 0..(@$cells-1);
    foreach my $row (@{$rows}){
        my ($probeid, $sum, $bit_string, $feature, $cpg_island_relationship) = @$row;
        $overlaps->{'MVPS'}->{$probeid}->{'SUM'} = $sum;
        $overlaps->{'MVPS'}->{$probeid}->{'PARAMS'} = join("\t", $feature,
$cpg_island_relationship);
        die "For $data, found ".scalar(@$cells)." cells for ".length($bit_string)." bits\n" if
(scalar(@$cells) ne length($bit_string));
        foreach my $index (@indexes) {
            ## $bit_string is a string made of 0s and 1s. If it is a 1 for this position, count and
push
            if (substr($bit_string, $index, 1)) {
                push @{$overlapping_probes_per_cell[$index]}, $probeid;
            }
        }
        my $index = 0;
        foreach my $cell (@$cells){
            if ($overlapping_probes_per_cell[$index] and
@{$overlapping_probes_per_cell[$index]}) {
                $overlaps->{'CELLS'}->{$cell}->{'COUNT'} =
scalar(@{$overlapping_probes_per_cell[$index]});
                $overlaps->{'CELLS'}->{$cell}->{'MVPS'} = $overlapping_probes_per_cell[$index];
            }
            $index++;
        }

        return $overlaps;
    }
}

=head2 get_all_datasets

Arg[1]      : DB-handle $dbh
Arg[2]      : (optional) string $species_name
Returns     : arrayref of hashes (tag/name)
Example     : my $all_datasets = get_all_dataset($dbh);

```

Description : Returns the list of all datasets in the DB. It returns an arrayref of hashes. Each

hash has two keys: 'tag' (string ID of the dataset) and 'name' (full name). You can limit the dataset to the ones available for a given species.

Exceptions :

=cut

```
sub get_all_datasets {
    my ($dbh, $species_name) = @_;
    my $datasets;

    my $sql = "SELECT dataset_tag, dataset_name FROM dataset ORDER BY dataset_id
DESC";
    my @bind_params = ();
    if ($species_name) {
        $sql .= " WHERE species_name = ?";
        push(@bind_params, $species_name);
    }
    my ($tag, $name);
    my $sth = $dbh->prepare($sql);
    $sth->execute(@bind_params);
    $sth->bind_columns(\ $tag, \ $name);
    while ($sth->fetch) {
        push(@$datasets, {tag => $tag, name => $name});
    }

    return($datasets);
}
```

=head2 get_all_arrays

Arg[1] : DB-handle \$dbh

Arg[2] : (optional) string \$species_name

Returns : arrayref of hashes (tag/name)

Example : my \$all_arrays = get_all_arrays(\$dbh);

Description : Returns the list of all arrays in the DB. It returns an arrayref of hashes.

Each

hash has two keys: 'tag' (string ID of the array) and 'name' (full name). You can limit the arrays to the ones available for a given species.

Exceptions :

=cut

```
sub get_all_arrays {
    my ($dbh, $species_name) = @_;
    my $arrays;

    my $sql = "SELECT array_tag, array_name FROM array order by array_id DESC";
    my @bind_params = ();
    if ($species_name) {
        $sql .= " WHERE species_name = ?";
        push(@bind_params, $species_name);
    }
```



```

    }
    my ($tag, $name);
    my $sth = $dbh->prepare($sql);
    $sth->execute();
    $sth->bind_columns(\ $tag, \ $name);
    while ($sth->fetch) {
        push(@$arrays, {tag => $tag, name => $name});
    }

    return($arrays);
}

```

=head2 get_all_proximity_filters

Arg[1] : DB-handle \$dbh
 Arg[2] : (optional) string \$array
 Returns : hashref of proxy-filters (string)
 Example : my \$proximity_filters = get_all_proximity_filters(\$dbh, '450k');
 Description : Returns the list of proxy filters available in the DB. It returns a hashref whose
 keys are the bkgd names (i.e. array tags) and values are the name of the proxy filter. At the moment the schema supports just one proxy filter per array.
 If you provide a \$array, the result will be limited to that array. Note that you will still get a hashref with exactly the same structure. Only the content will be limited.

Exceptions :

=cut

```

sub get_all_proximity_filters {
    my ($dbh, $array_tag) = @_;
    my $proximity_filters;

    my $sql = "SELECT array_tag, description FROM proxy_filter_info JOIN array USING
(array_id)";
    my @bind_params = ();
    if ($array_tag) {
        $sql .= " WHERE array_tag = ?";
        push(@bind_params, $array_tag);
    }
    my ($this_array_tag, $description);
    my $sth = $dbh->prepare($sql);
    $sth->execute();
    $sth->bind_columns(\ $this_array_tag, \ $description);
    while ($sth->fetch) {
        $proximity_filters->{$this_array_tag} = $description;
    }

    return($proximity_filters);
}

```

=head2 get_probe_annotations_and_overlap_for_dataset

Arg[1] : DB-handle \$dbh
 Arg[2] : string \$dataset_tag
 Arg[3] : string \$array
 Arg[4] : arrayref of strings \$probe_ids
 Returns : arrayref of arrays containing the probe_id, sum, bitstring, gene-group and
 cgi-group

Example : my \$annotated_probes =
 get_probe_annotations_and_overlap_for_dataset(\$dbh,
 'erc', '450k', \$probe_list);

Description : Fetches the gene and CGI related annotation for the set of probes as well
 as the

overlaps with the features defined in the selected dataset

Exceptions :

=cut

```
sub get_probe_annotations_and_overlap_for_dataset {
  my ($dbh, $dataset_tag, $array_tag, $probe_ids) = @_ ;
  my $results = [];

  for (my $loop = 0; $loop * $MAX_SQL_VARIABLES < @$probe_ids; $loop++) {
    my $start = $loop * $MAX_SQL_VARIABLES;
    my $end = ($loop + 1) * $MAX_SQL_VARIABLES - 1;
    $end = @$probe_ids - 1 if ($end >= @$probe_ids);

    my $sql = "SELECT probe_id, sum, bit, gene_group, cgi_group
      FROM probe_annotation
      JOIN probe_bitstring USING (array_id, probe_id)
      JOIN dataset USING (dataset_id)
      JOIN array ON (array.array_id = probe_annotation.array_id)
      WHERE array_tag = ? and dataset_tag = ?
      AND probe_id IN (?, ?" x ($end - $start)).)";

    my $sth = $dbh->prepare_cached($sql);
    $sth->execute($array_tag, $dataset_tag, @$probe_ids[$start..$end]);

    while (my $row = $sth->fetchrow_arrayref()) {
      push @$results, [$row];
    }
    $sth->finish();
  }

  return $results;
}
```

=head2 fetch_all_probe_ids

Arg[1] : DB-handle \$dbh
 Arg[2] : string \$array
 Arg[3] : arrayref of locations [\$chr, \$pos]
 Returns : arrayref of probe IDs (string)

Description : Given a list of chromosome names and location pairs, the method fetches the name

of the probe ID for that location. To do this successfully, this method requires to know the background (i.e. array) you want to translate the locations to.

Exceptions :

=cut

```
sub fetch_all_probe_ids {
    my ($dbh, $array, $locations) = @_;

    my $sth = $dbh->prepare("SELECT probe_id
                              FROM probe_mapping
                              WHERE chr_name = ? AND position = ?");
    my $probe_ids = [];

    foreach my $this_loc (@$locations) {
        my ($chr, $pos) = @$this_loc;
        $sth->execute($chr, $pos);
        my $result = $sth->fetchall_arrayref();
        my $probe_id;
        foreach my $row (@{$result}) {
            push(@$probe_ids, $$row[0]);
        }
    }
    $sth->finish();
    return $probe_ids;
}
```

=head2 proximity_filter

Filter MVPs from the MVP list if they are within 1 kb of each other. The rationale is that the first MVP to be identified in a block is chosen, and others are removed.

=cut

```
sub proximity_filter {
    my ($dbh, $array_tag, $probe_ids, $filter) = @_;
    my %prox_excluded_probes; # a hash to store MVPs found in proximity (1 kb) with an
MVP in the list
    my %filtered_probes; # The list of MVPs filtered (i.e. after filtering, the ones to keep)
    my %missing_probes;

    # Get the full list of probes as a hash (also removes redundancy)
    my %probe_id_hash;
    foreach my $probe_id (@$probe_ids){
        $probe_id_hash{$probe_id} = 1;
    }
    $probe_ids = [keys %probe_id_hash];

    for (my $loop = 0; $loop * $MAX_SQL_VARIABLES < @$probe_ids; $loop++) {
        my $start = $loop * $MAX_SQL_VARIABLES;
        my $end = ($loop + 1) * $MAX_SQL_VARIABLES - 1;
        $end = @$probe_ids - 1 if ($end >= @$probe_ids);
```

```

my $sql = "SELECT probe_id, proxy_probes FROM proxy_filter JOIN array USING
(array_id) WHERE".
    " array_tag = ? AND probe_id IN (?". ("?" x ($end - $start)).")";
my $sth = $dbh->prepare($sql); #get the blocks form the ld table
$sth->execute($array_tag, @$probe_ids[$start..$end]);
my $result = $sth->fetchall_arrayref();
$sth->finish();

foreach my $row (@{$result}){
    my ($probe_id, $probe_id_list) = @$row;
    # if the probe is in the proximity filtered set already ignore it
    next if exists $prox_excluded_probes{$probe_id};
    # if this is the first time it is seen, add it to the filtered mvps, and remove anything
in proximity with it
    $filtered_probes{$probe_id} = 1;
    next if $probe_id_list =~ /NONE/; # nothing is in proximity
    my (@other_probe_ids) = split (/\\/, $probe_id_list);
    foreach my $other_probe_id (@other_probe_ids) {
        if (exists $probe_id_hash{$other_probe_id}) {
            $prox_excluded_probes{$other_probe_id} = $probe_id; #Add to the excluded
mvps, if it is in proximity with the current mvp, and it its one of the test mvps.
        }
    }
}
}
}
}

```

```

#note that if an MVP doesn't exist in the proximity file it will be rejected regardless,
may need to add these back
return (\%prox_excluded_probes, [keys %filtered_probes]);
}

```

=head2 get_samples_from_dataset

Read the correct cell list based on data (erc, erc2, blueprint or encode). Also gets the tissue names for the cells.

=cut

```

sub get_samples_from_dataset {
    my ($dbh, $dataset_tag) = @_ ;
    my ($cells, $samples);

    my $sth = $dbh->prepare("SELECT shortcell, tissue, datatype, file, acc FROM dataset
JOIN sample USING (dataset_id) WHERE dataset_tag = ? ORDER BY sample_order");
    $sth->execute($dataset_tag);
    my ($shortcell, $tissue, $datatype, $file, $acc);
    $sth->bind_columns(\ $shortcell, \ $tissue, \ $datatype, \ $file, \ $acc);
    while ($sth->fetch) {
        $shortcell = "$shortcell|$file"; # Sometimes the same cell is used twice, with a
differnt file so need to record separately (e.g. WI-38).
        push @$cells, $shortcell;
        $$samples{$shortcell}{$tissue} = $tissue; # this is the hash that is used to connect
cells and tissues and ultimately provide the sorting order
    }
}

```

```

    $$samples{$shortcell}{'datatype'} = $datatype;
    $$samples{$shortcell}{'file'} = $file;
    $$samples{$shortcell}{'acc'} = $acc;
  }
# use Data::Dumper;
# print Dumper %$tissues;

return ($cells, $samples); # return
}

1;

==> pasted/eForge/ePlot.pm <==
package eForge::ePlot;

=head1 NAME

eForge::ePlot - Plotting utilities for eForge

=head1 VERSION

Version 0.01

=head1 LICENCE AND COPYRIGHT

Copyright (C) [2014-2015] EMBL - European Bioinformatics Institute and University
College London

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; version 2 dated June, 1991 or at your option
any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

A copy of the GNU General Public License is available in the source tree;
if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

=head1 CONTACT

Charles Breeze, C<< <c.breeze at ucl.ac.uk> >>

Javier Herrero, C<< <javier.herrero at ucl.ac.uk> >>

=head1 ACKNOWLEDGEMENTS

This software is based on the FORGE tool developed by Ian Dunham at the EMBL-EBI

=cut

```

```

use 5.010;
use strict;
use warnings FATAL => 'all';
use Sort::Naturally;

our $VERSION = '0.01';

our (@ISA, @EXPORT, @EXPORT_OK);
use Exporter;
@ISA = qw(Exporter);
@EXPORT = qw(Chart dChart table); # Symbols to export by default

=head1 SYNOPSIS

Provide plotting utilities for different plots to eForge

=head1 EXPORT

Chart
dChart
table

=head1 SUBROUTINES/METHODS

=head2 Chart

This is the original code using standard R plot to generate a static pdf.

=cut

sub Chart{
    print "Making static chart.\n";
    my ($filename, $lab, $resultsdir, $tissues, $cells, $label, $t_marginal, $t_strict, $data) =
    @_;
    my $Rdir = $resultsdir;
    my $chart = "$lab.chart.pdf";
    my $rfile = "$Rdir/$lab.chart.R";
    #set some colors
    my ($sig, $msig, $ns, $abline, $tline) = qw(red palevioletred1 steelblue3 lightpink1
    brown); #alternate msig = pink2

    open my $rfh, ">", $rfile;
    #results\Class<-cut(results\pvalue, breaks =c(min(results\pvalue), $t_marginal,
    $t_strict, max(results\pvalue)), labels=FALSE, include.lowest=TRUE) # 99 and 95% CIs
    1, 2, 3
    $t_marginal = sprintf("%.2f", $t_marginal);
    $t_strict = sprintf("%.2f", $t_strict);
    print $rfh "setwd('$Rdir')
    results<-read.table('$filename', header=TRUE,sep='\\t')

    # Class splits the data into non-significant, marginally significant and significant
    according to $t_marginal and $t_strict (in -log10 scale)

```

```

results\[extract_itex]Class <- cut(results\[extract_itex]Pvalue, breaks = c(0,[/extract_itex]t_strict, [extract_itex]t_marginal,
1)/length(unique(results[, 'Tissue'])), labels=FALSE, include.lowest=TRUE)

# Class splits the data into non-significant, marginally significant and significant
according to q-value (B-Y FDR adjusted)
results\[extract_itex]Class2 <- cut(results\[extract_itex]Qvalue, breaks = c(0, [extract_itex]t_strict, [extract_itex]t_marginal, 1),
labels=FALSE, include.lowest=TRUE)

# Re-order the entries according to tissue first and then cell type/line
tissue.cell.order <- unique(results[, c('Tissue', 'Cell')])
tissue.cell.order <- tissue.cell.order[order(tissue.cell.order[,1], tissue.cell.order[,2]), ]
# Collapse into a single string (to support same cell type in different tissues)
tissue.cell.order2 <- apply(tissue.cell.order, 1, paste, collapse = ' -- ')
results\[extract_itex]TissueCell <- apply(results[, c('Tissue', 'Cell')], 1, paste, collapse = ' -- ')
results\[extract_itex]TissueCell <- factor(results\[extract_itex]TissueCell, levels=tissue.cell.order2)

# Plot an empty chart first
pdf('[/extract_itex]chart', width=22.4, height=8)
ymax = max(-log10(results\[extract_itex]Pvalue), na.rm=TRUE)*1.1
ymin = -0.1
par(mar=c(15.5,4,3,1)+0.1)
plot(NA, ylab="", xlab="", main='MVPs in DNase1 sites (probably TF sites) in cell lines for
[/extract_itex]data [extract_itex]label',
ylim=c(ymin,ymax), las=2, pch=19, col = results\[extract_itex]Class2, xaxt='n',
xlim=c(0,length(levels(results\[extract_itex]TissueCell))), cex.main=2)

# Add horizontal guide lines for the Y-axis
abline(h=par('yaxp')[1]:par('yaxp')[2], lty=1, lwd=0.1, col='#e0e0e0')

# Add vertical lines and labels to separate the tissues
tissues <- c(0, cumsum(summary(tissue.cell.order[, 'Tissue'])))
abline(v=tissues[2:(length(tissues)-1)]+0.5, lty=6, col='[/extract_itex]tline')
text((tissues[1:(length(tissues)-1)] + tissues[2:length(tissues)]) / 2 + 0.5, ymax,
names(tissues[2:length(tissues)]), col='[/extract_itex]tline', adj=1, srt=90, cex=1.4)

# Add points (internal color first)
palette(c('[/extract_itex]sig', '[extract_itex]msig', 'white'))
points(results\[extract_itex]TissueCell, -log10(results\[extract_itex]Pvalue), pch=19, col = results\[extract_itex]Class2,
xaxt='n')

# Add contour to the points
palette(c('black', '[/extract_itex]msig', '[extract_itex]ns'))
points(results\[extract_itex]TissueCell, -log10(results\[extract_itex]Pvalue), pch=1, col = results\[extract_itex]Class2,
xaxt='n')

# Add X-axis (use cell name only and not TissueCell)
axis(1, seq(1,length(tissue.cell.order[,2])), labels=tissue.cell.order[,2], las=2,
cex.axis=0.9)
mtext(1, text='Cell', line=14, cex=1.4)
mtext(2, text='-log10 binomial p-value', line=2, cex=1.4)

# Add legend (internal color first)
palette(c('white', '[/extract_itex]msig', '[extract_itex]sig'))

```

```
legend('topleft', pch=19, legend=c('q < $t_strict', 'q < $t_marginal', 'non-sig'), col = 3:1,
cex=0.8, inset=c(0.001, 0.005), box.col='white', title='FDR q-value', text.col='white',
bg='white')
```

```
# Add contour to the points in the legend
palette(c('$ns', '$msig', 'black'))
legend('topleft', pch=1, legend=c('q < $t_strict', 'q < $t_marginal', 'non-sig'), col = 3:1,
cex=0.8, inset=c(0.001, 0.005), box.col='darkgrey', title='FDR q-value')
```

```
palette('default')
dev.off()
";
```

```
#run the R code
system("R", "--no-save", "--quiet", "--slave", "--file=$rfile");
}
```

```
=head2 dChart
```

Make dimple interactive chart.

```
=cut
```

```
sub dChart{
  my ($filename, $lab, $resultsdir, $data, $label, $t_marginal, $t_strict, $web) = @_;

  print "Making dChart.\n";
  my $chart = "$lab.dchart.html";
  my $Rdir = $resultsdir;
  my $rfile = "$Rdir/$lab.dChart.R";
  open my $rcfh, ">", $rfile;
  print $rcfh "setwd(\"$Rdir\")
results<-read.table(\"$filename\", header = TRUE, sep=\"\\t\\t\")
```

```
# Class splits the data into non-significant, marginally significant and significant
according to $t_marginal and $t_strict (in -log10 scale)
results$Class <- cut(results$Pvalue, breaks = c(0, $t_strict, $t_marginal,
1)/length(unique(results[, 'Tissue'])), labels=FALSE, include.lowest=TRUE)
```

```
# Class splits the data into non-significant, marginally significant and significant
according to q-value (B-Y FDR adjusted)
results$Class2 <- cut(results$Qvalue, breaks = c(0, $t_strict, $t_marginal, 1),
labels=FALSE, include.lowest=TRUE)
color.axis.palette = c();
if (length(which(results$Class2 == 1)) > 0 ) {
  color.axis.palette = c('red');
}
if (length(which(results$Class2 == 2)) > 0 ) {
  color.axis.palette = c(color.axis.palette, '#FF82ab');
}
color.axis.palette = c(color.axis.palette, 'lightblue');
if (length(color.axis.palette) < 2) {
```



```

    color.axis.palette = c(color.axis.palette, 'lightblue'); # Add it twice to force the color if
    only non-significant values
  }

results$log10pvalue <- -log10(results$Pvalue)

# Re-order the entries according to tissue first and then cell type/line
tissue.cell.order <- unique(results[, c('Tissue', 'Cell')])
tissue.cell.order <- tissue.cell.order[order(tissue.cell.order[,1], tissue.cell.order[,2]), ]
# Collapse into a single string (to support same cell type in different tissues)
tissue.cell.order2 <- apply(tissue.cell.order, 1, paste, collapse = ' -- ')
results$TissueCell <- apply(results[, c('Tissue', 'Cell')], 1, paste, collapse = ' -- ')
results$TissueCell <- factor(results$TissueCell, levels=tissue.cell.order2)

# Count number of cell types for each tissue (to be able to draw the vertical separation
lines afterwards
tissues <- c(0, cumsum(summary(tissue.cell.order[, 'Tissue'])))

require(rCharts)

dplot.height=900
dplot.width=2000
bounds.x=60
bounds.y=50
bounds.height=dplot.height - 300
bounds.width=dplot.width - bounds.x - 20

# Create a dimple plot, showing p-value vs cell, split data by tissue, cell, probe, etc to see
individual points instead of aggregate avg
d1 <- dPlot(
  y = 'log10pvalue',
  x = c('TissueCell'),
  groups = c('TissueCell', 'Accession', 'Pvalue', 'Qvalue', 'Datatype', 'Probe'),
  data = results,
  type = 'bubble',
  width = dplot.width,
  height = dplot.height,
  bounds = list(x=bounds.x, y=bounds.y, height=bounds.height, width=bounds.width),
  id = 'chart.$lab'
)

# Force the order on the X-axis
d1$xAxes( type = 'addCategoryAxis', grouporderRule = 'Cell', orderRule =
tissue.cell.order[,2])
d1$xAxes( type = 'addCategoryAxis', grouporderRule = 'TissueCell', orderRule =
as.factor(tissue.cell.order2))

d1$yAxis( type = 'addMeasureAxis' )

# Color points according to the q-value
d1$colorAxis(
  type = 'addColorAxis',
  colorSeries = 'Class2',
  palette = color.axis.palette)

```

```

# Builds a JS string to add labels for tissues
labels.string = paste(paste0(\"
// Adds labels for tissues
myChart.svg.insert('text', 'g')
.attr('x', 0)
.attr('y', 0)
.attr('font-size', 20)
.attr('font-family', 'Arial')
.style('fill', 'brown')
.attr('transform', 'translate(\"\", (bounds.x - 5 + bounds.width *
(tissues[1:(length(tissues)-1)] + tissues[2:length(tissues)]) / (2 * max(tissues))), \"\", 60)
rotate(90)')
.attr('text-anchor', 'top')
.text(\"\", names(tissues[2:length(tissues)]), \"\")
\"), collapse=)

# Builds a JS string to add vertical lines to separate tissues
lines.string = paste(paste0(\"
// Adds vertical lines between tissues
myChart.svg.append('line')
.attr('x1', \"\", (bounds.x + bounds.width * tissues[2:(length(tissues)-1)]/
max(tissues)), \"\")
.attr('y1', 50)
.attr('x2', \"\", (bounds.x + bounds.width * tissues[2:(length(tissues)-1)]/
max(tissues)), \"\")
.attr('y2', \"\", (50 + bounds.height), \"\")
.style('stroke', 'brown')
.style('stroke-dasharray', '10,3,3,3')
\"), collapse=)

# Adds some JS to be run after building the plot to get the image we want
d1\\$setTemplate(afterScript = paste0(\"
<script>
myChart.draw()

// Substitutes TissueCell labels in X-axis by Cell labels
myChart.axes[1].shapes
.selectAll('text')
.text(function (d) {
  var i;
  for (i = 0; i < data.length; i += 1) {
    if (data[i].TissueCell === d) {
      return data[i].Cell;
    }
  }
});

// Adds title for X-axis
myChart.axes[1].titleShape
.style('font-size', 20)

// Adds title for Y-axis
myChart.axes[2].titleShape

```

```

        .style('font-size', 20)
        .text('-log10 binomial p-value')

// Adds main title
myChart.svg.append('text')
    .attr('x', \", (dplot.width / 2), \")
    .attr('y', \", (bounds.y / 2), \")
    .attr('font-size', 24)
    .attr('font-family', 'Arial')
    .attr('font-weight', 'bold')
    .style('fill', 'black')
    .attr('text-anchor', 'middle')
    .text('MVPs in DNase1 sites (probably TF sites) in cell lines for $data $label')
    \", labels.string, \")
    \", lines.string, \")
// Adds vertical line at the far right of the plot
myChart.svg.append('line')
    .attr('x1', \", (bounds.x + bounds.width), \")
    .attr('y1', \", bounds.y, \")
    .attr('x2', \", (bounds.x + bounds.width), \")
    .attr('y2', \", (bounds.y + bounds.height), \")
    .style('stroke', 'rgb(0,0,0)')
</script>
\"))

```

```

d1\${save('$chart', cdn = F)}\n";

```

```

system("R", "--no-save", "--quiet", "--slave", "--file=$rfile");

```

```

if ($web) {
    $web =~ s/\$//;
    open(FILE, "$resultsdir/$chart") or die;
    my @lines = <FILE>;
    close(FILE);
    open(FILE, ">", "$resultsdir/$chart") or die;
    foreach my $line (@lines) {
        $line =~ s/src='.*\$/js/src='$web\libraries\dimple\js/;
        print FILE $line;
    }
    close(FILE);
}
}

```

```

=head2 table

```

```

=cut

```

```

sub table{
    my ($filename, $lab, $resultsdir, $web) = @_ ;

    # Make Datatables table

```

```

print "Making Table.\n";
my $chart = "$lab.table.html";
my $Rdir = $resultsdir;
my $rfile = "$Rdir/$lab.table.R";
open my $rcfh, ">", $rfile;
print $rcfh "setwd('$Rdir')
results <- read.table('$filename', header = TRUE, sep='\\t')
results <- subset(results, T, select = c('Cell', 'Tissue', 'Datatype', 'Accession', 'Pvalue',
'Qvalue', 'Probe'))
require(rCharts)
dt <- dTable(
  results,
  sScrollY = '600',
  bPaginate = F,
  sScrollX = '100%',
  width = '680px'
)
dt\\$save('$chart', cdn = F)\n";

system("R", "--no-save", "--quiet", "--slave", "--file=$rfile");

if ($web) {
  $web =~ s/\\/$//;
  open(FILE, "$resultsdir/$chart") or die;
  my @lines = <FILE>;
  close(FILE);
  open(FILE, ">", "$resultsdir/$chart") or die;
  foreach my $line (@lines) {
    $line =~ s/href='.*\\css/href='$web\\libraries\\/datatables\\css/;
    $line =~ s/src='.*\\js/src='$web\\libraries\\/datatables\\js/;
    print FILE $line;
  }
  close(FILE);
}
}

1;

==> pasted/eForge/eStats.pm <==
package eForge::eStats;

=head1 NAME

eForge::eStats - Stats for use in eForge

=head1 VERSION

Version 0.01

=head1 LICENCE AND COPYRIGHT

```

Copyright (C) [2014-2015] EMBL - European Bioinformatics Institute and University College London

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; version 2 dated June, 1991 or at your option any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

A copy of the GNU General Public License is available in the source tree; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

=head1 CONTACT

Charles Breeze, C<< <c.breeze at ucl.ac.uk> >>

Javier Herrero, C<< <javier.herrero at ucl.ac.uk> >>

=head1 ACKNOWLEDGEMENTS

This software is based on the FORGE tool developed by Ian Dunham at the EMBL-EBI

=cut

```
use 5.010;
use strict;
use warnings FATAL => 'all';
```

```
our $VERSION = '0.01';
```

```
our (@ISA, @EXPORT);
use Exporter;
@ISA = qw(Exporter);
@EXPORT = qw(mean variance std zscore log10 binomial factorial fdr); # Symbols to
export by default
```

=head1 SYNOPSIS

Provide various stats for eForge to do its stuff

=head1 EXPORT

```
mean
variance
std
log10
binomial
factorial
```

fdr

=head1 SUBROUTINES/METHODS

=head2 mean

Calculates the biased mean of an array

pass it a float array and it will return the mean
reused from Ben Brown

=cut

```
sub mean {
    my $sum = 0;
    foreach (@_){
        $sum+= $_;
    }
    return $sum/($#_+1);
}
```

=head2 variance

Calculates the biased variance of an array

Pass it a float array and it will return the variance

Reused from Ben Brown

=cut

```
sub variance {
    my $ev = mean(@_);
    my $sum = 0;
    foreach (@_) { $sum += ($_ - $ev)**2 };

    return $sum/($#_+1);
}
```

=head2 std

Calculates the standard deviation of an array: this is just the sqrt of the var

=cut

```
sub std { sqrt(variance(@_)) }
```

=head2 log10

log 10 since perl doesn't have

=cut

```
sub log10 {
  my $n = shift;
  return log($n)/log(10);
}
```

=head2 zscore

Calculates the z-score for a given result and an array of values

=cut

```
sub zscore {
  my ($teststat, $values) = @_;
  my $zscore;

  my $mean = mean(@$values);
  my $sd = std(@$values);
  if ($sd == 0) {
    $zscore = "NA";
  } else {
    $zscore = sprintf("%.3f", ($teststat-$mean)/$sd);
  }

  return $zscore;
}
```

=head2 binomial

Exact solution of binomial probability for k picks out of n, for n or greater need to sum for each k up to n

=cut

```
sub binomial {

  my ($k, $n, $p) = @_;

  my $prob = exp($k*log($p) + ($n-$k)*log(1 - $p) + log_factorial($n) - log_factorial($k)
- log_factorial($n - $k));

  return $prob;
}
```

=head2 log_factorial

Calculate log(N!). Required for binomial.

Uses a cache to speed up the calculation.

=cut

my %_log_factorial_cache;

```
sub log_factorial{
```

```

my ($n) = shift;
return 0 if($n <=1 ); # log(1) = 0
return $_log_factorial_cache{$n} if (exists($_log_factorial_cache{$n}));
my $result = 0; # log(1) = 0;
for (my $i = $n; $i > 1; $i--) {
    $result += log($i);
}
$_log_factorial_cache{$n} = $result;
return $_log_factorial_cache{$n};
}

```

=head2 fdr

Empirical false discovery rate = FP/TP+FP.

Need to modify this now have switched to binomial p values.

=cut

```

sub fdr{
    my ($tp, $mvps, $cells) = @_ ;
    if ($tp == 0){
        return "NA";
    }
    else{
        my $fpr = 0.0085 * exp(-0.04201 * $mvps) + 0.00187; # from simulations of random
data (Forge, not eForge->) 0.0085*exp(-0.04201. SNPs) + 0.00187
        my $fdr = ($cells * $fpr) / $tp;
        return $fdr;
    }
}

```

1;

==> pasted/database/README.txt <==

This directory contains the scripts and files necessary to rebuild the eFORGE database from scratch.

In summary, the steps are:

- Create an empty DB
- Load the information about the arrays
- Load the dataset with DHS, Histone peaks, etc
- Move the DB to its final destination

1. CREATE AN EMPTY DATABASE

Please refer to the help of init_db.pl for information about the different options

```

rm eforge_1.2.db
perl init_db.pl --db_name eforge_1.2.db

```

2. LOAD THE ARRAYS

Please refer to the help of load_450k_array.pl for information about the different options


```
perl load_450k_array.pl --work_dir input --db_name eforge_1.2.db
perl load_27k_array.pl --work_dir input --db_name eforge_1.2.db
```

3. LOAD THE DATASETS

Please refer to the help of load_dataset.pl for information about the different options

```
perl load_dataset.pl --db_name eforge_1.2.db --tag erc --name 'Roadmap Epigenomics
(2012 data) - DHS' --decode_file erc.decode --work_dir input/erc/
perl load_dataset.pl --db_name eforge_1.2.db --tag encode --name 'ENCODE - DHS' --
decode_file encode.decode --work_dir input/encode/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-DHS --name 'Consolidated
Roadmap Epigenomics - DHS' --decode_file erc2.decode --work_dir input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-H3K27me3 --name
'Consolidated Roadmap Epigenomics - H3K27me3' --decode_file erc2-
H3K27me3.decode --work_dir input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-H3K36me3 --name
'Consolidated Roadmap Epigenomics - H3K36me3' --decode_file erc2-
H3K36me3.decode --work_dir input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-H3K4me3 --name 'Consolidated
Roadmap Epigenomics - H3K4me3' --decode_file erc2-H3K4me3.decode --work_dir
input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-H3K9me3 --name 'Consolidated
Roadmap Epigenomics - H3K9me3' --decode_file erc2-H3K9me3.decode --work_dir
input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-H3K4me1 --name 'Consolidated
Roadmap Epigenomics - H3K4me1' --decode_file erc2-H3K4me1.decode --work_dir
input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-H3-all --name 'Consolidated
Roadmap Epigenomics - All H3 marks' --decode_file erc2-H3-all.decode --work_dir
input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag blueprint --name 'Blueprint - DHS' --
decode_file blueprint.decode --work_dir input/blueprint/
```

4. MOVE THE DATABASE TO ITS FINAL LOCATION

```
mv eforge_1.2.db ..

==> pasted/database/init_db.pl <==
#!/usr/bin/env perl
use strict;
use warnings;

use DBI;
use Getopt::Long;

my $db_dir = ".";
my $db_name = "eforge_1.2.db";

my $help;

my $desc = qq{init_db.pl [--db_name $db_name] [--db_dir $db_dir]}
```

```

where:
--db_name is the name of the SQLite file [def: $db_name]
--db_dir is the location of the SQLite file [def: $db_dir]

};

GetOptions(
    "help" => \$help,
    "db_name=s" => \$db_name,
    "db_dir=s" => \$db_dir,
);

if ($help) {
    print $desc;
    exit(0);
}

my $dsn = "dbi:SQLite:dbname=$db_dir/$db_name";
my $dbh = DBI->connect($dsn, "", "") or die $DBI::errstr;

$dbh->do("CREATE TABLE IF NOT EXISTS assembly (
    assembly_id INTEGER PRIMARY KEY AUTOINCREMENT,
    species_name,
    assembly_name,
    UNIQUE (species_name, assembly_name))");

$dbh->do("CREATE TABLE IF NOT EXISTS array (
    array_id INTEGER PRIMARY KEY AUTOINCREMENT,
    array_tag UNIQUE,
    array_name UNIQUE,
    species_name)");
$dbh->do("CREATE TABLE IF NOT EXISTS probe_mapping_info (
    probe_mapping_id INTEGER PRIMARY KEY AUTOINCREMENT,
    array_id INTEGER NOT NULL REFERENCES array(array_id),
    assembly_id INTEGER NOT NULL,
    url,
    UNIQUE (array_id, assembly_id))");
$dbh->do("CREATE TABLE IF NOT EXISTS probe_mapping (
    probe_mapping_id INTEGER NOT NULL REFERENCES
probe_mapping_info(probe_mapping_id),
    probe_id,
    chr_name,
    position INTEGER,
    UNIQUE (probe_mapping_id, probe_id))");
$dbh->do("CREATE INDEX probe_mapping_idx1 on probe_mapping (position,
chr_name)");

$dbh->do("CREATE TABLE IF NOT EXISTS proxy_filter_info (
    array_id INTEGER NOT NULL REFERENCES array(array_id),
    description,

```

```

    UNIQUE(array_id));
$dbh->do("CREATE TABLE IF NOT EXISTS proxy_filter (
    array_id INTEGER NOT NULL REFERENCES array(array_id),
    probe_id,
    proxy_probes,
    UNIQUE (array_id, probe_id));

$dbh->do("CREATE TABLE IF NOT EXISTS probe_annotation_info (
    array_id INTEGER NOT NULL REFERENCES array(array_id),
    gene_reference_name NOT NULL,
    cgi_reference_name NOT NULL,
    url,
    UNIQUE(array_id));
$dbh->do("CREATE TABLE IF NOT EXISTS probe_annotation (
    array_id INTEGER NOT NULL REFERENCES probe_annotation_info(array_id),
    probe_id NOT NULL,
    gene_group,
    cgi_group,
    UNIQUE (array_id, probe_id));

$dbh->do("CREATE TABLE IF NOT EXISTS dataset (
    dataset_id INTEGER PRIMARY KEY AUTOINCREMENT,
    dataset_tag UNIQUE,
    dataset_name UNIQUE,
    species_name);
$dbh->do("CREATE TABLE IF NOT EXISTS sample (
    dataset_id INTEGER REFERENCES dataset(dataset_id),
    sample_order INTEGER,
    file,
    lab,
    datatype,
    cell,
    tissue,
    shortcell,
    individual,
    acc,
    url);
$dbh->do("CREATE TABLE IF NOT EXISTS probe_bitstring (
    array_id INTEGER NOT NULL REFERENCES array(array_id),
    probe_id INTEGER NOT NULL,
    dataset_id INTEGER NOT NULL REFERENCES dataset(dataset_id),
    sum INTEGER,
    bit,
    UNIQUE(array_id,probe_id,dataset_id));

==> pasted/database/load_27k_array.pl <==
#!/usr/bin/env perl
use strict;
use warnings;

use Getopt::Long;
use File::Spec qw(splitpath);

```

```

use DBI;

my $db_dir = ".";
my $db_name = "eforge_1.2.db";
my $array_tag = "27k";
my $array_name = "Illumina Human 27k array (subset of 450k only)";
my $species = "Homo sapiens";
my $proxy_threshold = 1000;
my $illumina27k_bpm_file = 'ftp://webdata:webdata@ussd-
ftp.illumina.com/downloads/ProductFiles/HumanMethylation27/humanmethylation27
_270596_v1-2.bpm';
my $illumina450k_csv_file = 'ftp://webdata2:webdata2@ussd-
ftp.illumina.com/downloads/ProductFiles/HumanMethylation450/HumanMethylation
450_15017482_v1-2.csv';
my $work_dir = ".";
my $bedtools = "bedtools";

my $help;

my $desc = qq{load_27k_array.pl [options]

DESCRIPTION:

This script loads the subset of probe locations and annotations of the Illumina Human
27k
methylation array that are shared with the 450k array into the eFORGE database.

Note that you *must* load the arrays before loading the datasets. If you want to include
new arrays
at a later date, you will have to reload all the datasets again (i.e. you will have to re-start
from
scratch).

Optional arguments:
--db_name <name>
    is the name of the SQLite file [def: $db_name]
--db_dir <path>
    is the location of the SQLite file [def: $db_dir]
--work_dir <path>
    is the location where temporary files will be downloaded/created [def: $work_dir]
--bedtools <name>
    is the name of the bedtools executable you want to use [def: $bedtools]
};

GetOptions(
    "help" => \$help,
    "db_name=s" => \$db_name,
    "db_dir=s" => \$db_dir,
    "work_dir=s" => \$work_dir,
    "bedtools=s" => \$bedtools,
);

my $dsn = "dbi:SQLite:dbname=$db_dir/$db_name";
my $dbh = DBI->connect($dsn, "", "") or die $DBI::errstr;

```

```

my ($vol, $path, $file) = File::Spec->splitpath($illumina450k_csv_file);

if (!-e "$work_dir/$file") {
    download_url($illumina450k_csv_file, $work_dir);
}

my $array450k_info = parse_450k_file("$work_dir/$file");

my ($vol27k, $path27k, $file27k) = File::Spec->splitpath($illumina27k_bpm_file);

if (!-e "$work_dir/$file27k") {
    download_url($illumina27k_bpm_file, $work_dir);
}

my $array27k_info = parse_27k_file("$work_dir/$file27k");

my $array_info = subset_array($array450k_info, $array27k_info);

load_array($dbh, $db_name, $species, $array_tag, $array_name, $array_info);

load_proxy_filter($dbh, $db_name, $array_tag, "GRCh37", $work_dir, $proxy_threshold);

exit();

sub download_url {
    my ($url, $work_dir) = @_ ;

    print "Downloading $url...\n";
    system("wget -q -N -P $work_dir $url");
}

sub parse_450k_file {
    my ($illumina450k_csv_file) = @_ ;
    my $array;

    open(CSV, $illumina450k_csv_file) or die;
    my $annotation;
    my @gene_annotations = ("TSS200", "TSS1500", "1stExon", "Body", "3'UTR", "5'UTR",
"IGR");
    my $cpg_annotations = {
        'N_Shelf' => "Shelf_Shore",
        'S_Shelf' => "Shelf_Shore",
        'N_Shore' => "Shelf_Shore",
        'S_Shore' => "Shelf_Shore",
        'Island' => "Island"};
    while (<CSV>) {
        chomp;
        my @data = split(",", $_);
        my $probe_id = $data[0];
        next if ($probe_id !~ /^cg/ and $probe_id !~ /^ch\./);
        my $probe_chr37 = $data[11];

```

```

my $probe_loc37 = $data[12];
my $probe_chr36 = $data[14];
my $probe_loc36 = $data[15];
my $this_gene_annotation_arrayStr = $data[23];
my $this_cpg_annotation = $data[25]?$cpg_annotations->{$data[25]}:"NA";

my $this_gene_annotation;
if (!defined($this_gene_annotation_arrayStr) or $this_gene_annotation_arrayStr eq
"") {
    $this_gene_annotation = "IGR";
} else {
    my @this_gene_annotations = split(";", $this_gene_annotation_arrayStr);
    foreach my $this_a (@this_gene_annotations) {
        if (grep {$_ eq $this_a} @gene_annotations) {
            $this_gene_annotation = $this_a;
            last;
        }
    }
}

$annotation->{$this_gene_annotation."-".$this_cpg_annotation}++;

$array->{$probe_id} = [$probe_chr36, $probe_loc36, $probe_chr37, $probe_loc37,
$this_gene_annotation, $this_cpg_annotation];
}
close(CSV);
#   foreach my $this_a (keys %$annotation) { #@gene_annotations) {
#       print $annotation->{$this_a}, "\t", $this_a, "\n";
#   }

return $array;
}

sub parse_27k_file {
    my ($illumina27k_bpm_file) = @_ ;
    my $array;

    open(CSV, $illumina27k_bpm_file) or die;
    my $block = "";
    while (<CSV>) {
        if (/^\s*[(\w+)\s]\s/) {
            $block = $1;
        }
    }
    next if ($block ne "Assay");

    chomp;
    my @data = split(" ", $_);
    my $probe_id = $data[0];
    my $name = $data[1];
    my $probe_chr36 = $data[8];
    my $probe_loc36 = $data[9];

    if ($probe_id !~ /^cg/ and $probe_id !~ /^ch\./) {

```

```
#      print "Skipping $probe_id\n";
      next;
    }
    if ($probe_id ne $name) {
      print "ERROR: $probe_id ne $name\n";
    }
    $array->{$probe_id} = [$probe_chr36, $probe_loc36];
  }
  close(CSV);

  return $array;
}

sub subset_array {
  my ($full_array, $sub_array) = @_;
  my $array;
  foreach my $probe_id (keys %$full_array) {
    if ($sub_array->{$probe_id}) {
      $array->{$probe_id} = $full_array->{$probe_id};
      print "MISMATCH $probe_id\n" if (($full_array->{$probe_id}->[0] ne $sub_array-
>{$probe_id}->[0])
      or
      ($full_array->{$probe_id}->[1] != $sub_array->{$probe_id}-
>[1]));
    }
  }
  return $array;
}

sub load_array_1 {
  my ($dbh, $code_name, $array_info) = @_;

  my $table_name = $code_name;
  $table_name =~ s/\W//g;
  $table_name = "array_$table_name";
  $dbh->do("CREATE TABLE IF NOT EXISTS $table_name (probe_id, location,
gene_group, cgi_group)");

  my $sth = $dbh->prepare("INSERT INTO array_$code_name VALUES (?, ?, ?, ?)");
  foreach my $this_probe_id (sort keys $array_info) {
    my ($chr, $loc, $gene_group, $cgi_group) = @{$array_info->{$this_probe_id}};
    my $location = "$chr:$loc-$loc";
    $sth->execute($this_probe_id, $location, $gene_group, $cgi_group);
  }
  $sth->finish();

  return $table_name;
}

sub load_array {
  my ($dbh, $db_name, $species_name, $array_tag, $array_name, $array_info) = @_;
```

```

my $sth;
$sth = $dbh->prepare("INSERT OR IGNORE INTO array (array_tag, array_name,
species_name) VALUES (?, ?, ?)");
$sth->execute($array_tag, $array_name, $species_name);
my $array_id = $dbh->last_insert_id("", "", "", "");
$sth->finish();
if ($array_id == 0) {
    $array_id = $dbh->selectrow_array("SELECT array_id FROM array WHERE
array_name = '$array_name' AND species_name = 'Homo sapiens'");
}

$sth = $dbh->prepare("INSERT OR IGNORE INTO assembly (species_name,
assembly_name) VALUES (?, ?)");
$sth->execute($species_name, "GRCh37");
$sth->execute($species_name, "NCBI36");
$sth->finish();
my $human36_assembly_id = $dbh->selectrow_array("SELECT assembly_id FROM
assembly WHERE species_name = 'Homo sapiens' AND assembly_name = 'NCBI36'");
my $human37_assembly_id = $dbh->selectrow_array("SELECT assembly_id FROM
assembly WHERE species_name = 'Homo sapiens' AND assembly_name = 'GRCh37'");

$sth = $dbh->prepare("INSERT OR IGNORE INTO probe_mapping_info (array_id,
assembly_id, url) VALUES (?, ?, ?)");
$sth->execute($array_id, $human36_assembly_id, "");
my $probe_mapping_human36_id = $dbh->last_insert_id("", "", "", "");
$sth->execute($array_id, $human37_assembly_id, "");
my $probe_mapping_human37_id = $dbh->last_insert_id("", "", "", "");
$sth->finish();

$sth = $dbh->prepare("INSERT OR IGNORE INTO probe_annotation_info (array_id,
gene_reference_name, cgi_reference_name, url) VALUES (?, ?, ?, ?)");
$sth->execute($array_id, "RefSeq", "UCSC CpG Islands", "");
# my $probe_annotation_id = $dbh->last_insert_id("", "", "", "");
$sth->finish();

open(CSV1, ">probe_mapping.csv") or die;
open(CSV2, ">probe_annotation.csv") or die;
# my $sth1 = $dbh->prepare("INSERT OR IGNORE INTO probe_mapping
(probe_mapping_id, probe_id, chr_name, position) VALUES (?, ?, ?, ?)");
# my $sth2 = $dbh->prepare("INSERT OR IGNORE INTO probe_annotation
(probe_annotation_id, probe_id, gene_group, cgi_group) VALUES (?, ?, ?, ?)");
foreach my $this_probe_id (sort keys $array_info) {
    my ($chr36, $loc36, $chr37, $loc37, $gene_group, $cgi_group) = @{$array_info-
>{$this_probe_id}};
    print CSV1 join(",", $probe_mapping_human36_id, $this_probe_id, "chr$chr36",
$loc36), "\n";
    print CSV1 join(",", $probe_mapping_human37_id, $this_probe_id, "chr$chr37",
$loc37), "\n";
    # print CSV2 join(",", $probe_annotation_id, $array_id, $this_probe_id, $gene_group,
$cgi_group), "\n";
    print CSV2 join(",", $array_id, $this_probe_id, $gene_group, $cgi_group), "\n";
    # $sth1->execute($probe_mapping_human36_id, $this_probe_id, "chr$chr36",
$loc36) if ($probe_mapping_human36_id);

```



```
# $sth1->execute($probe_mapping_human37_id, $this_probe_id, "chr$chr37",
$loc37) if ($probe_mapping_human37_id);
# $sth2->execute($probe_annotation_id, $this_probe_id, $gene_group, $cgi_group) if
($probe_annotation_id);
}
close(CSV1);
close(CSV2);
system("echo '.mode csv
.import probe_mapping.csv probe_mapping
.import probe_annotation.csv probe_annotation' | sqlite3 $db_name");
# $sth1->finish();
# $sth2->finish();

}

sub load_proxy_filter {
    my ($dbh, $db_name, $array_tag, $assembly_name, $work_dir, $distance_threshold) =
@_;

    my $array_id = $dbh->selectrow_arrayref("SELECT array_id FROM array WHERE
array_tag = '$array_tag'")->[0];
    my $bed_file = dump_array_bed_file($dbh, $array_id, $assembly_name, $work_dir);
    my $this_output_bed_file = "$work_dir/array_{$array_id}.proxy.bed";
    my $runstr = "$bedtools window -w $distance_threshold -a $bed_file -b $bed_file >
$this_output_bed_file";
    system($runstr) == 0 or die "Error while running bedtools: $?";

    my $sth = $dbh->prepare("INSERT OR IGNORE INTO proxy_filter_info (array_id,
description) VALUES (?, ?)");
    $sth->execute($array_id, "1kb");

    open(BED, $this_output_bed_file) or die;
    my $all_probes;
    my $mapping_probes;
    while (<BED>) {
        chomp;
        my ($chr1, $start1, $end1, $probe1, $chr2, $start2, $end2, $probe2) = split("\t", $_);
        $all_probes->{$probe1} = 1;
        if ($probe1 ne $probe2) {
            $mapping_probes->{$probe1}->{$probe2} = 1;
        }
    }
    close(BED);
    open(CSV, ">proxy_filter.csv") or die;
    foreach my $probe (sort keys %$all_probes) {
        if (defined($mapping_probes->{$probe})) {
            print CSV "$array_id,$probe,", join("|", sort keys %{$mapping_probes->{$probe}}), "\n";
        } else {
            print CSV "$array_id,$probe,NONE\n";
        }
    }
    close(CSV);
    system("echo '.mode csv
```

```
.import proxy_filter.csv proxy_filter' | sqlite3 $db_name");

}

sub dump_array_bed_file {
    my ($dbh, $this_array_id, $assembly_name, $work_dir) = @_;
    my $sql = "SELECT probe_mapping_id
        FROM probe_mapping_info
        JOIN assembly USING (assembly_id)
        WHERE array_id = $this_array_id
        AND assembly_name = '$assembly_name'";
    my $probe_mapping_id = $dbh->selectrow_array($sql);
    if (!$probe_mapping_id) {
        die "Cannot find a mapping for array $this_array_id and assembly
$assembly_name\n";
    }

    $sql = "SELECT probe_id, chr_name, position FROM probe_mapping WHERE
probe_mapping_id = $probe_mapping_id";
    my $probes = $dbh->selectall_arrayref($sql);

    my $bed_file = "$work_dir/array_${this_array_id}.bed";
    open(BED, ">$bed_file") or die;
    foreach my $this_probe (sort_sort_probes @$probes) {
        print BED join("\t", $this_probe->[1], $this_probe->[2], $this_probe->[2]+1,
$this_probe->[0]), "\n";
    }
    close(BED);

    return $bed_file;
}

sub _sort_probes {
    my $chr_a = $a->[1];
    my $chr_b = $b->[1];
    my $loc_a = $a->[2];
    my $loc_b = $b->[2];
    $chr_a =~ s/chr//;
    $chr_b =~ s/chr//;
    if ($chr_a eq $chr_b) {
        return $loc_a <=> $loc_b;
    } elsif ($chr_a =~ /^d/ and $chr_b =~ /^d/) {
        return $chr_a <=> $chr_b;
    } elsif ($chr_a =~ /^d/) {
        return -1;
    } elsif ($chr_b =~ /^d/) {
        return 1;
    } else {
        return $chr_a cmp $chr_b;
    }
}

exit();
```

```

==> pasted/database/load_450k_array.pl <==
#!/usr/bin/env perl
use strict;
use warnings;

use Getopt::Long;
use File::Spec qw(splitpath);
use DBI;

my $db_dir = ".";
my $db_name = "eforge_1.2.db";
my $array_tag = "450k";
my $array_name = "Illumina Human 450k array";
my $species = "Homo sapiens";
my $proxy_threshold = 1000;
my $illumina450k_csv_file = 'ftp://webdata2.webdata2@ussd-
ftp.illumina.com/downloads/ProductFiles/HumanMethylation450/HumanMethylation
450_15017482_v1-2.csv';
my $work_dir = ".";
my $bedtools = "bedtools";

my $help;

my $desc = qq{load_450k_array.pl [options]

DESCRIPTION:

This script loads the probe locations and annotations of the Illumina Human 450k
methylation array
into the eFORGE database.

Note that you must load the arrays before loading the datasets. If you want to include
new arrays
at a later date, you will have to reload all the datasets again (i.e. you will have to re-start
from
scratch).

Optional arguments:
--db_name <name>
    is the name of the SQLite file [def: $db_name]
--db_dir <path>
    is the location of the SQLite file [def: $db_dir]
--work_dir <path>
    is the location where temporary files will be downloaded/created [def: $work_dir]
--bedtools <name>
    is the name of the bedtools executable you want to use [def: $bedtools]
};

GetOptions(
    "help" => \$help,
    "db_name=s" => \$db_name,
    "db_dir=s" => \$db_dir,
    "work_dir=s" => \$work_dir,

```

```

"bedtools=s" => \ $bedtools,
);

my $dsn = "dbi:SQLite:dbname=$db_dir/$db_name";
my $dbh = DBI->connect($dsn, "", "") or die $DBI::errstr;

my ($vol, $apth, $file) = File::Spec->splitpath($illumina450k_csv_file);

if (!-e "$work_dir/$file") {
    download_url($illumina450k_csv_file, $work_dir);
}

my $array_info = parse_450k_file("$work_dir/$file");

load_array($dbh, $db_name, $species, $array_tag, $array_name, $array_info);

load_proxy_filter($dbh, $db_name, $array_tag, "GRCh37", $work_dir, $proxy_threshold);

exit();

sub download_url {
    my ($url, $work_dir) = @_;

    print "Downloading $url...\n";
    system("wget -q -N -P $work_dir $url");
}

sub parse_450k_file {
    my ($illumina450k_csv_file) = @_;
    my $array;

    open(CSV, $illumina450k_csv_file) or die;
    my $annotation;
    my @gene_annotations = ("TSS200", "TSS1500", "1stExon", "Body", "3'UTR", "5'UTR",
"IGR");
    my $cpg_annotations = {
        'N_Shelf' => "Shelf_Shore",
        'S_Shelf' => "Shelf_Shore",
        'N_Shore' => "Shelf_Shore",
        'S_Shore' => "Shelf_Shore",
        'Island' => "Island"};
    while (<CSV>) {
        chomp;
        my @data = split(",", $_);
        my $probe_id = $data[0];
        next if ($probe_id !~ /^cg/ and $probe_id !~ /^ch\./);
        my $probe_chr37 = $data[11];
        my $probe_loc37 = $data[12];
        my $probe_chr36 = $data[14];
        my $probe_loc36 = $data[15];
        my $this_gene_annotation_arrayStr = $data[23];
        my $this_cpg_annotation = $data[25]?$cpg_annotations->{$data[25]}:"NA";
    }
}

```

```

my $this_gene_annotation;
if (!defined($this_gene_annotation_arrayStr) or $this_gene_annotation_arrayStr eq
"" ) {
    $this_gene_annotation = "IGR";
} else {
    my @this_gene_annotations = split(";", $this_gene_annotation_arrayStr);
    foreach my $this_a (@this_gene_annotations) {
        if (grep {$_ eq $this_a} @gene_annotations) {
            $this_gene_annotation = $this_a;
            last;
        }
    }
}

$array->{$this_gene_annotation."-".$this_cpg_annotation}++;

$array->{$probe_id} = [$probe_chr36, $probe_loc36, $probe_chr37, $probe_loc37,
$this_gene_annotation, $this_cpg_annotation];
}
close(CSV);
# foreach my $this_a (keys %$annotation) { #@gene_annotations) {
#     print $annotation->{$this_a}, "\t", $this_a, "\n";
# }

return $array;
}

sub load_array_1 {
    my ($dbh, $code_name, $array_info) = @_;

    my $table_name = $code_name;
    $table_name =~ s/\W//g;
    $table_name = "array_$table_name";
    $dbh->do("CREATE TABLE IF NOT EXISTS $table_name (probe_id, location,
gene_group, cgi_group)");

    my $sth = $dbh->prepare("INSERT INTO array_$code_name VALUES (?, ?, ?, ?)");
    foreach my $this_probe_id (sort keys $array_info) {
        my ($chr, $loc, $gene_group, $cgi_group) = @{$array_info->{$this_probe_id}};
        my $location = "$chr:$loc-$loc";
        $sth->execute($this_probe_id, $location, $gene_group, $cgi_group);
    }
    $sth->finish();

    return $table_name;
}

sub load_array {
    my ($dbh, $db_name, $species_name, $array_tag, $array_name, $array_info) = @_;

    my $sth;

```

```

    $sth = $dbh->prepare("INSERT OR IGNORE INTO array (array_tag, array_name,
species_name) VALUES (?, ?, ?)");
    $sth->execute($array_tag, $array_name, $species_name);
    my $array_id = $dbh->last_insert_id("", "", "", "");
    $sth->finish();
    if ($array_id == 0) {
        $array_id = $dbh->selectrow_array("SELECT array_id FROM array WHERE
array_name = '$array_name' AND species_name = 'Homo sapiens'");
    }

    $sth = $dbh->prepare("INSERT OR IGNORE INTO assembly (species_name,
assembly_name) VALUES (?, ?)");
    $sth->execute($species_name, "GRCh37");
    $sth->execute($species_name, "NCBI36");
    $sth->finish();
    my $human36_assembly_id = $dbh->selectrow_array("SELECT assembly_id FROM
assembly WHERE species_name = 'Homo sapiens' AND assembly_name = 'NCBI36'");
    my $human37_assembly_id = $dbh->selectrow_array("SELECT assembly_id FROM
assembly WHERE species_name = 'Homo sapiens' AND assembly_name = 'GRCh37'");

    $sth = $dbh->prepare("INSERT OR IGNORE INTO probe_mapping_info (array_id,
assembly_id, url) VALUES (?, ?, ?)");
    $sth->execute($array_id, $human36_assembly_id, "");
    my $probe_mapping_human36_id = $dbh->last_insert_id("", "", "", "");
    $sth->execute($array_id, $human37_assembly_id, "");
    my $probe_mapping_human37_id = $dbh->last_insert_id("", "", "", "");
    $sth->finish();

    $sth = $dbh->prepare("INSERT OR IGNORE INTO probe_annotation_info (array_id,
gene_reference_name, cgi_reference_name, url) VALUES (?, ?, ?, ?)");
    $sth->execute($array_id, "RefSeq", "UCSC CpG Islands", "");
    # my $probe_annotation_id = $dbh->last_insert_id("", "", "", "");
    $sth->finish();

    open(CSV1, ">probe_mapping.csv") or die;
    open(CSV2, ">probe_annotation.csv") or die;
    # my $sth1 = $dbh->prepare("INSERT OR IGNORE INTO probe_mapping
(probe_mapping_id, probe_id, chr_name, position) VALUES (?, ?, ?, ?)");
    # my $sth2 = $dbh->prepare("INSERT OR IGNORE INTO probe_annotation
(probe_annotation_id, probe_id, gene_group, cgi_group) VALUES (?, ?, ?, ?)");
    foreach my $this_probe_id (sort keys $array_info) {
        my ($chr36, $loc36, $chr37, $loc37, $gene_group, $cgi_group) = @{$array_info-
>{$this_probe_id}};
        print CSV1 join(",", $probe_mapping_human36_id, $this_probe_id, "chr$chr36",
$loc36), "\n";
        print CSV1 join(",", $probe_mapping_human37_id, $this_probe_id, "chr$chr37",
$loc37), "\n";
        # print CSV2 join(",", $probe_annotation_id, $array_id, $this_probe_id, $gene_group,
$cgi_group), "\n";
        print CSV2 join(",", $array_id, $this_probe_id, $gene_group, $cgi_group), "\n";
        # $sth1->execute($probe_mapping_human36_id, $this_probe_id, "chr$chr36",
$loc36) if ($probe_mapping_human36_id);
        # $sth1->execute($probe_mapping_human37_id, $this_probe_id, "chr$chr37",
$loc37) if ($probe_mapping_human37_id);
    }

```

```
# $sth2->execute($probe_annotation_id, $this_probe_id, $gene_group, $cgi_group) if
($probe_annotation_id);
}
close(CSV1);
close(CSV2);
system("echo '.mode csv
.import probe_mapping.csv probe_mapping
.import probe_annotation.csv probe_annotation' | sqlite3 $db_name");
# $sth1->finish();
# $sth2->finish();

}

sub load_proxy_filter {
    my ($dbh, $db_name, $array_tag, $assembly_name, $work_dir, $distance_threshold) =
    @_;

    my $array_id = $dbh->selectrow_arrayref("SELECT array_id FROM array WHERE
array_tag = '$array_tag'")->[0];
    my $bed_file = dump_array_bed_file($dbh, $array_id, $assembly_name, $work_dir);
    my $this_output_bed_file = "$work_dir/array_${array_id}.proxy.bed";
    my $runstr = "$bedtools window -w $distance_threshold -a $bed_file -b $bed_file >
$this_output_bed_file";
    system($runstr) == 0 or die "Error while running bedtools: $?";

    my $sth = $dbh->prepare("INSERT OR IGNORE INTO proxy_filter_info (array_id,
description) VALUES (?, ?)");
    $sth->execute($array_id, "1kb");

    open(BED, $this_output_bed_file) or die;
    my $all_probes;
    my $mapping_probes;
    while (<BED>) {
        chomp;
        my ($chr1, $start1, $end1, $probe1, $chr2, $start2, $end2, $probe2) = split("\t", $_);
        $all_probes->{$probe1} = 1;
        if ($probe1 ne $probe2) {
            $mapping_probes->{$probe1}->{$probe2} = 1;
        }
    }
    close(BED);
    open(CSV, ">proxy_filter.csv") or die;
    foreach my $probe (sort keys %$all_probes) {
        if (defined($mapping_probes->{$probe})) {
            print CSV "$array_id,$probe,", join("|", sort keys %{$mapping_probes-
>{$probe}}), "\n";
        } else {
            print CSV "$array_id,$probe,NONE\n";
        }
    }
    close(CSV);
    system("echo '.mode csv
.import proxy_filter.csv proxy_filter' | sqlite3 $db_name");
}
```

```

}

sub dump_array_bed_file {
    my ($dbh, $this_array_id, $assembly_name, $work_dir) = @_;
    my $sql = "SELECT probe_mapping_id
                FROM probe_mapping_info
                JOIN assembly USING (assembly_id)
                WHERE array_id = $this_array_id
                AND assembly_name = '$assembly_name'";
    my $probe_mapping_id = $dbh->selectrow_array($sql);
    if (!$probe_mapping_id) {
        die "Cannot find a mapping for array $this_array_id and assembly
$assembly_name\n";
    }

    $sql = "SELECT probe_id, chr_name, position FROM probe_mapping WHERE
probe_mapping_id = $probe_mapping_id";
    my $probes = $dbh->selectall_arrayref($sql);

    my $bed_file = "$work_dir/array_${this_array_id}.bed";
    open(BED, ">$bed_file") or die;
    foreach my $this_probe (sort_sort_probes @$probes) {
        print BED join("\t", $this_probe->[1], $this_probe->[2], $this_probe->[2]+1,
$this_probe->[0]), "\n";
    }
    close(BED);

    return $bed_file;
}

sub _sort_probes {
    my $chr_a = $a->[1];
    my $chr_b = $b->[1];
    my $loc_a = $a->[2];
    my $loc_b = $b->[2];
    $chr_a =~ s/chr//;
    $chr_b =~ s/chr//;
    if ($chr_a eq $chr_b) {
        return $loc_a <=> $loc_b;
    } elsif ($chr_a =~ /^d/ and $chr_b =~ /^d/) {
        return $chr_a <=> $chr_b;
    } elsif ($chr_a =~ /^d/) {
        return -1;
    } elsif ($chr_b =~ /^d/) {
        return 1;
    } else {
        return $chr_a cmp $chr_b;
    }
}

exit();

==> pasted/database/load_dataset.pl <==

```



```
#!/usr/bin/env perl
use strict;
use warnings;

use Getopt::Long;
use DBI;

my $db_dir = ".";
my $db_name = "eforge_1.1.db";
my $dataset_name = "ENCODE";
my $dataset_tag = "encode";
my $decode_file = "encode.decode";
my $work_dir = "/tmp";
my $species_name = "Homo sapiens";
my $assembly_name = "GRCh37";
my $bedtools = "bedtools";

my $help;

my $desc = qq{load_dataset.pl [options]}
```

DESCRIPTION:

This script reads a 'decode' file which contains references to a list of samples, each of them corresponding to a given dataset. Each of these files is a BED or BED-like file that is read by bedtools to find overlaps with each and every array loaded in the eFORGE database.

Note that you *must* load the arrays first. If you want to include new arrays at a later date, you will have to reload all the datasets again (i.e. you will have to re-start from scratch).

Required parameters:

```
--tag <tag> or --dataset_tag <tag>
    the ID for this database. This needs to be unique.
--name <name> or --dataset_name <name>
    the name for this database. This can be a longer description. It will be used in the web
interface.
--decode_file <file>
    the file with the information about each sample in this dataset
```

Optional parameters:

```
--db_name <name>
    is the name of the SQLite file [def: $db_name]
--db_dir <path>
    is the location of the SQLite file [def: $db_dir]
--work_dir <path>
    is the location where temporary files will be downloaded/created [def: $work_dir]
--bedtools <name>
    is the name of the bedtools executable you want to use [def: $bedtools]
--species <species>
    is the name of the species [def: $species_name]
--assembly <assembly>
```

```

    is the name of the assembly [def: $assembly_name]
};

GetOptions(
    "help" => \ $help,
    "db_name=s" => \ $db_name,
    "db_dir=s" => \ $db_dir,
    "tag|dataset_tag=s" => \ $dataset_tag,
    "name|dataset_name=s" => \ $dataset_name,
    "decode_file=s" => \ $decode_file,
    "work_dir=s" => \ $work_dir,
    "species=s" => \ $species_name,
    "assembly=s" => \ $assembly_name,
);

if ($help) {
    print $desc;
    exit(0);
}

my $dsn = "dbi:SQLite:dbname=$db_dir/$db_name";
my $dbh = DBI->connect($dsn, "", "") or die $DBI::errstr;

system("mkdir -p $work_dir");

my $decode_table = get_decode_table($decode_file);

my $dataset_id = load_dataset($dbh, $species_name, $decode_table, $dataset_name,
$dataset_tag);

download_bed_files($decode_table, $work_dir);

my $arrays = get_all_arrays_for_species($dbh, $species_name);

foreach my $this_array (@$arrays) {
    my ($this_array_id, $this_array_name) = @$this_array;
    my $sorted_array_bed_file = dump_array_bed_file($dbh, $this_array_id,
$assembly_name, $work_dir);
    my $single_overlaps_bed_files = run_single_overlap_bedtools($bedtools, $work_dir,
$sorted_array_bed_file, $decode_table, $this_array_id, $this_array_name);
    my $concatenated_overlaps_bed_file = paste_files($single_overlaps_bed_files,
$work_dir, $this_array_id);
    my $final_overlaps_bed_file =
add_sum_column_to_concatenated_bedfile($concatenated_overlaps_bed_file, $work_dir,
$this_array_id);
    load_bitstrings($dbh, $db_name, $final_overlaps_bed_file, $this_array_id, $dataset_id);
}

#

exit();

my $input_bed_files = get_input_bed_files_from_decode_table($decode_file);

```

```

exit(0);

sub load_dataset {
    my ($dbh, $species_name, $decode_table, $dataset_name) = @_;
    my $dataset_id;

    my $sth;
    $sth = $dbh->prepare("INSERT OR IGNORE INTO dataset (dataset_tag, dataset_name,
species_name) VALUES (?, ?, ?)");
    $sth->execute($dataset_tag, $dataset_name, $species_name);
    $dataset_id = $dbh->last_insert_id("", "", "", "");
    $sth->finish();
    if ($dataset_id == 0) {
        $dataset_id = $dbh->selectrow_array("SELECT dataset_id FROM dataset WHERE
dataset_name = '$dataset_name' AND species_name = '$species_name'");
    }

    $sth = $dbh->prepare("INSERT OR IGNORE INTO sample (dataset_id, sample_order,
file, lab, datatype, cell, tissue, shortcell, individual, acc, url)
VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?)");
    my $sample_order = 1;
    foreach my $this_sample (@$decode_table) {
        $sth->execute($dataset_id,
            $sample_order,
            $this_sample->{file},
            $this_sample->{lab},
            $this_sample->{datatype},
            $this_sample->{cell},
            $this_sample->{tissue},
            $this_sample->{shortcell},
            $this_sample->{individual},
            $this_sample->{acc},
            $this_sample->{url});
        $sample_order++;
    }

    return($dataset_id);
}

sub load_bitstrings {
    my ($dbh, $db_name, $bed_file, $array_id, $dataset_id) = @_;

    # my $sql = "INSERT INTO probe_bitstring (array_id, probe_id, dataset_id, sum, bit)
    # VALUES (?, ?, ?, ?, ?)";
    # my $sth = $dbh->prepare($sql);
    open(BED, $bed_file) or die "Cannot open BED file <$bed_file>\n";
    open(CSV, ">probe_bitstring.csv") or die "Cannot open CVS temporary file
<probe_bitstring.csv>\n";
    while(<BED>) {
        chomp;
        my ($chr, $start, $end, $probe_id, $sum, $bitstring) = split("\t", $_);
        print CSV join(",", $array_id, $probe_id, $dataset_id, $sum, $bitstring), "\n";
    }
}

```

```
# $sth->execute($array_id, $probe_id, $dataset_id, $sum, $bitstring);
}
close(BED);
close(CSV);
system("echo '.mode csv
.import probe_bitstring.csv probe_bitstring' | sqlite3 $db_name");
# $sth->finish();
}
```

=head2 add_sum_column_to_concatenated_bedfile

Arg[1] : string \$concatenated_overlaps_bed_file (location of the input BED file with 0/1 flags on the 4th column)
Arg[2] : string \$work_dir (where to put the temporary files)
Example : my \$final_overlaps_bed_file =
add_sum_column_to_concatenated_bedfile(\$concatenated_overlaps_bed_file, \$work_dir);
Description : Reads the input BED file (\$concatenated_overlaps_bed_file) which contains a series of 0 and 1 flags in the 4th column. This function reads the number of ones in that column and include that value in the output BED file. The output BED file will contain that number in the 4th column and the series of flag in the 5th column.
Returns : string \$final_overlaps_bed_file (the location of the resulting BED file)
Exceptions : Dies if error when opening the files

=cut

```
sub add_sum_column_to_concatenated_bedfile {
    my ($concatenated_overlaps_bed_file, $work_dir, $array_id) = @_;
    my $final_overlaps_bed_file = "$work_dir/final_overlaps.array_${array_id}.bed";

    open(BED_IN, $concatenated_overlaps_bed_file) or die "Cannot open BED file
<$concatenated_overlaps_bed_file>\n";
    open(BED_OUT, ">$final_overlaps_bed_file") or die "Cannot open BED file
<$final_overlaps_bed_file>\n";
    while(<BED_IN>) {
        chomp;
        my ($this_chr, $this_start, $this_end, $this_probe, $this_bitstring) = split("\t", $_);
        my $num = $this_bitstring =~ tr/1/1/;
        print BED_OUT join("\t", $this_chr, $this_start, $this_end, $this_probe, $num,
$this_bitstring), "\n";
    }
    close(BED_IN);
    close(BED_OUT);

    return $final_overlaps_bed_file;
}
```

=head2 run_single_overlap_bedtools

Arg[1] : string \$bedtools (either full path or just the binary if in the \$PATH)
Arg[2] : string \$work_dir (where to put the temporary files)

```

Arg[3]    : string $sorted_450k_bed_file (location of the BED file with sorted 450K
features)
Arg[4]    : arrayref of hash $decode_table
Example   : my $single_overlaps_bed_files = run_single_overlap_bedtools($bedtools,
$work_dir, $sorted_450k_bed_file, $decode_table);
Description : Run bedtools on the sorted 450K features vs all the BED DNase features,
one at a time.
Returns   : arrayref of string $total_overlaps_bed_files (the locations of the resulting
BED files)
Exceptions : Dies if error when running bedtools

```

```
=cut
```

```

sub run_single_overlap_bedtools {
    my ($bedtools, $work_dir, $sorted_array_bed_file, $decode_table, $array_id,
$array_name) = @_;
    my $single_overlap_bed_files = [];

    foreach my $this_input_bed_file (map {$_->{"file"}} @$decode_table) {
        print "Overlap between $array_name and $this_input_bed_file...\n";
        my $this_output_bed_file = $this_input_bed_file;
        $this_output_bed_file =~ s/./+\\/;
        $this_output_bed_file =
"$work_dir/single_overlaps.array_${array_id}.$this_output_bed_file";
        my $runstr = "$bedtools intersect -c -a $sorted_array_bed_file -b
$work_dir/$this_input_bed_file > $this_output_bed_file";
        system($runstr) == 0 or die "Error while running bedtools: $?";
        push(@$single_overlap_bed_files, $this_output_bed_file);
    }

    return $single_overlap_bed_files;
}

```

```

sub get_decode_table {
    my ($decode_file) = @_;
    my $decode_table;

    open(DECODER, $decode_file) or die "Cannot open decode file <$decode_file>\n";
    my $whole_text = join("\n", <DECODER>);
    $whole_text =~ s/[\r\n]+/\n/g;
    my @lines = split("\n", $whole_text);
    close(DECODER);

    my @header = split("\t", shift(@lines));

    foreach my $this_line (@lines) {
        my @data = split("\t", $this_line);
        my $decode_record = {};
        for (my $i=0; $i<@header; $i++) {
            $decode_record->{$header[$i]} = $data[$i];
        }
        # Check that file and URL match
    }
}

```

```

        die "File: ".$decode_record->{"file"}."\nURL: ".$decode_record->{"url"}."\nOn line:
$this_line"
        if ($decode_record->{"url"} !~ $decode_record->{"file"});
        $decode_record->{"file"} = $decode_record->{"url"};
        $decode_record->{"file"} =~ s/./\//g;
        push(@$decode_table, $decode_record);
    }

    return $decode_table;
}

sub download_bed_files {
    my ($decode_table, $work_dir) = @_;

    foreach my $this_decode_entry (@$decode_table) {
        my $url = $this_decode_entry->{"url"};
        my $file = $this_decode_entry->{"file"};
        if (!-e "$work_dir/$file") {
            download_url($url, $work_dir);
        }
    }
}

sub download_url {
    my ($url, $work_dir) = @_;

    print "Downloading $url...\n";
    system("wget", "-q", "-N", "-P", $work_dir, $url) == 0
        or die "Error: wget -q -N -P $work_dir $url\n!";
}

sub paste_files {
    my ($files, $work_dir, $array_id) = @_;
    my $output_file = "$work_dir/concat_overlaps.array_${array_id}.bed";

    my $max_num_files = 200;
    my $c = 0;
    my $temp_output_file;
    my @original_files = @$files;
    while (@$files > 1) {
        $c++;
        $temp_output_file = "$work_dir/temp.$$.$c.paste_files.txt";
        my $input_files = [splice(@$files, 0, $max_num_files)];
        # print STDERR "Merging ", join(", ", @$input_files), " into $temp_output_file\n";
        merge_files($temp_output_file, $input_files);
        unshift(@$files, $temp_output_file);
    }
    rename($temp_output_file, $output_file);

    unlink glob "$work_dir/temp.$$.*.paste_files.txt";

#

```

```

#
#
#
#   unlink @original_files;
#
#
#
#

    return $output_file;
}

sub merge_files {
    my ($output_file, $input_files) = @_;
    my @fhs;
    my $c = 0;

    foreach my $this_input_file (@$input_files) {
        open($fhs[$c++], $this_input_file) or die "Cannot open $this_input_file: $!\n";
    }
    open(OUT, ">$output_file") or die "Cannot open $output_file: $!\n";

    my $first_fh = shift(@fhs);

    while (1) {
        my $line = <$first_fh>;
        chomp($line);
        my ($chr, $start, $end, $probe_id, $value) = split("\t", $line);
        print OUT join("\t", $chr, $start, $end, $probe_id, $value);
        foreach my $this_fh (@fhs) {
            my $line = <$this_fh>;
            chomp($line);
            my ($this_chr, $this_start, $this_end, $this_probe_id, $this_value) = split("\t",
$line);
            if ((($this_chr ne $chr) or ($this_start != $start) or ($this_end != $end) or
($this_probe_id ne $probe_id)) {
                die "Files do not contain the same $chr-$start-$end-$probe_id lines\n";
            }
            print OUT $this_value;
        }
        print OUT "\n";
        if (eof($first_fh)) {
            last;
        }
    }

    foreach my $this_fh (@fhs) {
        close($this_fh);
    }
    close(OUT);
}

sub get_all_arrays_for_species {

```

```

my ($dbh, $species) = @_;
my $arrays;

my $sth = $dbh->prepare("SELECT array_id, array_name FROM array WHERE
species_name = ?");
$sth->execute($species);
$arrays = $sth->fetchall_arrayref();
$sth->finish();

return $arrays;
}

sub get_probe_mapping_id {
my ($dbh, $this_array_id, $assembly_name) = @_;
my $sql = "SELECT probe_mapping_id
FROM probe_mapping_info
JOIN assembly USING (assembly_id)
WHERE array_id = $this_array_id
AND assembly_name = '$assembly_name'";
my $probe_mapping_id = $dbh->selectrow_array($sql);
return($probe_mapping_id);
}

sub dump_array_bed_file {
my ($dbh, $this_array_id, $assembly_name, $work_dir) = @_;
my $sql = "SELECT probe_mapping_id
FROM probe_mapping_info
JOIN assembly USING (assembly_id)
WHERE array_id = $this_array_id
AND assembly_name = '$assembly_name'";
my $probe_mapping_id = $dbh->selectrow_array($sql);

$sql = "SELECT probe_id, chr_name, position FROM probe_mapping WHERE
probe_mapping_id = $probe_mapping_id";
my $probes = $dbh->selectall_arrayref($sql);

my $bed_file = "$work_dir/array_${this_array_id}.bed";
open(BED, ">$bed_file") or die "Cannot open temporary BED file <$bed_file>\n";
foreach my $this_probe (sort_sort_probes @$probes) {
print BED join("\t", $this_probe->[1], $this_probe->[2], $this_probe->[2]+1,
$this_probe->[0]), "\n";
}
close(BED);

return $bed_file;
}

sub _sort_probes {
my $chr_a = $a->[1];
my $chr_b = $b->[1];
my $loc_a = $a->[2];
my $loc_b = $b->[2];
$chr_a =~ s/chr//;
$chr_b =~ s/chr//;

```



```
if ($chr_a eq $chr_b) {  
    return $loc_a <=> $loc_b;  
} elsif ($chr_a =~ /\d/ and $chr_b =~ /\d/) {  
    return $chr_a <=> $chr_b;  
} elsif ($chr_a =~ /\d/) {  
    return -1;  
} elsif ($chr_b =~ /\d/) {  
    return 1;  
} else {  
    return $chr_a cmp $chr_b;  
}  
}  
  
exit();
```