Genomic Foundation Models

Josh Meehl

2025-12-01

Table of contents

Introduction

> **Warning**
>
> This book is in active development. Sections and examples may change as the field and my understanding evolve.

We can now sequence a human genome for a few hundred dollars and store millions of genomes in a single biobank. What we cannot yet do, reliably, is tell you what most of those variants mean. The gap between sequencing capacity and interpretive capacity defines the central problem of modern genomics. It is exactly the gap that genomic foundation models aim to close.

Meanwhile, deep learning has transformed how we represent language, proteins, and now DNA itself. Large models trained on broad sequence data can now be adapted to tasks ranging from variant interpretation to clinical risk prediction, all without retraining from scratch for each new problem.

This book is about that intersection: **genomic foundation models (GFMs)** - large, reusable models trained on genomic and related data that can be adapted to many downstream tasks. Rather than offering a general introduction to genomics or machine learning, the goal is narrower and more opinionated:

> To give you a *conceptual and practical map* of how modern deep models for DNA, RNA, and proteins are built, what they actually learn, and how they can be used responsibly in research and clinical workflows.

The chapters that follow connect classic genomics pipelines, early deep regulatory models, sequence language models, and multi-omic GFMs into a single narrative arc.

Why Genomic Foundation Models?

Traditional genomic modeling has usually been **task-specific**:

- A variant caller tuned to distinguish sequencing errors from true variants.
- A supervised CNN trained to predict a fixed set of chromatin marks.
- A risk score fit for one trait, in one ancestry group, in one health system.

These models can work very well in the setting they were designed for, but they often do not transfer gracefully to new assays, tissues, ancestries, or institutions.

The **foundation model** paradigm takes a different view:

1. **Scale**
   Train large models on massive, heterogeneous datasets, across assays, tissues, species, and cohorts, so they learn reusable structure.
2. **Self-supervision**
   Use objectives such as masked-token prediction, next-token modeling, or contrastive learning that do not require manual labels, allowing us to exploit unlabeled genomes, perturbation screens, and population variation.
3. **Reusability**
   Treat the model as a *backbone*: for new tasks, we probe, adapt, or fine-tune the same representation instead of training a new model from scratch.

In genomics, this paradigm is still evolving and far from settled. Some tasks benefit dramatically from pretraining; others barely move beyond strong classical baselines. This book leans into that tension and asking when foundation models actually help, and when simpler approaches suffice (Bommasani et al. 2022; Guo et al. 2025).

Recurring Themes

Several threads run through the book; individual chapters can be read as different views of the same underlying questions.

Data and Architecture Co-evolve

We will see how:

- Early deleteriousness scores built on hand-engineered features and shallow models.
- CNNs enabled direct learning of regulatory "motifs" and local grammar from raw sequence.
- Transformers and other long-context models opened the door to capturing broader regulatory neighborhoods and chromatin structure.
- GFMs push toward representations that span multiple assays, tissues, and even organisms.

At each stage, the interesting question is not "Is this model fancier?" but "How does the available data constrain what the model can sensibly learn?"

Context Length and Genomic Geometry

Many genomic phenomena are intrinsically non-local: enhancers regulating distant genes, looping interactions, polygenic effects spread across the genome. The book returns repeatedly to "how far" a model can see, how it represents long-range dependencies, and what is gained (and lost) as context windows and architectures scale.

Prediction Versus Design

Most current models are used as **predictors**: given sequence and context, what happens? But the same models can be embedded in **design** and **closed-loop** workflows, from variant prioritization to sequence or library design. We will explore how foundation models change the boundary between analysis and experimental planning, and what new failure modes emerge in the process.

From Benchmarks to Decisions

Benchmark scores are seductive and easy to compare. Real biological and clinical decisions are messy, multi-objective, and often constrained by data drift, bias, and poorly specified endpoints. A recurring theme is the gap between "state-of-the-art AUC" and actual impact—and how careful evaluation, confounder analysis, and calibration can narrow that gap.

Interpretability and Mechanism

Finally, we return often to interpretability, not as optional decoration, but as a design constraint. We will ask when saliency maps, motif extraction, or more mechanistic analyses genuinely deepen understanding, and when they simply provide a veneer of comfort over confounded or brittle models.

How the Book Is Organized

The book is organized into six parts plus three short appendices. Each part can be read on its own, but they are designed to build on one another.

Part I — Data & Pre-DL Methods

Part I lays the **genomic and statistical foundation** that later models rest on.

- Chapter 1 introduces next-generation sequencing, alignment, and variant calling, highlighting sources of error and the evolution from hand-crafted pipelines to learned variant callers.

- Chapter 2 surveys the core data resources that underlie most modern work: reference genomes, population variation catalogs, clinical variant databases, and functional genomics consortia. It also discusses how they are used as training targets and evaluation benchmarks.
- Chapter 3 reviews genome-wide association studies, linkage disequilibrium, fine-mapping, and polygenic scores, emphasizing what these "variant-to-trait" associations do and do not tell us.
- Chapter 4 covers conservation-based and machine learning-based variant effect predictors such as CADD, including their feature sets, label construction, and issues like circularity and dataset bias.

Together, Part I answers: *What data and pre-deep-learning tools form the backdrop that any genomic foundation model must respect, integrate with, or improve upon?*

Part II — CNN Seq-to-Function Models

Part II turns to the first wave of **deep sequence-to-function models**, largely built on convolutional neural networks.

- Chapter 5 presents CNN-based models that predict chromatin accessibility, histone marks, and related regulatory annotations directly from DNA sequence, and explores what they learn about motifs and regulatory grammar.
- Chapter 6 extends from chromatin to gene expression, showing how models combine sequence, regulatory features, and context to predict expression levels and perturbation effects.
- Chapter 7 focuses on deep models of pre-mRNA splicing and splice-site choice, and how these models can be used to interpret variant effects on splicing in both research and clinical contexts.

Part III — Transformer Models

Part III introduces **transformer-based and related architectures** for representing biological sequence.

- Chapter 8 examines how we turn genomic and protein sequences into model-compatible tokens, including k-mers, byte-pair encodings, and other schemes, and how these choices shape downstream models.
- Chapter 9 describes large protein language models trained on sequence databases, their emergent structure and function representations, and applications to variant effect prediction and protein design.
- Chapter 10 surveys DNA language models and other genomic foundation backbones, including their training corpora, objectives, evaluation suites, and limitations.

- Chapter 11 covers hybrid CNN/transformer and related architectures designed to handle long genomic contexts, such as models that predict regulatory readouts over tens to hundreds of kilobases.

Part IV — GFMs & Multi-omics

Part IV is the conceptual core of the book, focusing explicitly on **genomic foundation models** and their multi-omic extensions.

- Chapter 12 provides a working definition and taxonomy of genomic FMs, design dimensions (architecture, context length, conditioning), and practical guidance for using pretrained backbones in downstream tasks…
- Chapter 13 recasts variant effect prediction in the foundation-model era, spanning protein and DNA-based approaches, and discusses calibration, uncertainty, and integration into existing pipelines.
- Chapter 14 broadens the view from isolated sequences to multi-omic and systems-level representations, including models that integrate genomic, transcriptomic, proteomic, and phenotype data.

Part V — Reliability & Interpretation

Part V pulls out cross-cutting issues that apply to essentially every model in the book.

- Chapter 15 develops a unified framework for evaluating models across molecular, variant-level, trait-level, and clinical tasks, and discusses data splitting, metric choice, and the link between benchmarks and real-world decisions.
- Chapter 16 details sources of confounding and data leakage, from batch effects and ancestry structure to label bias and covariate shift, and offers practical strategies for detection and mitigation.
- Chapter 17 explores interpretability tools from classical motif discovery and attribution methods to emerging mechanistic approaches, and asks when these tools genuinely reveal biological mechanisms.

Part VI — Applications

Part VI moves from methods to **end-to-end workflows** in research and clinical practice.

- Chapter 18 discusses risk prediction tasks that combine genomic features (including outputs from GFMs) with clinical and environmental data, focusing on discrimination, calibration, fairness, and deployment in health systems.
- Chapter 19 examines how models fit into rare disease and cancer workflows, including variant prioritization pipelines, integration with family and tumor-normal data, and lab-in-the-loop validation.
- Chapter 20 looks at how GFMs intersect with target discovery, functional genomics screens, biomarker development, and biotech/industry workflows, including build-vs-buy and organizational considerations.

Appendices

Two short appendices provide background and pointers:

- Appendix A is a compact introduction to neural networks, CNNs, transformers, training, and evaluation, aimed at genomics-first readers who want enough ML background to engage with the main chapters. :contentReferenceoaicite:2
- Appendix B is a curated set textbooks, courses, software, and papers for deeper dives into genomics, statistical genetics, and deep learning.
- Appendix C is a glossary of key terms.

A Moving Target

Genomic foundation models are a moving target: architectures, datasets, and evaluation suites are evolving quickly. This book is not intended as a frozen survey of "the state of the art," but as a framework for reasoning about new models as they appear.

If it succeeds, you should finish able to:

- Place a new model in the landscape of data, architecture, objective, and application.
- Design analyses and experiments that use GFMs as components—features, priors, or simulators—without overclaiming what they can do.

- Recognize common pitfalls in training, evaluation, and deployment, especially in clinical and translational settings.

- Decide where foundation models are genuinely useful, and where simpler methods or classical baselines are sufficient.

The next chapter now turns to the foundations: how we get from raw reads to variants, and from variants to the datasets and benchmarks on which all of these models depend.

Preface

Working on genomic foundation models means context-switching constantly: debugging data artifacts one week, reproducing a transformer-based variant effect predictor the next, and arguing about clinical patient cohorts the week after. The knowledge required is scattered across textbooks, methods papers, and tribal folklore - genomics on one shelf, deep learning on another, clinical deployment in someone else's head entirely.

This book is my attempt to put those pieces in one place: to connect the mature, statistically grounded tradition of human genetics with the rapidly changing ecosystem of deep learning and foundation models, and to make that transition legible for people who live in one corner of the triangle and are trying to get oriented to the others.

I wrote it first for myself and my collaborators: as a way to organize wiki pages, markdown files, and half-finished slide decks into something coherent. Over time it became clear that turning those notes into a book might be useful to others navigating the same landscape.

Why I Wrote This Book

What I wanted, but could not find, was a **conceptual throughline**:

- How do we get from reads to variants in a way that a deep model can trust?
- How should we think about polygenic scores, fine-mapping, and functional assays in the era of foundation models?
- When we say a model "understands" regulatory grammar or protein function, what does that actually mean?
- And what does it take to move from a promising preprint to a tool that can support decisions about real patients?

This book is my best attempt at answering those questions in a way that is historically grounded, technically honest, and practically oriented.

How This Book Came Together

The structure of the book reflects the way these ideas evolved in my own work.

Early sections grew out of teaching and mentoring conversations: explaining next-generation sequencing, variant calling, and pre-deep-learning interpretation methods to new team members who were strong in statistics or ML but new to genomics (and vice versa).

The middle sections emerged from a series of "journal club + experiments" cycles, where we:

- read papers on sequence-to-function CNNs, protein language models, and genomic transformers,
- tried to reproduce key results or adapt them to key datasets,
- and documented the pain points—data formats, training instabilities, evaluation pitfalls, which never quite fit into a methods section.

The later parts were shaped by collaborations around clinical prediction, variant interpretation pipelines, and larger multi-omic models. Many of the examples and caveats come directly from these projects: places where a model that looked excellent on paper behaved in surprising ways when exposed to real-world data, or where simple baselines outperformed much fancier architectures once confounding and distribution shift were handled correctly.

Because of that origin, the book has a particular bias: it is written from the perspective of someone who spends much of their time trying to get models to work in messy, high-stakes settings. You will see this in the emphasis on data quality, evaluation, and clinical translation.

How to Read This Book

This is **not** a genomics textbook, a complete review of every DNA or protein model, or a deep-learning-from-scratch course. Instead, it is meant to be:

- a **roadmap** to the main kinds of data, models, and objectives that matter for genomic foundation models today
- a **bridge** between classical statistical genetics and modern representation learning
- a **practical guide** to the kinds of failure modes and design choices that matter in real applications.

You do **not** need to read the book cover-to-cover in order.

- If your background is in **genomics or statistical genetics**, you may want to skim the early deep-learning motivations and focus more on the sections that introduce convolutional models, transformers, and self-supervision, then move on to evaluation and applications.
- If you come from **machine learning**, it may be more helpful to start with the genomic data and pre-deep-learning methods, then dive into the sequence-to-function and transformer-based chapters with an eye toward how the data and objectives differ from text or images.
- If you are a **clinician or translational researcher**, you might care most about the reliability, confounding, and clinical deployment discussions, dipping back into the modeling parts as needed to interpret results or communicate with technical collaborators.

The book is organized into six parts:

- **Part I** introduces genomic data and pre-deep-learning interpretation methods, from sequencing and variant calling to early pathogenicity scores and polygenic models.
- **Part II** focuses on supervised sequence-to-function models, with an emphasis on convolutional architectures, regulatory prediction, and splicing.
- **Part III** turns to transformer-based models and self-supervision, covering protein and DNA language models and hybrid architectures that combine CNNs and transformers.
- **Part IV** discusses what makes a model a *foundation model* in genomics, including multi-omic architectures, variant effect modeling, and emergent capabilities.
- **Part V** examines reliability, evaluation, confounding, and interpretability—how we know whether a model is learning what we think it is, and how to detect when it is not.
- **Part VI** looks at applications: clinical and risk prediction, variant interpretation workflows, and early steps toward drug discovery and biotech use cases.

Within each part, the goal is not to catalogue every paper, but to highlight representative examples and the design principles they illustrate. References are there to give you starting points, not to serve as a comprehensive literature review.

What This Book Assumes (and What It Does Not)

The book assumes:

- basic familiarity with probability and statistics (regression, hypothesis testing, effect sizes),
- core genomics concepts (genes, variants, linkage disequilibrium, GWAS at a high level),
- and some exposure to machine learning ideas (training versus test data, overfitting, loss functions).

It **does not** assume that you have implemented deep learning models yourself, or that you are fluent in every area. When a chapter leans heavily on a particular background (for example, causal inference or modern self-supervised learning), it will either provide a brief refresher or point you to an appendix or external resource.

If you are missing some of this background, that is fine. The intent is for you to be able to read actively: to pause, look up side topics, and then return to the main arc without feeling lost.

A Note on Scope and Opinions

Genomic foundation models are evolving quickly. Any snapshot is, by definition, incomplete and slightly out of date.

Rather than chasing every new architecture or benchmark, the book focuses on **durable ideas**:

- how different data types fit together,
- what kinds of objectives encourage useful representations,
- how evaluation can fail in genomics-specific ways,
- and where deep models complement (rather than replace) classical approaches.

Inevitably, there are judgment calls about which papers, methods, and perspectives to emphasize. Those choices reflect my own experiences and biases. They are not an official position of any institution I work with, and they will certainly differ from other reasonable views in the field.

You should treat the book as one opinionated map of the landscape, not the landscape itself.

Acknowledgements

This book exists because of many generous people who shared their time, ideas, and encouragement.

First, I owe a deep debt of gratitude to my colleagues in the **Mayo Clinic GenAI** and broader data science community. The day-to-day conversations, whiteboard sessions, and "what went wrong here?" post-mortems with this group shaped much of the perspective and many of the examples in the chapters.

I am especially grateful to the **principal investigators and clinicians** whose questions kept the focus on real patients and real decisions:
Dr. Shant Ayanian, **Dr. Elena Myasoedova**, and **Dr. Alexander Ryu**.

To **leadership at Mayo Clinic** who supported the time, computing resources, and institutional patience needed for both the models and this book:
Dr. Matthew Callstrom, **Dr. Panos Korfiatis**, and **Matt Redlon**.

To my **data science and machine learning engineering colleagues**, whose work and feedback directly shaped many of the workflows and case studies:
Bridget Toomey, **Carl Molnar**, **Zach Jensen**, and **Marc Blasi**.

I am also grateful for the architectural creativity, hardware insight, and willingness to experiment from our **collaborators at Cerebras**:
Natalia Vassilieva, **Jason Wolfe**, **Omid Shams Solari**, **Vinay Pondenkandath**, **Bhargav Kanakiya**, and **Faisal Al-khateeb**.

And to our **collaborators at GoodFire**, whose partnership helped push these ideas toward interpretable and deployable systems:
Daniel Balsam, **Nicholas Wang**, **Michael Pearce**, and **Mark Bissell**.

I would also like to thank my former colleagues at **LGC** for foundational work and conversations around protein language models and large-scale representation learning:
Prasad Siddavatam and **Robin Butler**.

Beyond these named groups, I owe a broader debt to the geneticists, molecular biologists, statisticians, clinicians, and engineers whose work this book draws on. The field moves forward because people share code, publish honest benchmarks, and insist that models be connected back to biologically meaningful questions. Thank you for setting that standard.

Finally, I am grateful to my wife, Alyssa, and our two kids for their patience with the evenings and weekends this book consumed. You gave me the space to finish it and the reasons to step away from it.

If this book helps you connect a new model to a real biological question, design a more robust evaluation, or communicate more clearly across disciplinary boundaries then it will have done its job.

— *Josh Meehl*

Part I.

Part I: Data & Pre-DL Methods

1. Sequencing: From Reads to Variants

> ⚠ Warning
>
> **TODO:**
>
> - Citations:
>   - ...
> - Add notes on imputaition and boosting

1.1. The Challenge of NGS Data

Next-generation sequencing (NGS) has transformed genomics by making it routine to generate tens to hundreds of gigabases of sequence from a single individual in a few days. Modern instruments produce short reads—typically 100–300 bp paired-end Illumina reads—at very high throughput, but with non-trivial error profiles including substitutions, context-specific errors, and base quality uncertainties. These reads are then aligned to imperfect reference genomes that omit structural variation and some segmental duplications (Goodwin, McPherson, and McCombie 2016).

Turning these raw reads into a reliable list of variants is therefore not just a matter of comparing strings. Variant calling pipelines must disentangle sequencing errors (instrument noise, PCR artifacts), alignment artifacts (mis-mapping in repeats, paralogous regions, pseudogenes), and genuine biological variation (germline variants, somatic mutations, mosaicism). Historically, this was addressed by complex, modular pipelines combining probabilistic models and hand-crafted heuristics (Nielsen et al. 2011). Deep learning now plays an important role in simplifying and improving parts of this stack, but it is helpful to understand the classical pipeline first.

1.2. Targeting Strategies: Panels, Exomes, and Genomes

NGS is not a single technology; it is deployed in different targeting strategies, each with distinct trade-offs.

1.2.1. Targeted and Panel Sequencing

Targeted gene panels capture tens to hundreds of genes selected for a clinical indication such as cardiomyopathy or hereditary cancer syndromes. These panels offer high depth in a limited region

(often 200–500×), relatively low cost per sample, and simple interpretation workflows tied to well-curated gene lists. However, panels miss novel genes outside their design, most structural variants and non-coding regulatory changes, and opportunities for reanalysis as gene–disease knowledge evolves.

1.2.2. Whole-Exome Sequencing

Whole-exome sequencing (WES) enriches coding exons and some flanking splice regions genome-wide. Typical coverage ranges from 80–150× for exonic targets, though capture efficiency varies across GC content and repetitive exons. Non-coding regions are largely unobserved.

WES has been especially successful for Mendelian disease gene discovery and diagnostic workflows. At the same time, it misses non-coding and structural causes, has non-uniform coverage leading to heterogeneous sensitivity across genes, and requires careful handling of capture biases and batch effects.

1.2.3. Whole-Genome Sequencing

Whole-genome sequencing (WGS) samples nearly all bases in the genome. Typical coverage is 30–60× across the genome, with more uniform depth than WES. Because there is no capture step, WGS produces fewer batch-specific artifacts and enables detection of non-coding variants, structural variants, and copy-number changes along with SNVs and indels.

WGS is increasingly favored for new large cohorts and rare disease diagnostics despite higher cost, because the data are reusable for many downstream analyses (GWAS, PGS, rare variant burden tests), it simplifies pipelines by eliminating the need to track changing capture designs, and it supports more complete variant catalogs for the models discussed later in this book.

Throughout this text, we assume a WES/WGS-style pipeline where we start from aligned reads and aim to call high-confidence SNVs and small indels.

1.2.4. Long-Read Sequencing Technologies

While short-read Illumina sequencing dominates population-scale studies, long-read technologies are increasingly important for resolving complex genomic regions and structural variation.

Pacific Biosciences (PacBio) HiFi produces reads of 10–25 kb with accuracy exceeding 99.9% through circular consensus sequencing (Wenger et al. 2019). These reads can span repetitive elements, segmental duplications, and structural variants that confound short-read alignment. Oxford Nanopore Technologies (ONT) generates ultra-long reads—routinely 10–100 kb, with some exceeding 1 Mb—at somewhat lower per-base accuracy (roughly 95–99% for recent chemistries). ONT's portability and real-time sequencing enable novel applications ranging from field diagnostics to direct RNA sequencing (Dabernig-Heinz et al. 2024).

Long reads transform variant calling in several ways. Structural variants—deletions, insertions, inversions, and complex rearrangements that are invisible or ambiguous in short-read data—become directly observable. Tandem repeats, segmental duplications, and transposable element insertions can be traversed rather than collapsed. A single long read can span many heterozygous sites,

enabling direct, read-backed phasing over tens of kilobases. The T2T-CHM13 reference genome, completed with long reads, added approximately 200 Mb of previously unresolved sequence, including centromeres and acrocentric chromosome arms (Nurk et al. 2022).

Specialized variant callers have been developed for long-read data. DeepVariant includes models trained on PacBio and ONT data, while tools like Clair3 and PEPPER-Margin-DeepVariant are optimized for nanopore error profiles (Poplin et al. 2018; Z. Zheng et al. 2022; Shafin et al. 2021). For structural variants, callers such as Sniffles, pbsv, and cuteSV exploit the unique properties of long reads (Smolka et al. 2024; "PacificBiosciences/Pbsv" 2025; Jiang et al. 2020).

This chapter focuses on short-read pipelines, which remain the workhorse for large cohorts and cost-sensitive applications. However, hybrid approaches—combining short-read depth with long-read phasing and structural variant detection—are increasingly common, and the models in later chapters must accommodate variants discovered by either technology.

1.3. Classical Variant Calling Pipelines

While every institution implements its own details, a classical short-read pipeline has several common stages.

The process begins with **base calling and demultiplexing**, where instrument software converts fluorescent images to base calls and quality scores, and reads are demultiplexed by barcode into sample-specific FASTQ files.

Next comes **read alignment**, in which short reads are aligned to a reference genome (such as GRCh38 or T2T-CHM13) using seed-and-extend mappers such as BWA-MEM or minimap2 (Heng Li 2013, 2018). Aligners must cope with mismatches, small indels, and repetitive sequence.

Post-alignment processing follows, including marking or removing PCR duplicates, base quality score recalibration (BQSR) to model systematic quality score errors, and local realignment around indels in older pipelines.

Per-sample variant calling then takes place, where tools like the Genome Analysis Toolkit (GATK) HaplotypeCaller fit local haplotypes using hidden Markov models, de Bruijn graphs, or other probabilistic frameworks (McKenna et al. 2010). These tools produce candidate variants with genotype likelihoods for each sample.

Finally, for **cohort variant calling**, joint genotyping and cohort refinement recombines per-sample likelihoods to enforce a consistent set of variants across individuals. Raw calls are then filtered to distinguish true variants from artifacts. Simple hard filters apply fixed cutoffs to individual metrics, but GATK's Variant Quality Score Recalibration (VQSR) takes a more sophisticated approach: it fits a Gaussian mixture model in the multi-dimensional space of variant annotations (depth, strand bias, mapping quality, read position bias, etc.), using variants at known polymorphic sites as a training set. Each candidate receives a recalibrated score reflecting how well it matches the learned "true variant" distribution, allowing joint consideration of multiple quality axes rather than independent hard thresholds.

These steps are encoded in pipelines like GATK Best Practices and similar frameworks. The key point is that each step uses hand-designed summary features and mechanistic models chosen by experts, not learned end-to-end (Van der Auwera et al. 2018).

1.3.1. Probabilistic Framework

At the core of GATK's HaplotypeCaller is a Bayesian genotype likelihood model. For a candidate genotype G at a given site, the posterior probability given the read data D is:

$$P(G \mid D) \propto P(G) \prod_{r \in \text{reads}} P(r \mid G)$$

where $P(G)$ is a prior over genotypes (often assuming Hardy-Weinberg equilibrium) and $P(r \mid G)$ is the likelihood of observing read r given genotype G. Computing $P(r \mid G)$ is non-trivial: GATK uses a pair hidden Markov model (pair-HMM) to marginalize over possible alignments between the read and candidate haplotypes, incorporating base quality scores to weight the contribution of each base.

This formulation assumes conditional independence of reads given the genotype—an assumption known to be violated in practice due to correlated errors from PCR duplicates, systematic instrument biases, and local sequence context (DePristo et al. 2011). DeepVariant's CNN (see Section 1.8), by contrast, sees all reads in a pileup simultaneously and can learn to model these dependencies implicitly.

1.4. Haplotype Phasing

Diploid organisms carry two copies of each autosomal chromosome—one inherited from each parent. Standard variant calling produces unphased genotypes: we know that an individual is heterozygous at two nearby sites (say, A/G and C/T), but not which alleles reside on the same physical chromosome. Phasing resolves this ambiguity by assigning each allele to a specific haplotype.

1.4.1. Why Phasing Matters

Phased haplotypes are essential for multiple applications. In recessive disease, two deleterious variants in the same gene cause disease only if they are on different haplotypes (in *trans*); unphased calls cannot distinguish *cis* from *trans* configurations, making compound heterozygosity detection impossible without phase information. Some regulatory and coding effects depend on the combination of alleles on a single chromosome, enabling haplotype-aware variant effect prediction. Reference panels used for genotype imputation (such as TOPMed and 1000 Genomes) are stored as phased haplotypes, and accurate phasing improves imputation quality. Finally, haplotype structure carries information about population history, recombination, and natural selection that drives ancestry and selection analyses.

1.4.2. Phasing Methods

Phasing can be achieved through several approaches. **Read-backed phasing** exploits physical linkage: when heterozygous variants are close enough to be spanned by the same sequencing read or read pair, the linkage directly reveals phase. Short-read data can phase variants within a few hundred base pairs, while long reads extend this to tens of kilobases or more.

Statistical or population-based phasing tools such as SHAPEIT, Eagle, and Beagle infer phase by leveraging linkage disequilibrium patterns observed in reference panels (O'Connell et al. 2014; Loh et al. 2016; Browning et al. 2021). These methods are highly accurate for common variants but struggle with rare variants that lack informative LD.

Pedigree-based phasing becomes possible when parent–offspring trios or larger pedigrees are available; Mendelian inheritance rules can resolve phase with high confidence.

Long-read and linked-read technologies provide direct observation of haplotype structure. PacBio HiFi and Oxford Nanopore reads span tens of kilobases, while linked-read methods (such as the now-discontinued 10x Genomics platform) tag short reads originating from the same long DNA molecule, providing intermediate-range phasing.

Modern pipelines often combine these approaches: statistical phasing across the genome, refined by read-backed evidence where available, and augmented by trio data when present. The result is a phased VCF with haplotype assignments (for example, `0|1` rather than `0/1`), which downstream analyses can exploit.

1.5. Sources of Error and Uncertainty

Even with modern pipelines, variant calls are imperfect. Understanding the important failure modes is essential for interpreting downstream analyses (Heng Li 2014).

Mapping ambiguity arises when reads from segmental duplications, paralogous genes, and low-complexity regions are mis-aligned. Reference bias can favor the reference allele in ambiguous regions, causing systematic undercalling of alternate alleles.

Systematic sequencing artifacts include context-specific errors in homopolymers and GC-rich regions, as well as batch effects across runs, instruments, or library preparations. These artifacts can create correlated false positives that are difficult to filter.

Low-coverage regions present another challenge. WES capture dropouts or WGS coverage dips can create false negatives for heterozygous variants, and somatic or mosaic variants at low allele fraction can be mistaken for noise.

Complex variants are also problematic. Small indels near homopolymers or repetitive elements are difficult to call accurately, and multi-nucleotide variants may be decomposed into multiple SNVs depending on the caller's representation choices.

The deep learning models in later chapters inherit these errors as input noise. Understanding where variant calls are reliable—and where they are not—is essential when training sequence-to-function models, building polygenic scores, or interpreting predicted variant effects.

1.6. Difficult-to-Call Regions

Not all genomic regions are created equal. Some areas of the genome are systematically problematic for short-read variant calling due to their sequence properties.

1.6.1. Segmental Duplications and Paralogs

Regions with high sequence identity to other parts of the genome cause reads to map ambiguously. Paralogous genes—such as *SMN1* and *SMN2*, or *CYP2D6* and its pseudogenes—are particularly challenging. A read originating from one copy may align equally well to another, leading to false variant calls or missed true variants.

1.6.2. Low-Complexity and Repetitive Sequence

Homopolymers, short tandem repeats, and other low-complexity regions have elevated error rates on most sequencing platforms. Indel calling in these regions is especially unreliable, and many pipelines mask or flag them.

1.6.3. The HLA Region: A Case Study

The human leukocyte antigen (HLA) locus on chromosome 6p21 is among the most challenging regions in the human genome—and among the most clinically important.

HLA is difficult to call for several reasons. The HLA genes (*HLA-A*, *HLA-B*, *HLA-C*, *HLA-DRB1*, and others) are the most polymorphic coding genes in the human genome, with thousands of known alleles per gene (Robinson et al. 2020). Standard reference-based alignment struggles because reads may match the reference poorly even when they represent common, well-characterized alleles. The MHC region contains segmental duplications, copy-number variable genes (such as *HLA-DRB3/4/5*), and pseudogenes that confound read mapping. Different HLA alleles may differ by only a few nucleotides, making accurate allele-level typing difficult with short reads alone. Reads carrying non-reference HLA alleles may fail to align or align with low mapping quality, causing systematic undercalling of alternate alleles.

Despite these challenges, accurate HLA typing is essential for several clinical applications. In transplantation, HLA matching between donor and recipient is critical for organ and hematopoietic stem cell transplantation outcomes. HLA alleles are the strongest genetic risk factors for many autoimmune conditions, including type 1 diabetes, rheumatoid arthritis, and multiple sclerosis; fine-mapping causal alleles and amino acid positions requires accurate genotyping (Sakaue et al. 2023; Padyukov 2022). Specific HLA alleles—such as *HLA-B*57:01* for abacavir and *HLA-B*15:02* for carbamazepine—are pharmacogenomic markers for severe adverse drug reactions (Mallal et al. 2008; Chung et al. 2004). HLA diversity also shapes immune responses to pathogens, including HIV, hepatitis, and SARS-CoV-2.

Because standard variant callers perform poorly in HLA, specialized tools have been developed. HLA imputation methods, including those available through the Michigan Imputation Server, use dense reference panels to impute HLA alleles from array genotypes, enabling large-scale association studies (Sakaue et al. 2023). Sequence-based typing tools such as T1K perform HLA and KIR (killer immunoglobulin-like receptor) genotyping directly from WES, WGS, or RNA-seq data by aligning reads against allele databases (such as IPD-IMGT/HLA) rather than the linear reference genome (Song et al. 2022). T1K is notable for its speed, accuracy across sequencing platforms, and ability to handle both DNA and RNA data. Graph-based approaches that incorporate known HLA alleles as alternate paths can also improve alignment and variant calling in this region (Garrison et al. 2018; Liao et al. 2023).

For the purposes of this book, HLA exemplifies a broader lesson: regions of extreme diversity, structural complexity, or clinical importance may require specialized methods beyond generic variant calling pipelines. Later chapters show how variant effect models handle these challenging regions—often by excluding them entirely or applying specialized processing.

1.7. Benchmarking and Ground Truth

Evaluating variant callers requires high-confidence truth sets against which predictions can be compared. The Genome in a Bottle (GIAB) Consortium, coordinated by NIST, provides extensively characterized reference samples with validated variant calls across most of the genome (Zook et al. 2019).

1.7.1. GIAB Reference Samples

The primary GIAB samples include NA12878 (also known as HG001), a well-studied female of European ancestry from the CEPH/Utah pedigree with the longest history of characterization. The collection also includes HG002 through HG007: an Ashkenazi Jewish trio (HG002–HG004) and a Han Chinese trio (HG005–HG007), providing diversity and enabling trio-based validation.

For each sample, GIAB provides high-confidence variant calls—consensus calls derived from multiple sequencing technologies and variant callers, representing the best current estimate of true genotypes. They also define high-confidence regions, genomic intervals where the truth set is believed to be reliable; difficult regions such as segmental duplications and centromeres are excluded. Benchmarking tools like `hap.py` and RTG Tools enable standardized comparison of callsets against truth, reporting precision, recall, and F1 by variant type (Krusche et al. 2019; "RealTimeGenomics/Rtg-Core" 2025).

1.7.2. Benchmarking Metrics

Standard metrics for variant calling include recall (sensitivity), defined as TP / (TP + FN) or the fraction of true variants detected; precision (positive predictive value), defined as TP / (TP + FP) or the fraction of called variants that are true; and the F1 score, the harmonic mean of precision and recall. These are typically reported separately for SNVs and indels and may be stratified by genomic context, such as performance inside versus outside difficult regions.

1.7.3. Limitations of Current Benchmarks

GIAB truth sets have known limitations. They are derived primarily from short-read data and may miss complex variants, structural variants, and variation in difficult regions. High-confidence regions cover only approximately 85–90% of the genome, so performance in excluded regions is unknown. Sample diversity is limited, and performance may differ in underrepresented populations.

Ongoing efforts—including the T2T Consortium's complete genome assemblies and the Human Pangenome Reference Consortium's diverse haplotype collection—are expanding the scope of benchmarking resources (Liao et al. 2023).

1.8. DeepVariant: CNNs for Variant Calling

DeepVariant replaces much of the hand-engineered logic in classical pipelines with a deep convolutional neural network trained to classify candidate variants directly from read pileups (Poplin et al. 2018).

1.8.1. Image-Like Pileup Representation

Around each candidate site, DeepVariant constructs a six-channel tensor resembling an image. Each row corresponds to a read overlapping the site, with channels encoding reference match/mismatch status, Phred-scaled base quality, mapping quality, strand orientation, allele support (reference vs. alternate), and additional alignment features. The reference sequence and candidate alleles are overlaid. This representation allows the CNN to distinguish patterns consistent with true variants—balanced allele support across strands, consistent base qualities, clean alignments—from artifacts like strand-biased support or mismatches clustered at read ends.

1.8.2. Inception-Style CNN Classifier

DeepVariant uses an Inception-style CNN originally developed for image classification. Trained on high-confidence truth sets such as GIAB genomes, it learns to recognize true variant patterns and reject artifacts (strand bias, mapping pileups in repeats, inconsistent quality profiles). Once trained, the same architecture generalizes across whole-genome versus whole-exome data, PCR-free versus PCR-amplified libraries, and different instrument models and read lengths.

Crucially, DeepVariant learns to weigh quality signals jointly and end-to-end, rather than relying on post-hoc recalibration. Where VQSR fits a separate model on hand-selected annotations after calling, DeepVariant integrates the raw evidence directly into its classification—the CNN sees the same strand bias and quality patterns that VQSR would use, but learns their relationship to true variant status during training rather than in a decoupled second step.

1.8.3. Cohort Calling with DeepVariant and GLnexus

For cohort calling, DeepVariant can be combined with joint genotyping tools such as GLnexus to scale to tens or hundreds of thousands of samples while maintaining high accuracy (Yun et al. 2021). In this setup, DeepVariant produces per-sample gVCFs (genomic VCFs) containing genotype likelihoods at all sites, not just variant sites, and GLnexus merges gVCFs across samples to produce a cohort-wide callset.

Joint calling matters for several reasons. It improves sensitivity for rare variants: a variant observed in only one or two individuals may have weak per-sample evidence, but by combining likelihoods across carriers, joint calling can recover true rare variants that would be filtered in single-sample analysis. Joint calling ensures consistent representation, so that the same variants are genotyped across all samples, avoiding the problem of comparing different candidate variant sites across samples. Cohort-level quality filters can identify and remove systematic artifacts that affect subsets of samples, reducing batch effects and improving allele frequency estimates for downstream GWAS and PRS accuracy.

This combination has become a de facto standard for large WES and WGS cohorts, including recent releases of gnomAD and UK Biobank (Karczewski et al. 2020; Bycroft et al. 2018).

1.8.4. Comparison: Classical Pipelines vs. DeepVariant

Aspect	GATK HaplotypeCaller	DeepVariant
Core approach	Pair-HMM + hand-crafted heuristics	CNN on pileup tensors
Feature engineering	Expert-designed (MQ, DP, FS, etc.)	Learned end-to-end
Read independence	Assumed (violated in practice)	Implicitly models dependencies
Calibration	VQSR post-hoc recalibration	Well-calibrated likelihoods
Generalization	Requires species/platform tuning	Transfers across species and platforms
Structural variants	Limited (SNVs/indels only)	Limited (SNVs/indels only)

Both approaches achieve comparable accuracy on high-quality Illumina data, but DeepVariant shows advantages in difficult contexts and generalizes more readily to new sequencing technologies.

1.9. Significance for Genomic Deep Learning

NGS and variant calling set the stage for everything else in this book.

1.9.1. Defining the Atoms We Model

The output of WES and WGS pipelines—a VCF of SNVs, indels, and increasingly structural variants—is the raw material for nearly all downstream analyses. Polygenic scores (Chapter 3) aggregate variant effects across the genome. Rare variant burden tests collapse variants by gene or functional annotation. Variant effect predictors (Chapter 4 and later chapters) learn to score individual variants for deleteriousness or functional impact. The quality of variant calls directly limits the quality of these downstream models: false positives introduce noise, while false negatives create blind spots.

1.9.2. Constraining Downstream Models

If an assay never observes a class of variants, deep models cannot learn about them. Short-read WGS misses many structural variants and complex rearrangements. WES captures coding regions but ignores most non-coding regulatory variation. Difficult regions such as HLA and segmental duplications may be systematically excluded from training data. Models trained on these data inherit their biases and gaps. Chapter 16 returns to these issues when discussing confounders and dataset artifacts.

1.9.3. Motivating End-to-End Learning

DeepVariant is an early example of replacing a hand-designed pipeline with a learned model that operates on raw-ish data and directly optimizes accuracy. This paradigm—replacing feature engineering with learned representations—recurs throughout the book. DeepSEA and Basenji (Chapter 5) learn regulatory grammars from sequence. SpliceAI (Chapter 7) predicts splicing from local sequence context. DNA language models (Chapter 10) learn general-purpose representations from unlabeled genomes. In each case, the question is whether learned representations outperform hand-crafted features, and under what conditions.

1.9.4. Looking Ahead

The remaining chapters in Part I describe how variant calls are aggregated into genome-wide association studies (Chapter 3), which identify variant-trait associations; polygenic scores (Chapter 3), which predict complex traits from many variants; and deleteriousness scores (Chapter 4), which prioritize variants by predicted pathogenicity. These serve as baselines, inputs, and evaluation targets for the deep learning models that follow.

2. The Genomic Data Landscape

> ⚠ Warning
>
> **TODO:** - Somewhere in project, discuss correlated but distinct rare variants vs gene legthal variants vs late-onset disease variants - Multispecies genomes - ref/pangenome - Zoonomia - DMS and (… multiplexed assays) - ProteinGim; TraitGym

2.1. Why Genomic Data Resources Matter

Once we can sequence genomes and call variants, we immediately face a new problem: interpretation. No single dataset is sufficient to decide whether a variant is benign, pathogenic, or relevant to a trait. Instead, we rely on a mosaic of complementary resources: reference genomes and gene annotations that define coordinates and consequences, population variation catalogs that reveal what survives in healthy individuals, cohort and biobank datasets that link variation to phenotypes, functional genomics atlases that map biochemical activity, and clinical databases that aggregate expert interpretations.

This chapter surveys these foundational resources. Later chapters draw from them repeatedly—either directly as model inputs or indirectly as labels, benchmarks, and priors. We begin with general genomic infrastructure (references, variation catalogs, cohorts) and then turn to functional and expression resources (ENCODE, GTEx-like datasets) that provide the training labels for sequence-to-function models.

2.2. Reference Genomes and Gene Annotations

Every genomic analysis begins with a coordinate system. Reference genomes define the scaffold onto which sequencing reads are mapped, while gene annotations overlay that scaffold with biological meaning, specifying where transcripts begin and end, which regions encode protein, and how exons are spliced together. These resources are so foundational that their assumptions often become invisible: a variant's consequence, a gene's constraint score, and a model's training labels all depend on choices embedded in the reference assembly and annotation release. Understanding these dependencies is essential for interpreting results, recognizing systematic biases, and anticipating how analyses will generalize across datasets built on different genomic foundations.

2.2.1. Reference Assemblies

Most modern pipelines align reads to a small number of reference assemblies, predominantly GRCh38 or the newer T2T-CHM13 (Nurk et al. 2022). A reference genome is not simply a consensus sequence; it encodes a series of consequential decisions about how to represent duplications, alternate haplotypes, and unresolved gaps, all annotated with coordinates that downstream tools assume are stable.

The choice of reference shapes everything that follows. It determines which regions are "mappable" by short reads, how structural variants are represented, and how comparable results will be across cohorts and over time. Graph-based and pangenome references relax the assumption of a single linear reference, but the majority of datasets used in this book, and the models trained on them, are still built on GRCh37 or GRCh38 (Liao et al. 2023).

2.2.2. Gene Models

Gene annotation databases such as GENCODE and RefSeq define the exon–intron structures, canonical and alternative transcripts, start and stop codons, and untranslated regions that allow us to interpret variants in biological context (Frankish et al. 2019; O'Leary et al. 2016). These annotations are critical for distinguishing coding from non-coding variants, identifying splice-disrupting mutations, and mapping functional genomics signals to genes.

The MANE Select project provides a single matched transcript per protein-coding gene that is identical between GENCODE and RefSeq, simplifying clinical interpretation but further privileging a single isoform over biological complexity (Morales et al. 2022).

Many downstream resources, from variant effect predictors to polygenic score pipelines, implicitly assume that gene models are correct and complete. In practice, new isoforms continue to be discovered, alternative splicing remains incompletely cataloged, and cell-type-specific transcripts may be missing from bulk-derived annotations. These gaps propagate through every tool built on them.

2.3. Population Variant Catalogs and Allele Frequencies

Population variant catalogs provide the empirical foundation for distinguishing pathogenic mutations from benign polymorphisms. Allele frequency, the proportion of chromosomes in a reference population carrying a given variant, serves as a powerful prior: variants observed at appreciable frequency in healthy individuals are unlikely to cause severe early-onset disease, while ultra-rare variants demand closer scrutiny. Beyond simple filtering, allele frequencies inform statistical frameworks for case-control association, provide training signal for deleteriousness predictors, and enable imputation of ungenotyped variants through linkage disequilibrium The catalogs described below have progressively expanded in sample size, ancestral diversity, and annotation depth, transforming variant interpretation from an ad hoc exercise into a quantitative discipline.

2.3.1. dbSNP and the Variant Universe

Historically, dbSNP aggregated known single nucleotide polymorphisms and short indels into a single catalog, providing stable identifiers (rsIDs) that serve as common currency across tools and publications, basic frequency information where available, and a convenient handle for linking to other resources (Sherry et al. 2001). Modern whole-exome and whole-genome sequencing cohorts routinely discover millions of previously unseen variants, but dbSNP identifiers remain the standard way to refer to known polymorphisms.

2.3.2. 1000 Genomes and Early Reference Panels

The 1000 Genomes Project provided one of the first widely used multi-population reference panels, enabling imputation and linkage-disequilibrium-based analyses on genotyping arrays (Auton et al. 2015). Its samples continue to serve as benchmarks for variant calling performance, and its haplotype structure underlies many imputation servers and downstream analyses (Yun et al. 2021).

2.3.3. The Genome Aggregation Database (gnomAD)

The Genome Aggregation Database aggregates exome and genome data from a wide array of cohorts into harmonized allele frequency resources (Karczewski et al. 2020). gnomAD provides high-resolution allele frequencies for SNVs and indels across diverse ancestries, constraint metrics such as pLI and LOEUF that summarize a gene's intolerance to loss-of-function variation, and per-variant annotations flagging poor quality regions, low complexity, and other caveats.

These resources are indispensable for filtering common variants in Mendelian disease diagnostics, distinguishing extremely rare variants from recurrent ones, and providing population genetics priors used by variant effect predictors and deleteriousness scores like CADD (Rentzsch et al. 2019; Schubach et al. 2024). The constraint metrics, in particular, have become standard features in machine learning models that prioritize disease-relevant genes and variants.

2.4. Cohorts, Biobanks, and GWAS Summary Data

Large-scale biobanks and population cohorts have transformed human genetics from a discipline reliant on family studies and candidate gene approaches into one powered by population-level statistical inference. These resources link genomic data to electronic health records, lifestyle questionnaires, imaging, and longitudinal outcomes, enabling discovery of genetic associations across thousands of traits simultaneously. However, the composition of these cohorts carries consequences: the overrepresentation of European-ancestry individuals in most major biobanks creates systematic gaps in variant discovery, effect size estimation, and polygenic score portability that propagate through downstream analyses. These ancestry biases, and strategies for addressing them, are discussed in detail in Chapter 16.

2.4.1. Large Population Cohorts

Modern human genetics relies on large cohorts with genome-wide variation and rich phenotyping. UK Biobank, with approximately 500,000 participants and deep phenotyping, has become the dominant resource for methods development and benchmarking (Bycroft et al. 2018). FinnGen leverages Finland's population history and unified healthcare records (Kurki et al. 2023). The All of Us Research Program prioritizes diversity, aiming to enroll one million participants with deliberate oversampling of historically underrepresented groups (All of Us Research Program Investigators 2019). Additional resources include the Million Veteran Program, Mexican Biobank, BioBank Japan, China Kadoorie Biobank, and emerging African genomics initiatives such as H3Africa (Sirugo, Williams, and Tishkoff 2019). Together, these efforts enable genome-wide association studies for thousands of traits, development and evaluation of polygenic scores, and fine-mapping of causal variants and genes (Marees et al. 2018; Mountjoy et al. 2021).

While this book focuses on models rather than specific cohorts, it is important to recognize that most GWAS and polygenic score methods in Chapter 3 assume data from either array genotyping with imputation or whole-exome/whole-genome sequencing with joint calling, as in DeepVariant/GLnexus-style pipelines (Yun et al. 2021). The ascertainment, quality control, and population composition of these cohorts shape what signals can be detected and how well models generalize.

2.4.2. GWAS Summary Statistics

Beyond individual-level data, many resources distribute GWAS summary statistics: per-variant effect sizes and p-values aggregated across cohorts. The GWAS Catalog compiles published results across traits (Sollis et al. 2023), while the PGS Catalog provides curated polygenic score weights and metadata for reproducibility (Lambert et al. 2021). Frameworks like Open Targets Genetics integrate fine-mapped signals and candidate causal genes across loci (Mountjoy et al. 2021).

These summary data are the raw material for many polygenic score methods (Chapter 3) and statistical fine-mapping algorithms. They enable meta-analysis across cohorts, transfer of genetic findings to new populations, and integration with functional annotations to prioritize causal variants.

2.5. Functional Genomics and Regulatory Landscapes

The vast majority of the human genome lies outside protein-coding exons, yet this non-coding space harbors the regulatory logic that governs when, where, and how much each gene is expressed. Functional genomics assays provide the experimental means to map this regulatory landscape: identifying transcription factor binding sites, nucleosome positioning, chromatin accessibility, histone modifications, and three-dimensional genome organization across cell types and conditions. For the purposes of this book, these datasets serve a dual role. First, they supply the biological vocabulary for interpreting non-coding variants, linking sequence changes to potential regulatory consequences. Second, and more directly, they provide the training labels for sequence-to-function deep learning models. When a model learns to predict chromatin accessibility or histone marks from DNA sequence alone, it is learning a compressed representation of the regulatory code implicit in thousands of functional genomics experiments.

2.5.1. ENCODE, Roadmap, and Related Consortia

Projects like ENCODE, Roadmap Epigenomics, and Gene Expression Omnibus (GEO) are primary data generation efforts that designed coordinated experimental campaigns, selected cell types and tissues for profiling, and produced comprehensive compendia of transcription factor ChIP-seq, histone modification ChIP-seq, open chromatin assays (DNase-seq, ATAC-seq), and chromatin conformation data (Hi-C and related methods) (Kagda et al. 2025; Kundaje et al. 2015; Edgar, Domrachev, and Lash 2002). These datasets map regulatory elements, chromatin states, and higher-order genome structure with tight experimental control and uniform processing pipelines.

The significance of these consortia for this book is less about any individual experiment than about the scale and standardization they provide. By generating hundreds of assays across dozens of cell types with consistent protocols, ENCODE and Roadmap created canonical reference datasets that define the regulatory landscape for the cell types they profiled.

2.5.2. The Cistrome Data Browser

While ENCODE and Roadmap produced authoritative datasets for their chosen cell types and factors, they represent only a fraction of publicly available functional genomics experiments. The Cistrome Data Browser addresses this gap by aggregating thousands of human and mouse ChIP-seq and chromatin accessibility datasets from ENCODE, Roadmap, GEO, and individual publications into a reprocessed, searchable repository (R. Zheng et al. 2019). All datasets pass through a uniform quality control and processing pipeline, enabling comparisons across experiments that were originally generated by different labs with different protocols.

Cistrome provides uniform peak calls and signal tracks, metadata for cell type, factor, and experimental conditions, and tools for motif analysis and regulatory element annotation. The tradeoff is heterogeneity: while the reprocessing harmonizes computational steps, the underlying experiments vary in sample preparation, sequencing depth, and experimental design. Cistrome thus expands coverage at the cost of the tight experimental control found in the primary consortia.

2.5.3. From Assays to Training Labels

Sequence-to-function models transform these functional genomics resources into supervised learning problems. Models like DeepSEA (see Chapter 5) draw training labels from ENCODE, Roadmap, and Cistrome-style datasets collectively: each genomic window is associated with binary or quantitative signals indicating transcription factor binding, histone modifications, or chromatin accessibility across many assays and cell types (J. Zhou and Troyanskaya 2015; J. Zhou et al. 2018).

The quality, coverage, and biases of these labels directly constrain what models can learn. Cell types absent from the training compendium cannot be predicted reliably. Factors with few high-quality ChIP-seq experiments will have noisier labels. And systematic differences between assay types (peak-based binary labels versus quantitative signal tracks) shape whether models learn to predict occupancy, accessibility, or something in between. These considerations become central when we examine model architectures and training strategies in Chapter 5.

2.6. Expression and eQTL Resources

Expression datasets link sequence variation to transcriptional consequences, providing a bridge between regulatory elements and gene-level effects. While functional genomics assays reveal where transcription factors bind and which chromatin regions are accessible, expression data answer the downstream question: does this regulatory activity actually change how much RNA a gene produces? Expression quantitative trait loci (eQTLs) formalize this relationship statistically, identifying genetic variants associated with changes in transcript abundance. For variant interpretation and genomic prediction, eQTLs offer mechanistic hypotheses connecting non-coding variants to specific genes and tissues. For model training, expression data provide quantitative labels that integrate across the many regulatory inputs converging on a single promoter. The resources below range from population-scale bulk tissue atlases to emerging single-cell datasets that resolve expression variation at cellular resolution.

2.6.1. Bulk Expression Atlases

Projects like the Genotype-Tissue Expression (GTEx) consortium provide RNA-seq expression profiles across dozens of tissues, eQTL maps linking variants to gene expression changes in cis, and splicing QTLs and other molecular QTLs (The GTEx Consortium 2020). With matched genotypes and expression data from nearly 1,000 post-mortem donors across 54 tissues, GTEx established foundational insights: most genes harbor tissue-specific eQTLs, regulatory variants typically act in cis over distances of hundreds of kilobases, and expression variation explains a meaningful fraction of complex trait heritability.

Even when not explicitly cited, GTEx-like resources underpin expression prediction models such as PrediXcan and TWAS frameworks, colocalization analyses that ask whether a GWAS signal and an eQTL share a causal variant, and expression-based prioritization of candidate genes at trait-associated loci (Gamazon et al. 2015; Gusev et al. 2016). The GTEx design has limitations: post-mortem collection introduces agonal stress artifacts, sample sizes per tissue vary considerably, and some disease-relevant tissues (such as pancreatic islets or specific brain regions) remain undersampled. Complementary resources like the eQTLGen Consortium aggregate eQTL results from blood across larger sample sizes, trading tissue diversity for statistical power (Võsa et al. 2021).

2.6.2. Single-Cell and Context-Specific Expression

Bulk RNA-seq averages expression across all cells in a tissue sample, obscuring the cell-type-specific programs that often mediate disease biology. Single-cell RNA-seq resolves this heterogeneity, identifying expression signatures for individual cell types, rare populations, and transitional states. Large-scale efforts like the Human Cell Atlas, Tabula Sapiens, and disease-focused single-cell consortia are building reference atlases that catalog cell types across organs and developmental stages (Regev et al. 2017; The Tabula Sapiens Consortium 2022).

For variant interpretation, single-cell data enable cell-type-specific eQTL mapping, revealing that a variant may influence expression in one cell type but not others within the same tissue. Spatial transcriptomics adds anatomical context, preserving tissue architecture while measuring gene expression. These technologies introduce computational challenges: sparsity from dropout, batch effects across samples and technologies, and the sheer scale of datasets with millions of cells.

In this book, single-cell and spatial resources appear primarily in later chapters on multi-omics integration and systems-level models, but they represent the direction toward which expression genetics is moving, promising to connect genetic variation to cellular phenotypes with unprecedented resolution.

2.7. Variant Interpretation Databases and Clinical Labels

Allele frequencies tell us what variants are tolerated in healthy populations, and functional genomics data reveal where the genome is biochemically active, but neither directly answers the clinical question: is this variant pathogenic? That determination requires integrating multiple lines of evidence, including family segregation, functional assays, computational predictions, and phenotypic observations, into a structured framework that can be applied consistently across variants, genes, and diseases. Clinical variant interpretation databases aggregate these assessments from laboratories, expert panels, and research groups, providing labels that inform diagnostic decisions, guide research, and serve as training data for machine learning models. These databases have become critical infrastructure for both clinical genomics and computational method development, though their labels carry biases and circularity that propagate through any analysis built on them.

2.7.1. ClinVar and Related Resources

ClinVar aggregates assertions of variant pathogenicity from clinical laboratories and researchers, with supporting evidence and conflicting interpretations where relevant (Landrum et al. 2018). Its labels are critical for diagnostic pipelines, benchmarking variant effect predictors, and training machine learning models in clinical genomics.

However, ClinVar's labels are not collected in isolation. As discussed in Chapter 4, clinical submissions increasingly incorporate computational scores like CADD as one piece of evidence, which creates subtle circularity when those same labels are used to evaluate or train computational predictors (Schubach et al. 2024). This circularity is a recurring methodological concern throughout the book.

2.7.2. ClinGen and Expert Curation

The Clinical Genome Resource (ClinGen) complements ClinVar by providing expert-curated assessments at multiple levels of granularity (Rehm et al. 2015). ClinGen expert panels evaluate gene-disease validity, asking whether variation in a particular gene can cause a specific disease, and dosage sensitivity, determining whether haploinsufficiency or triplosensitivity leads to clinical phenotypes. These evaluations build on the catalog of Mendelian phenotypes maintained by OMIM (Online Mendelian Inheritance in Man), which provides curated gene-disease associations, clinical synopses, and literature summaries that have long served as the reference for clinical genetics (Amberger et al. 2015).

For individual variants, ClinGen Variant Curation Expert Panels apply the ACMG/AMP criteria systematically, and the FDA has recognized these curations as a valid source of scientific evidence for clinical validity (Pejaver et al. 2022). ClinGen also develops calibrated thresholds for computational

predictors like CADD and REVEL, specifying score intervals that justify different strengths of evidence for pathogenicity or benignity. These calibrated thresholds directly inform how computational scores should be incorporated into variant classification workflows.

2.7.3. ClinPGx and Pharmacogenomics Resources

ClinPGx integrates the PharmGKB knowledge base, CPIC clinical guidelines, and PharmCAT annotation tool into a unified pharmacogenomics resource (Whirl-Carrillo et al. 2012). While most variant interpretation databases focus on disease-causing mutations, ClinPGx curates gene-drug associations that influence drug metabolism, efficacy, and adverse reactions. These pharmacogenomic variants are often common polymorphisms rather than rare pathogenic mutations, but their clinical importance for prescribing decisions makes them a distinct category of actionable genetic variation. The CPIC guidelines provide evidence-based recommendations for adjusting drug selection or dosing based on pharmacogene diplotypes, and ClinPGx-annotated FDA drug labels document the regulatory status of these associations.

2.8. How Later Chapters Use These Resources

The genomic deep learning models that follow inherit both the strengths and limitations of the data they are trained on. Chapter 3 draws on GWAS summary statistics and biobank-scale cohorts to construct polygenic scores. Chapter 4 examines how annotation-based methods compress population frequencies, conservation, and functional signals into genome-wide deleteriousness scores. Chapters 5-7 use ENCODE, Roadmap, and Cistrome-style functional data as training labels for sequence-to-function models, while Chapters 12-14 revisit these resources as inputs, labels, and priors for genomic foundation models.

By surveying the data landscape in one place, we establish a common reference that later chapters can build on rather than re-introducing each resource from scratch. The recurring theme is that biases, gaps, and circularity in these foundational datasets propagate through every model trained on them. A variant effect predictor trained on ClinVar labels inherits the ascertainment biases of clinical sequencing; a chromatin model trained on ENCODE cell lines may not generalize to primary tissues. Understanding these foundations is essential for interpreting what models learn and anticipating where they will fail.

3. GWAS & Polygenic Scores

> ⚠ Warning
>
> **TODO:**
>
> - review by Dr. Schaid's team
> - add manhattan plot and other visuals
> - "pugitive" variants

3.1. The GWAS Paradigm

Genome-wide association studies represent the dominant paradigm for mapping genetic contributions to complex traits. The core idea is conceptually simple: test each of millions of genetic variants for statistical association with a phenotype of interest, then identify variants or genomic regions where association signals exceed stringent thresholds for multiple testing. This brute-force approach, made feasible by advances in genotyping technology and the assembly of large cohorts, has catalogued thousands of trait-associated loci across the human genome. Yet GWAS is fundamentally a statistical exercise in association, not a direct window into biological mechanism. Understanding both its power and its limitations is essential groundwork for the mechanistic models we develop in later parts of this book.

A GWAS requires three ingredients: a large sample of genotyped or sequenced individuals, a well-defined phenotype (either binary, such as disease status, or quantitative, such as height or lipid levels), and a statistical model that relates genotype to phenotype while adjusting for confounders. In practice, the confounders typically include age, sex, and principal components of genetic ancestry that capture population structure (Marees et al. 2018). Alternative strategies for controlling confounding, including case-control matching on ancestry and other covariates, are discussed in Chapter 16.

The output of a GWAS is a set of associated variants and loci, not a direct map from variant to mechanism. Variants that pass the significance threshold are sometimes called "hits," but this terminology obscures an important ambiguity: the variant with the smallest p-value at a locus is not necessarily the variant that causes the phenotypic effect. It may simply be a statistical proxy for the true causal variant, a distinction that becomes central in later sections.

3.1.1. Continuous Phenotypes

For quantitative traits such as height, body mass index, or lipid levels, GWAS employs linear regression. At a single bi-allelic variant j, the standard model takes the form

$$y_i = \alpha + \beta_j g_{ij} + \gamma^\top c_i + \varepsilon_i,$$

where:

- y_i is the phenotype for individual i.
- α is the intercept representing the baseline phenotype when genotype dosage and covariates are zero.
- β_j is the per-allele effect size we wish to estimate.
- g_{ij} is the genotype dosage at variant j, coded as 0, 1, or 2 copies of the alternative allele, or as an imputed fractional dosage.
- γ is a vector of coefficients for the covariates.
- c_i is a vector of covariates.
- ε_i is the residual error term.

3.1.1.1. Effect Size

The coefficient β_j has a direct interpretation: it is the expected change in phenotype per additional copy of the alternative allele, holding covariates constant. A variant with $\widehat{\beta}_j = 0.05$ for height (measured in centimeters) would be associated with an average increase of 0.05 cm per copy of the effect allele. These effect sizes are typically small, often explaining far less than 1% of phenotypic variance individually.

3.1.1.1.1. Variance Explained and Polygenicity

The variance explained by a single variant depends on both its effect size and its allele frequency. For an additive model, the contribution to phenotypic variance is approximately $2p(1-p)\beta_j^2$, where p is the allele frequency. A variant with modest effect size but intermediate frequency will explain more variance than one with the same effect size but low frequency. Conversely, rare variants can harbor larger effect sizes while still explaining little population-level variance.

The distribution of effect sizes across the genome follows a characteristic pattern: most variants have negligible effects, a modest number have small but detectable effects, and very few have effects large enough to be individually meaningful. This "polygenicity" means that for most complex traits, thousands of variants contribute to heritability, each with a tiny increment. The degree of polygenicity varies by trait: height is highly polygenic, while some autoimmune diseases show more concentrated genetic architecture.

3.1.1.1.2. Winner's Curse

Effect sizes estimated in discovery GWAS tend to be inflated relative to their true values, a phenomenon known as winner's curse. Variants cross the significance threshold partly because sampling noise pushed their estimated effects upward; re-estimation in independent samples typically yields smaller effects. This inflation is most severe for variants near the significance threshold and motivates the use of independent replication cohorts.

3.1.1.1.3. Standardized Effect Sizes

GWAS results are often reported as standardized effect sizes rather than raw coefficients. When both the phenotype and genotype dosage are standardized to unit variance, β_j represents the correlation between genotype and phenotype, and its square approximates the proportion of variance explained. Standardization facilitates comparison across traits measured in different units but obscures the clinically interpretable magnitude of effects. Summary statistics from large consortia typically include both raw and standardized effect sizes, or provide sufficient information to convert between them.

3.1.1.2. Significance Testing and Multiple Comparisons

The test statistic $z_j = \hat{\beta}_j / \mathrm{SE}(\hat{\beta}_j)$ follows approximately a standard normal distribution under the null hypothesis of no association. The corresponding p-value quantifies how unlikely the observed association would be if the variant had no true effect on the phenotype.

3.1.1.2.1. Genome-Wide Significance

Testing millions of variants simultaneously creates a severe multiple comparisons problem. At a nominal significance level of $\alpha = 0.05$, we would expect roughly 50,000 false positives among one million independent tests. The field has converged on a genome-wide significance threshold of $p < 5 \times 10^{-8}$, derived from a Bonferroni correction for approximately one million independent common variants in the human genome after accounting for linkage disequilibrium (**pe?**'er_estimation_2008). This threshold is stringent by design: it controls the family-wise error rate, ensuring that across all tests, the probability of even one false positive remains below 0.05.

The Bonferroni correction is conservative when tests are correlated, as they are in GWAS due to LD. Alternative approaches, such as controlling the false discovery rate (FDR), permit more discoveries at the cost of accepting a known proportion of false positives among significant results. In practice, the 5×10^{-8} threshold has proven robust and remains the convention for declaring genome-wide significant associations, though suggestive thresholds (often $p < 10^{-5}$) are sometimes used to flag loci for replication.

3.1.1.2.2. Significance Versus Magnitude

The distinction between statistical significance and effect magnitude deserves emphasis. In sufficiently large samples, even tiny effects become highly significant. A variant explaining 0.01% of phenotypic variance might achieve $p < 10^{-50}$ in a million-person study. Significance tells us whether an association is likely real; effect size tells us whether it matters. For polygenic score construction and biological interpretation, effect sizes are the quantities that carry scientific meaning.

3.1.1.3. Covariates

The covariate vector c_i typically includes age, sex, and principal components of genetic ancestry. The ancestry components are essential for avoiding confounding due to population structure: if allele frequencies and phenotype means both vary across populations, a naive analysis will find spurious associations at variants that simply track ancestry rather than causally influencing the

trait. Additional covariates such as genotyping batch, recruitment site, or technical factors may be included depending on the study design.

The covariate coefficients γ have an analogous interpretation: each element represents the expected change in phenotype per unit change in the corresponding covariate, holding genotype and other covariates constant. For example, a coefficient of 0.5 for age would indicate that each additional year of age is associated with a 0.5-unit increase in the phenotype. Unlike β_j, the covariate coefficients are nuisance parameters in GWAS; they are included to avoid confounding but are not the quantities of scientific interest.

3.1.1.4. Ancestry PCs

Among the covariates, principal components of genetic ancestry deserve particular explanation because they address a confounding problem specific to genetic association studies: population stratification. Human populations differ in allele frequencies due to demographic history, and these frequency differences can correlate with phenotypic differences driven by environmental or cultural factors. If a study population includes individuals from multiple ancestral backgrounds, and if ancestry correlates with the phenotype for non-genetic reasons, variants that simply differ in frequency between populations can produce spurious associations.

Principal component analysis of the genotype matrix provides a standard solution (Patterson, Price, and Reich 2006; **price_pca_2006?**). The first few principal components of genome-wide genotype data capture axes of genetic variation that correspond primarily to continental ancestry and finer-scale population structure. Including these PCs as covariates in the regression model adjusts for ancestry-associated confounding: the GWAS tests whether a variant is associated with phenotype beyond what would be expected from shared ancestry. In practice, most studies include between 10 and 20 ancestry PCs, though the optimal number depends on the population structure present in the cohort. This approach does not eliminate all confounding, particularly from cryptic relatedness or fine-scale structure within ancestry groups, but it handles the dominant sources of stratification in typical GWAS designs. The broader implications of ancestry for model development and evaluation are discussed in Chapter 16.

3.1.1.5. Intercept and Residuals

In the linear model, α represents the expected phenotype value when the genotype dosage is zero (homozygous reference) and all covariates are also zero. This baseline is often not directly interpretable in practice, since "zero" may not be a meaningful value for covariates like age or ancestry principal components. If covariates are mean-centered before fitting, then α represents the expected phenotype for a reference-homozygous individual at average covariate values, which is somewhat more intuitive. In practice, the intercept is a nuisance parameter in GWAS. The scientific focus is on β_j, the per-allele effect size, not the baseline. The intercept anchors the model but is rarely reported or interpreted in GWAS summary statistics.

The residual term ε_i absorbs everything not captured by the modeled genotype and covariates. This includes measurement noise in the phenotype, environmental influences, gene-by-environment interactions, epistatic effects among variants, and the aggregate contribution of all other genetic variants not being tested at this particular locus. In this sense, ε_i reflects both the stochastic nature of complex phenotypes and the heritability gap: even a variant with a true causal effect explains

only a small fraction of phenotypic variance, leaving most variation in the residual. For highly polygenic traits, the per-variant R^2 is typically tiny, and the residual dominates.

3.1.2. Binary Phenotypes

For disease outcomes and other case-control phenotypes, linear regression is replaced by logistic regression. The phenotype y_i is now binary (1 for cases, 0 for controls), and we model the log-odds of disease:

$$\log \frac{P(y_i = 1)}{P(y_i = 0)} = \alpha + \beta_j g_{ij} + \gamma^\top c_i.$$

The left-hand side is the logit of the probability of being a case. The coefficient β_j now represents the change in log-odds per additional copy of the alternative allele, rather than a change in a continuous phenotype.

Exponentiating β_j yields the odds ratio (OR), which is the quantity most commonly reported for binary traits. An odds ratio of 1.2 means that each copy of the effect allele multiplies the odds of disease by 1.2, or equivalently increases the odds by 20%. Most common variants identified by GWAS have odds ratios between 1.05 and 1.5, reflecting modest individual effects that accumulate across many loci to influence disease risk.

The odds ratio is not the same as relative risk, though the two are often conflated. For rare diseases (prevalence below roughly 10%), the odds ratio approximates the relative risk, but for common outcomes the distinction matters. An odds ratio of 2.0 does not mean the risk is doubled; it means the odds are doubled, and converting to absolute risk requires knowledge of baseline disease prevalence.

Logistic regression shares the same covariate structure as linear regression, and the same concerns about population stratification apply. The residual error term disappears from the explicit model formulation because the outcome is binary, but the conceptual issue remains: most of the variation in disease liability is not captured by any single variant, and the per-variant contribution to overall risk discrimination is small.

3.2. Linkage Disequilibrium and Association Signals

A GWAS result is not a direct readout of which variants cause a phenotype. The genome is not a collection of independent loci; nearby variants are correlated because they are inherited together on haplotypes that have not been broken up by recombination. This correlation structure, known as linkage disequilibrium, means that when a GWAS identifies an association signal at a particular variant, the true causal variant may lie anywhere within the surrounding correlated region. Distinguishing causal variants from their correlated neighbors is one of the central challenges in human genetics, and it has direct implications for how we interpret polygenic scores and, later, how we train and evaluate sequence-based models.

3.2.1. Haplotype Structure and Recombination

The correlation between nearby variants arises from the mechanics of meiotic recombination. When chromosomes pair during meiosis, they exchange segments at crossover points, but these crossovers are relatively rare: on average, only one or two per chromosome arm per generation. Variants that are close together on a chromosome have a low probability of being separated by a crossover event, so they tend to be inherited together on the same haplotype across many generations. Variants that are farther apart, or on different chromosomes, are more likely to be shuffled independently. The result is a genome organized into regions of high correlation (sometimes called LD blocks) separated by recombination hotspots where correlation decays rapidly.

3.2.2. Measuring Correlation: The r^2 Statistic

The standard measure of linkage disequilibrium between two biallelic variants is r^2, the squared Pearson correlation between their genotype dosages in a population sample. Values of r^2 near 1.0 indicate that the two variants are nearly perfect proxies for each other: knowing the genotype at one variant almost completely determines the genotype at the other. Values near zero indicate that the variants segregate independently. In practice, r^2 decays with physical distance, but the rate of decay varies substantially across the genome depending on local recombination rates. Some regions harbor extended haplotypes where dozens or even hundreds of variants remain tightly correlated across tens or hundreds of kilobases; others show rapid LD decay within a few kilobases.

3.2.3. Causal Versus Tag Variants

This correlation structure has a critical implication for GWAS interpretation. When we observe a significant association between a variant and a phenotype, we cannot immediately conclude that this variant is responsible for the phenotypic effect. The association may instead reflect the variant's correlation with a true causal variant nearby. The tested variant is acting as a statistical proxy, or tag, for the underlying causal signal. In many GWAS loci, the variant with the smallest p-value is not the true causal variant; it is simply the most strongly associated tag in that LD block, often because it was genotyped or imputed with higher quality, or because its allele frequency happened to provide more statistical power.

This ambiguity motivates a terminological distinction that will recur throughout this book. A causal variant is one where changing the allele would, in the relevant biological context, change the phenotype. It exerts a direct mechanistic effect on some molecular process that ultimately influences the trait. A tag variant (or proxy variant) is statistically associated with the phenotype only because it is correlated with one or more causal variants through linkage disequilibrium. If we could somehow break the LD by examining a population with different haplotype structure, the tag variant would lose its association while the causal variant would retain it.

In practice, we rarely know with certainty which variants are causal. We therefore adopt two working categories. A putative causal variant is one with strong statistical evidence (such as a high posterior probability from fine-mapping) combined with supportive functional data (such as overlap with regulatory elements or experimental validation). A purely associative variant is one that achieves statistical significance in GWAS but where the weight of evidence suggests it is tagging underlying causal variation rather than contributing mechanistically. The boundary between these

categories is not sharp, and many variants occupy an ambiguous middle ground. Polygenic scores, as typically constructed, do not distinguish between these categories at all: they assign weights based on statistical association, regardless of whether the variants are causal or purely associative. This limitation becomes important when we consider how scores transfer

3.3. From Association Signals to Fine-Mapping

Given that GWAS associations typically implicate broad genomic regions rather than individual causal variants, a natural follow-up question is: which variant (or variants) within an associated locus is actually responsible for the phenotypic effect? Statistical fine-mapping attempts to answer this question by modeling the joint contribution of all variants in a region while accounting for their correlation structure. The output is not a single answer but a probability distribution over candidate variants, allowing us to quantify our uncertainty about which variants are causal. This probabilistic framing has important downstream consequences: it influences how we construct polygenic scores, how we prioritize variants for experimental follow-up, and how we evaluate whether deep learning models have learned biologically meaningful signals.

3.3.1. Bayesian Fine-Mapping Framework

The conceptual shift from marginal to joint modeling lies at the heart of fine-mapping. Standard GWAS tests each variant independently, asking whether that variant's genotype is associated with the phenotype after adjusting for covariates. This marginal approach ignores the fact that multiple variants in the same region share information through LD. If two variants are highly correlated, they will both show significant associations even if only one of them (or neither, if both are tagging a third variant) is truly causal. Fine-mapping methods instead fit models that consider all variants in a region simultaneously, asking which subset of variants best explains the observed association signal given the correlation structure among them.

Most modern fine-mapping approaches adopt a Bayesian framework. For each variant in the region, the method estimates a posterior inclusion probability (PIP), which represents the probability that the variant is causal given the observed data and the assumed model. A variant with PIP of 0.95 has strong statistical evidence of causality; a variant with PIP of 0.05 is unlikely to be causal and is probably tagging nearby causal variation. These probabilities are not guarantees, and they depend on modeling assumptions that may not hold perfectly in practice, but they provide a principled quantification of uncertainty that point estimates from GWAS cannot offer.

A related concept is the credible set, which is a minimal set of variants that together contain the causal variant (or variants) with high probability. A 95% credible set, for example, is constructed by ranking variants by their PIPs and including variants until their cumulative probability exceeds 0.95. In favorable cases where LD is limited and one variant stands out clearly, a credible set may contain only one or a few variants. In regions of extensive LD where many variants have similar statistical support, credible sets may contain dozens of candidates, reflecting genuine uncertainty about which is causal.

Fine-mapping methods differ in their underlying assumptions. Some assume a single causal variant per locus, which simplifies computation but may be unrealistic for complex loci harboring multiple independent signals. Others allow for multiple causal variants, at the cost of increased computational

complexity and the need for additional regularization or prior assumptions. Methods also differ in their prior distributions on effect sizes: some assume that causal effect sizes follow a normal distribution, while others use spike-and-slab priors that place most probability mass on zero (reflecting the expectation that most variants are not causal) with a diffuse component for the minority that are. Finally, methods differ in their data requirements. Some operate directly on individual-level genotype and phenotype data, which provides the most information but requires access to protected datasets. Others operate on GWAS summary statistics combined with LD estimates from a reference panel, sacrificing some precision for the practical advantage of working with publicly available data (Pasaniuc and Price 2016).

3.3.2. Applications and Multi-Ancestry Leverage

The outputs of fine-mapping feed into multiple downstream applications. Variants with high PIPs become candidates for experimental follow-up, whether through CRISPR perturbation, reporter assays, or other functional studies. Credible sets define the search space for identifying causal genes, often through integration with gene expression data or chromatin annotations. And PIP estimates can inform polygenic score construction: rather than weighting variants purely by their GWAS effect sizes, one can upweight variants with high posterior probability of causality and downweight those that appear to be tagging nearby causal variation. This reweighting does not dramatically improve predictive accuracy in most cases, but it can improve interpretability and, importantly, improve transferability across populations with different LD structures.

Multi-ancestry data provides particular leverage for fine-mapping. Because LD patterns differ across populations (reflecting distinct demographic histories and recombination landscapes), a variant that is tightly linked to a causal variant in one population may be less correlated in another. When the same association signal appears across ancestries but the pattern of correlated variants differs, fine-mapping algorithms can triangulate more precisely on the likely causal variant. Large resources such as Open Targets Genetics integrate association signals, LD information, fine-mapping results, and functional annotations across multiple ancestries to prioritize likely causal variants and their target genes for thousands of traits (Mountjoy et al. 2021).

3.3.3. Appropriate Expectations

It is important to maintain appropriate expectations about what fine-mapping can and cannot achieve. Statistical fine-mapping narrows the search space and quantifies uncertainty, but it does not definitively identify causal variants. Even a variant with PIP above 0.9 may not be causal if the model assumptions are violated or if the true causal variant was not included in the analysis (for example, because it is a rare variant not well captured by common variant arrays). Biological validation remains essential. What fine-mapping provides is a principled transition from the statement that something in this region is associated with the trait to the more refined statement that these few variants are the most plausible causal candidates, given current data and models.

3.4. Constructing Polygenic Scores

With GWAS summary statistics in hand, the next step for many applications is to aggregate genetic effects across the genome into a single number that summarizes an individual's genetic predisposition

to a trait. This aggregation, known as a polygenic score, treats the genome as a linear sum of variant effects, weighting each variant by its estimated contribution from GWAS. The simplicity of this formulation is both its strength and its limitation: it enables straightforward computation and interpretation, but it also encodes assumptions about additivity and ignores the distinction between truly causal variants and those that are merely correlated with causal variants. Several methodological traditions have emerged for constructing polygenic scores, ranging from simple heuristics to sophisticated Bayesian models that explicitly account for linkage disequilibrium.

i Terminology: PGS vs PRS

Before diving into the mechanics of genome-wide association studies and polygenic prediction, it is worth clarifying the terminology that pervades this literature. The field has accumulated several near-synonyms over the past two decades, reflecting both the evolution of methods and the expanding scope of applications from disease risk to quantitative traits. Establishing a consistent vocabulary here will prevent confusion in later chapters, where we build on these concepts to connect classical statistical genetics with deep learning approaches.
The literature uses several related terms:

- **Polygenic risk score (PRS)** – historically common, especially for disease endpoints
- **Polygenic score (PGS)** – more general, used for both disease risk and quantitative traits

In this book we use **polygenic score (PGS)** as the primary term, because many of the same methods are used for quantitative traits (e.g., height, LDL cholesterol), disease incidence (e.g., coronary artery disease), and intermediate molecular traits. When we cite work that uses "PRS," we treat PRS and PGS as synonyms unless the distinction matters.
Throughout, we will:

- Use **PGS** for the generic concept.
- Use **PRS** only when we are quoting or closely paraphrasing papers that do the same.

The mathematical form of a polygenic score is deceptively simple. For an individual i, the score is computed as

$$\mathrm{PGS}_i = \sum_{j=1}^{M} w_j \, g_{ij},$$

where g_{ij} is the genotype dosage for individual i at variant j (typically coded as 0, 1, or 2 copies of the effect allele, or as a fractional imputed dosage), w_j is a weight representing the estimated per-allele effect of variant j, and the sum runs over M variants included in the score. The weight w_j is usually derived from GWAS effect size estimates, though the precise derivation varies across methods. This formulation embodies a linear additive model: each variant contributes independently and additively to the score, with no interactions between variants and no nonlinear transformations of genotype.

The challenge in constructing a useful polygenic score lies in choosing which variants to include

and how to set their weights. Raw GWAS effect size estimates are noisy, particularly for variants that barely reach significance or for variants in regions of extensive LD where the signal is spread across many correlated markers. Simply summing all genome-wide variants weighted by their marginal effect estimates would produce a score dominated by noise. The various methods for PGS construction can be understood as different strategies for filtering variants, shrinking effect estimates, and accounting for correlation structure.

3.4.1. Clumping and Thresholding

The simplest approach, known as clumping and thresholding (often abbreviated C+T), applies two sequential filters to the GWAS results (Choi, Mak, and O'Reilly 2020). First, variants are filtered by p-value: only those exceeding some significance threshold are retained. This threshold might be the conventional genome-wide significance level of 5×10^{-8}, or it might be more permissive (such as 10^{-4}, 10^{-2}, or even 1.0 to include all variants) depending on the application and the polygenicity of the trait. Second, within each genomic region, variants are "clumped" by LD: the variant with the smallest p-value is retained as the index variant, and all other variants within a specified window that exceed an r^2 threshold (commonly 0.1 or 0.2) are removed. The surviving variants are then weighted by their GWAS effect size estimates, and the score is computed as the weighted sum.

Clumping and thresholding has the virtues of simplicity and computational efficiency. It can be implemented using only summary statistics and a reference panel for LD estimation, without access to individual-level data. It produces sparse scores with interpretable variant sets. And it remains competitive with more sophisticated methods for some traits, particularly those with large-effect variants that are well captured by stringent significance thresholds.

The limitations of C+T stem from its heuristic nature. By retaining only one variant per LD block, it discards information: if multiple variants in a region independently contribute to the trait (allelic heterogeneity), or if the true causal variant was not the one with the smallest p-value, the score will be suboptimal. The method treats all retained variants as equally reliable, making no distinction between a variant that clearly stands alone and one that narrowly beat several nearly equivalent neighbors. And the performance depends sensitively on the choice of p-value threshold and LD parameters, which are typically tuned by grid search in a validation dataset, introducing potential overfitting and limiting generalizability.

3.4.2. LD-Aware Bayesian Methods

A more principled approach models the joint distribution of effect sizes across all variants while explicitly accounting for their correlation structure. The family of LD-aware Bayesian methods, exemplified by LDpred (Vilhjálmsson et al. 2015), PRS-CS, SBayesR, and lassosum, shares a common conceptual framework: treat the true effect sizes as random variables drawn from some prior distribution, observe the noisy GWAS estimates, and compute posterior effect size estimates that optimally combine prior beliefs with observed data given the LD structure.

LDpred, for example, assumes that a fraction p of variants have nonzero effects drawn from a normal distribution, while the remaining $1 - p$ have exactly zero effect. Given GWAS summary statistics and an LD reference panel, the method computes posterior mean effect sizes by solving a system of equations that propagates information across correlated variants. Variants with strong marginal associations that are uncorrelated with other strong signals receive weights close to their GWAS

estimates; variants whose associations can be explained by LD with nearby signals are shrunk toward zero. The polygenicity parameter p can be estimated from the data or specified based on prior knowledge about the trait.

Other methods in this family make different modeling choices. PRS-CS uses a continuous shrinkage prior that adapts to local genetic architecture. SBayesR employs a mixture of normal distributions with different variances, allowing for a spectrum of effect sizes from large to small. Lassosum applies L1 penalization, which induces sparsity and can be computed efficiently. Despite these differences, all of these methods share the goal of producing effect size estimates that account for LD, shrink noisy estimates appropriately, and can leverage genome-wide information rather than treating each locus independently.

Compared to clumping and thresholding, LD-aware Bayesian methods generally achieve modestly higher predictive accuracy, particularly for highly polygenic traits where thousands of variants contribute small effects. They allow multiple correlated variants to share signal rather than forcing a winner-take-all selection. And they provide a coherent probabilistic framework that can, in principle, be extended to incorporate additional information such as functional annotations or multi-ancestry data. The cost is increased computational complexity and the need for careful specification of priors, LD reference panels, and other modeling choices.

3.4.3. Fine-Mapping-Informed Polygenic Scores

The methods described above derive weights from GWAS association statistics, which reflect a mixture of causal effects and LD-induced correlations. An alternative strategy incorporates fine-mapping results to emphasize variants that are more likely to be causal. If fine-mapping has produced posterior inclusion probabilities for variants across the genome, these probabilities can be integrated into PGS construction in several ways.

One approach filters variants by PIP, including only those above some threshold (such as 0.1 or 0.5) in the score. This produces a sparse score concentrated on high-confidence causal candidates. A related approach weights variants by their PIP, setting $w_j \propto \text{PIP}_j \times \hat{\beta}_j$ so that variants with low probability of causality contribute less even if their marginal associations are strong. Yet another approach operates at the level of credible sets: for each fine-mapped locus, select one representative variant (typically the one with highest PIP) or include all variants in the credible set with PIP-proportional weights.

These fine-mapping-informed strategies aim to shift the score away from purely associative variants toward putative causal variants. The practical benefits for prediction accuracy are often modest, since even tag variants carry predictive information as long as LD patterns are consistent between training and test populations. The more compelling advantages are interpretability and transferability. A score built from likely causal variants is easier to connect to biological mechanisms and target genes. And because causal variants should have consistent effects across populations (unlike tag variants, whose correlations with causal variants may differ), fine-mapping-informed scores may transfer better across ancestries, though this remains an active area of investigation.

3.5. Interpreting Polygenic Scores

Once a polygenic score has been computed for an individual, the question becomes: what does it mean? A raw score, expressed as a sum of weighted genotypes, has no inherent clinical or biological interpretation. Converting this number into something actionable, whether a relative risk compared to the population or an absolute probability of disease, requires additional modeling and calibration. Moreover, the interpretation of a polygenic score depends critically on the population in which it was derived and the population in which it is applied. Scores developed in one ancestry group often perform poorly in others, raising both scientific and ethical questions about equitable deployment.

3.5.1. Relative versus Absolute Risk

Polygenic scores are most naturally interpreted in relative terms. The raw score itself is an arbitrary number whose magnitude depends on how many variants are included, how effects are scaled, and various normalization choices. What matters is where an individual falls within the distribution of scores in a reference population. A person in the 95th percentile has a higher genetic predisposition than 95% of the reference population; a person in the 10th percentile has lower predisposition than 90% of the population. This percentile framing, or equivalently the number of standard deviations from the population mean, provides a natural way to communicate relative genetic risk.

The clinical literature often emphasizes tail comparisons: individuals in the top 1% or 5% of the PGS distribution compared to those in the middle or bottom of the distribution. For many common diseases, those in the top few percentiles have odds ratios of 2 to 5 (or occasionally higher) compared to population average, meaning their odds of disease are several-fold elevated. These tail effects can be clinically meaningful, particularly for diseases where early intervention or enhanced screening might benefit high-risk individuals. However, most of the population falls in the broad middle of the distribution, where the PGS provides only modest discrimination.

Translating a polygenic score into absolute risk, such as a statement that an individual's 10-year probability of developing a disease is 15%, requires substantially more modeling. The PGS alone provides only relative ranking; converting to absolute probability requires knowledge of the baseline incidence rate in the relevant population, which varies by age, sex, ancestry, calendar time, and other factors. It also requires integrating the PGS with non-genetic risk factors (clinical measurements, family history, lifestyle variables) that independently contribute to disease risk. Finally, the resulting risk model must be calibrated to ensure that predicted probabilities match observed outcomes in relevant validation cohorts. A model that assigns 20% risk to a group should see approximately 20% of that group develop the disease; miscalibration undermines clinical utility and patient trust. We return to these issues of risk modeling, calibration, and clinical decision thresholds in Chapter Chapter 18, where we discuss the integration of genomic and clinical data for patient stratification.

3.5.2. Ancestry, Linkage Disequilibrium, and Transferability

A polygenic score derived in one population often performs substantially worse when applied to another, and this transferability problem is one of the most pressing challenges in the field. The decline in predictive accuracy is not uniform: scores developed in European-ancestry cohorts (which dominate the GWAS literature due to historical recruitment patterns) typically lose 20-80% of their

predictive power when applied to African-ancestry or East Asian-ancestry populations, with the magnitude of decline varying by trait and methodology.

Several factors contribute to this transferability gap. The most fundamental is differences in linkage disequilibrium structure across populations. Human populations have distinct demographic histories involving bottlenecks, expansions, and admixture events that have shaped their haplotype patterns. A variant that tags a causal variant through tight LD in European populations may be only weakly correlated with that same causal variant in African populations, where LD blocks tend to be shorter due to greater ancestral diversity. If a PGS relies heavily on such tag variants rather than causal variants, it will perform well only in populations with similar LD structure.

Allele frequency differences compound this problem. A variant that is common in one population may be rare or absent in another, and vice versa. GWAS statistical power depends on allele frequency, so the set of variants that reach significance (and thus enter into PGS) is shaped by the allele frequency spectrum of the discovery population. Effect size estimates are also noisier for rarer variants, so weights learned in one population may not transfer accurately even for shared variants.

Beyond these genetic factors, environmental and gene-by-environment interactions may differ across populations. The phenotypic consequence of a genetic variant can depend on diet, pathogen exposure, healthcare access, and countless other environmental variables that vary geographically and socioeconomically. A variant that increases cardiovascular risk in the context of a Western diet may have different effects in other dietary contexts. These interactions are rarely modeled explicitly in standard GWAS and PGS frameworks, contributing to transferability failures that cannot be explained by LD and allele frequency differences alone.

Large biobank efforts that recruit diverse populations, such as the Million Veteran Program, have brought these issues into sharp focus (Verma et al. 2024). Studies in these cohorts consistently find that PGS developed in European-ancestry samples explain less phenotypic variance in other ancestry groups, sometimes dramatically so. This disparity has direct implications for health equity: if genomic medicine tools work well only for populations that are already overrepresented in research, their clinical deployment risks widening rather than narrowing health disparities.

Several strategies can mitigate transferability problems. Multi-ancestry GWAS meta-analyses increase power to detect variants with consistent effects across populations while down-weighting ancestry-specific signals. Fine-mapping, as discussed earlier, can identify putative causal variants that should have consistent effects regardless of local LD patterns. Functional annotations from resources like ENCODE and GTEx (described in Chapter Chapter 2) can prioritize variants in regulatory regions with evidence of molecular function, on the theory that biologically active variants are more likely to be causal and thus more likely to transfer. And methods that explicitly model LD differences across populations, or that leverage admixed individuals who carry haplotypes from multiple ancestral backgrounds, can improve cross-population prediction. None of these approaches fully solves the transferability problem, but together they point toward a future where PGS reflect shared human biology rather than ancestry-specific statistical artifacts.

3.6. Limitations of GWAS and PGS, and the Case for Mechanistic Models

Despite their success in identifying thousands of trait-associated loci, GWAS and polygenic scores have fundamental limitations that motivate the rest of this book. They are tools of statistical association, not biological mechanism. They rely on linkage disequilibrium patterns that vary across populations. They are dominated by noncoding variants whose functional effects are difficult to interpret. And they provide no information about how genetic risk might interact with environment, treatment, or disease stage. These limitations are not merely technical inconveniences; they represent gaps in our understanding that sequence-based deep learning models are uniquely positioned to address. The chapters that follow will show how models that learn directly from DNA sequence can complement and extend the classical GWAS framework, moving from association toward mechanism.

3.6.1. Achievements and the Clinical Adoption Gap

The achievements of GWAS and polygenic scores should not be understated. Over the past two decades, GWAS has produced a systematic catalog of genetic associations for thousands of human traits and diseases, transforming our understanding of the genetic architecture of complex phenotypes. For some traits, particularly highly heritable quantitative phenotypes like height and lipid levels, polygenic scores explain a meaningful fraction of phenotypic variance and can identify individuals at substantially elevated risk. The methodology is mature, the computational pipelines are well-established, and GWAS summary statistics are increasingly available as public resources that enable secondary analyses without access to protected individual-level data.

Yet despite these successes, polygenic scores have seen limited adoption in routine clinical practice. This gap between research promise and clinical implementation reflects the accumulated weight of the limitations discussed throughout this chapter. Clinicians and healthcare systems have proven cautious about integrating PGS into care pathways, and for understandable reasons: the scores provide probabilistic stratification rather than actionable diagnosis, their performance varies across the diverse patient populations that healthcare systems serve, and the path from a percentile ranking to a clinical decision remains unclear for most conditions. The enthusiasm that greeted early PGS publications has given way to a more sober assessment of what these tools can and cannot deliver in their current form.

3.6.2. Association Without Mechanism

The first fundamental limitation is that GWAS and PGS operate at the level of statistical association rather than biological mechanism. A polygenic score tells us that certain variants are correlated with disease risk in the populations studied, but it does not tell us why. It does not identify which gene is affected, what molecular pathway is perturbed, or how the genetic signal might interact with therapeutic interventions. Two variants with identical weights in a PGS may have entirely different biological stories: one might directly disrupt a protein coding sequence, while another might be a tag variant in weak LD with an unknown regulatory element. This mechanistic opacity limits both scientific interpretation and clinical utility. Without understanding mechanism, we cannot easily move from risk prediction to risk modification.

3.6.3. Population Transferability

The second limitation is the dependence on linkage disequilibrium patterns and the resulting problems with portability across populations. As discussed in the previous section, PGS developed in European-ancestry cohorts often perform substantially worse in other ancestry groups. This is not merely a statistical inconvenience; it raises serious questions about equitable deployment. A healthcare system that offers PGS-based risk stratification to patients will systematically provide less accurate information to patients from underrepresented populations. The fact that most GWAS have been conducted in European-ancestry samples is a historical artifact of recruitment patterns and funding priorities, but its consequences propagate forward into any clinical tool built on those data.

3.6.4. The Noncoding Variant Challenge

The third limitation concerns the noncoding nature of most GWAS signals. The majority of trait-associated variants fall outside protein-coding regions, in the vast genomic territory devoted to gene regulation, chromatin organization, and functions we do not yet fully understand. Interpreting these noncoding variants is far more difficult than interpreting coding variants. A missense mutation that changes an amino acid can be evaluated with structural models, evolutionary conservation, and biochemical assays. A variant in an intergenic region might affect an enhancer active only in a specific cell type at a specific developmental stage, or it might have no functional consequence at all and simply tag a causal variant nearby. Understanding noncoding variation requires models of regulatory grammar that traditional GWAS does not provide.

3.6.5. Static Scores in a Dynamic Context

The fourth limitation is the static nature of conventional PGS. The score is computed once from germline genotypes and treated as a fixed quantity, but disease risk is not static. It changes with age, accumulates through environmental exposures, responds to medications, and evolves through disease progression. A polygenic score for cardiovascular disease does not account for whether the patient is taking statins, has changed their diet, or has already experienced a myocardial infarction. Integrating genetic risk with the rich longitudinal data available in electronic health records, including laboratory values, imaging, medications, and clinical notes, is essential for genomic prediction that is truly useful in clinical contexts.

3.6.6. Missing Heritability

A foundational limitation predates the transferability concerns discussed above: the gap between heritability estimated from family studies and the variance explained by GWAS-identified variants. Twin and family studies consistently estimate that common complex traits have substantial heritability, often 40% to 80% for traits like height, BMI, and psychiatric conditions. Yet early GWAS, despite identifying dozens or hundreds of associated loci, could account for only a small fraction of this heritability. This discrepancy, termed the “missing heritability” problem, prompted extensive methodological development and debate (**manolio_missing_2009?**).

Several factors contribute to the gap. Common variants with effect sizes too small to reach genome-wide significance individually may collectively explain substantial heritability, a possibility confirmed by methods that estimate heritability from all SNPs rather than just significant hits (**yang_common_2010?**). Rare variants, poorly tagged by genotyping arrays, likely contribute additional signal. Structural variants, gene-gene interactions, and gene-environment interactions are largely invisible to standard GWAS designs. And some portion of twin-study heritability may reflect shared environment or assortative mating rather than additive genetic effects. While methodological advances have closed much of the gap for some traits, the phenomenon illustrates a core limitation: GWAS-based approaches capture only a subset of genetic architecture, and the portion they miss may be precisely where mechanistic insight is most needed.

3.6.7. Toward Mechanistic Models

These limitations collectively motivate the approaches that form the core of this book. Sequence-based deep learning models offer a path from association toward mechanism by learning the relationship between DNA sequence and molecular function directly. Convolutional neural networks trained on regulatory assay data, such as DeepSEA and its successors, can predict how sequence changes affect transcription factor binding, chromatin accessibility, and gene expression (J. Zhou and Troyanskaya 2015; J. Zhou et al. 2018). Splicing models can predict how variants affect pre-mRNA processing (Chapter Chapter 7). These predictions are mechanistic in a way that GWAS effect sizes are not: they make claims about molecular function that can be tested experimentally.

Variant effect predictions from deep learning models can complement GWAS and fine-mapping to prioritize putative causal variants at trait-associated loci. If a fine-mapped credible set contains ten variants with similar posterior probabilities, but only one of them is predicted to substantially alter enhancer activity in a disease-relevant cell type, that variant becomes a higher-priority candidate for experimental follow-up. Resources like Open Targets Genetics already integrate such predictions alongside association statistics and fine-mapping results (Mountjoy et al. 2021).

Beyond prioritization, mechanistic predictions can be incorporated into polygenic score frameworks themselves. Rather than weighting variants purely by their GWAS associations, one can use predicted functional effects as priors, features, or reweighting factors. A variant predicted to disrupt a splice site or abolish transcription factor binding might receive greater weight than a variant with similar association statistics but no predicted functional consequence. This integration of statistical association with mechanistic prediction represents a promising direction for building scores that are more interpretable, more transferable across populations, and potentially more amenable to therapeutic intervention.

The genomic foundation models discussed in Part IV extend these ideas further. By training on massive corpora of sequence data with self-supervised objectives, these models learn representations that capture evolutionary constraints, regulatory syntax, and sequence-function relationships at a scale and generality that task-specific models cannot match. The goal is not to replace GWAS but to complement it: to provide the mechanistic context that association studies lack, to enable predictions for rare variants and understudied populations, and ultimately to close the gap between statistical genetics and biological understanding.

In later chapters we will see how multi-omics integration (Chapter Chapter 14) and clinical modeling (Chapter Chapter 18) build on these foundations to combine genetic, molecular, and clinical data for robust and equitable genomic prediction. For now, the key takeaway is that polygenic scores,

as powerful as they are for certain applications, remain fundamentally associative tools. They summarize correlation patterns in training populations without capturing the biological mechanisms that generate those patterns. Understanding LD, fine-mapping, and the distinction between causal and purely associative variants is essential background not only for interpreting classical PGS but also for appreciating what sequence-based deep learning models aim to achieve and how they might eventually transform genomic medicine.

4. Deleteriousness Scores

> ⚠ Warning
>
> **TODO:**
>
> - contrastive learning parallel?
> - CADD - first pub date
> - condense PLM and CNN sections?
> - Fix : **Modern deep learning approaches exploit protein language models, structure prediction from AlphaFold, and end-to-end neural architectures that learn directly from sequence.**
> - drop use of "we"
> - lean out redundant/repeats; migrate PLM and CNNs to later chapters as possible.

4.1. The Variant Prioritization Challenge

A typical human genome contains approximately four to five million genetic variants relative to the reference assembly. The vast majority of these are functionally neutral, representing the accumulated diversity of human evolution and population history. For any individual with a suspected genetic condition, the central interpretive challenge is to identify the handful of variants that plausibly contribute to disease from this enormous background of benign variation.

The data resources surveyed in Chapter 2 provide multiple complementary views of variant function, each with distinct strengths and limitations. Population frequency databases such as gnomAD reveal which variants survive in large cohorts of ostensibly healthy individuals, offering a powerful filter for identifying rare, potentially deleterious alleles ("The Genome Aggregation Database (gnomAD)" n.d.). Functional genomics consortia including ENCODE and the Roadmap Epigenomics Project indicate which genomic regions show evidence of biochemical activity across diverse cell types and developmental contexts. Clinical databases such as ClinVar and HGMD collect expert-curated variant classifications drawn from case reports and diagnostic laboratories, providing ground truth labels for known pathogenic and benign variants.

Each of these sources is partial in important ways. Population databases are dominated by common variants, which are mostly tolerated by virtue of their high frequency. Functional genomics data is inherently noisy and often context-specific: a region active in liver hepatocytes may be quiescent in neurons, and vice versa. Clinical databases are sparse and heavily biased toward well-studied genes and variant types, leaving vast swaths of the genome without reliable clinical annotations. Moreover, the annotation density varies dramatically across the genome: protein-coding exons are densely labeled relative to deep intronic and intergenic sequences.

Before deep learning, variant effect predictors typically tackled this problem by focusing on one narrow signal. Conservation-based methods such as phyloP and GERP score each position according to its evolutionary constraint across multi-species alignments, under the logic that positions conserved over hundreds of millions of years are likely functionally important (Siepel et al. 2005; Davydov et al. 2010). Protein-level tools such as SIFT and PolyPhen predict the impact of amino acid substitutions based on sequence homology, physicochemical properties, and structural features (Ng and Henikoff 2003; Adzhubei et al. 2010). Positional annotations capture simple features like distance to splice sites or proximity to known regulatory elements. Each of these approaches captures a real biological signal, but each is also incomplete: conservation scores miss recently evolved functional elements, protein-level tools are blind to non-coding variants, and positional annotations lack the resolution to distinguish causal variants from linked neutral neighbors.

Combined Annotation-Dependent Depletion (CADD) represented a fundamental shift in this landscape (Rentzsch et al. 2019). Rather than relying on a single predictive signal, CADD defined a general framework for genome-wide variant prioritization that integrates dozens of heterogeneous annotations and uses evolutionary depletion as a proxy training label. The key insight was to avoid training directly on small sets of known pathogenic versus benign variants, which are scarce and biased toward certain genes and variant types. Instead, CADD contrasts variants that have survived purifying selection in the human lineage with matched simulated variants that could have occurred but did not. This evolutionary proxy strategy yields an enormous training set, enables genome-wide coverage, and produces scores that generalize across coding and non-coding regions alike.

This chapter focuses on the CADD framework because it establishes design patterns that recur throughout the deep learning models covered in subsequent chapters: proxy labels derived from evolutionary signals, large-scale training on millions of examples, integration of diverse features into unified scores, and genome-wide precomputation for downstream reuse.

4.2. The Evolutionary Proxy Training Strategy

CADD's most important conceptual contribution was to reframe variant effect prediction as a large-scale machine learning problem with labels derived from evolutionary signal rather than clinical curation. The scarcity and bias of known pathogenic variants has long limited supervised approaches to variant interpretation. ClinVar and similar databases contain tens of thousands of labeled variants, but these are concentrated in a small fraction of genes, skewed toward certain variant types (nonsense, frameshift, canonical splice site), and subject to ascertainment bias from clinical referral patterns. Training directly on such labels tends to produce models that perform well on variants similar to the training set but generalize poorly to the broader genome.

CADD sidesteps this problem by constructing a synthetic classification task: can a model distinguish variants that are actually observed in human populations from matched simulated variants that have not survived evolution? The observed variants serve as proxies for tolerated alleles, while the simulated variants serve as proxies for potentially deleterious alleles. This framing yields a training set of tens of millions of examples, far exceeding what clinical curation can provide, and covers the full spectrum of variant types and genomic contexts.

4.2.1. Proxy-Neutral Variants

The proxy-neutral class consists of variants that are actually observed in human populations. CADD draws these from large sequencing datasets such as the 1000 Genomes Project and early gnomAD-like resources, including both single nucleotide variants (SNVs) and short insertions and deletions (indels). The selection criteria typically favor variants with high derived allele frequency, reflecting the assumption that alleles which have drifted to appreciable frequency in human populations are unlikely to be strongly deleterious over recent evolutionary timescales.

This is not a perfect proxy for benign variants. Some observed alleles are genuinely pathogenic, particularly those with incomplete penetrance, late onset, or context-dependent effects. Variants under weak negative selection may persist at low to moderate frequencies for thousands of generations before eventual elimination. Population-specific bottlenecks and founder effects can elevate the frequency of otherwise deleterious alleles in particular groups. Despite these caveats, the proxy-neutral class is, on average, substantially enriched for tolerated alleles relative to a random sample of possible mutations. The key statistical insight is that systematic enrichment, even if imperfect at the individual variant level, provides a useful training signal when aggregated across millions of examples.

4.2.2. Proxy-Deleterious Variants

The proxy-deleterious class is constructed by simulating mutations across the genome according to realistic mutational processes. The simulation matches local sequence context, typically using trinucleotide frequencies to capture the strong dependence of mutation rates on the identity of flanking bases. CpG dinucleotides, for example, have elevated mutation rates due to spontaneous deamination of methylated cytosines, and the simulation accounts for this by generating more CpG transitions in the proxy-deleterious set. Regional variation in mutation rates, driven by factors including replication timing, chromatin state, and local sequence composition, is similarly incorporated by scaling mutation counts within genomic windows (Rentzsch et al. 2019; Schubach et al. 2024).

The logic underlying this construction is subtle but powerful. Simulated variants represent changes that could plausibly occur under human mutational processes but are generally not observed at high frequency in population databases. Many of these simulated variants would in fact be neutral if they were to arise; the simulation makes no attempt to identify truly deleterious mutations at the individual level. However, the proxy-deleterious class as a whole is enriched for alleles that are disfavored by selection, because the set of possible mutations includes many that disrupt conserved elements, alter protein function, or perturb regulatory sequences. By contrasting this set with the proxy-neutral class, CADD learns to recognize the annotation signatures that distinguish variants under purifying selection from those that have been tolerated.

4.2.3. Training Objective

With proxy-neutral and proxy-deleterious classes in hand, CADD trains a binary classifier to distinguish between them. The input to this classifier is a feature vector describing each variant, encompassing the diverse annotations surveyed in the following section: gene model features, conservation scores, epigenetic signals, protein-level predictions, and more. The label is simply

whether the variant was simulated (proxy-deleterious) or observed (proxy-neutral). The objective is to learn a scoring function that assigns higher values to simulated variants, reflecting their predicted deleteriousness.

Early CADD versions employed linear support vector machines trained on approximately 30 million simulated versus observed variants with 63 annotation features plus selected interaction terms (Rentzsch et al. 2019). This relatively simple architecture was sufficient to capture the main structure of the problem, in part because the features themselves encode substantial biological knowledge. Later versions, including CADD v1.7, employ logistic regression-style models with expanded feature sets, retaining the same fundamental paradigm of contrasting simulated and observed variants while accommodating richer annotations (Schubach et al. 2024).

This evolutionary depletion framework anticipates several themes that recur in modern self-supervised learning. The labels are not clinical ground truth but derived from a proxy signal (survival under selection) that is abundant and covers the entire genome. The training set is extremely large, enabling complex decision boundaries and robust generalization. The resulting scores are precomputed genome-wide and reused for diverse downstream tasks, from rare disease gene discovery to variant filtration pipelines to evaluation baselines for newer models. In this sense, CADD can be understood as an early example of pretraining on a large-scale proxy task followed by transfer to clinical applications, a pattern that defines modern foundation models.

4.3. Integration of Diverse Annotations

CADD's second conceptual pillar is the integration of many weak, noisy annotations into a single composite score. Where earlier variant effect predictors typically relied on one or a few signals, CADD combines more than 60 features in its original incarnation and substantially more in version 1.7 (Rentzsch et al. 2019; Schubach et al. 2024). This integrative approach recognizes that no single annotation captures the full complexity of variant function. Conservation scores miss recently evolved functional elements. Protein-level predictions are uninformative for non-coding variants. Regulatory annotations are noisy and incomplete. By learning optimal weights for combining these diverse signals, CADD achieves performance that exceeds any individual component.

Because Chapter 2 already surveys the underlying data resources in detail, this section focuses on the categories of features and how they function within the CADD framework. For specifics on individual databases such as ENCODE, Roadmap, gnomAD, and ClinVar, readers should consult the earlier chapter.

4.3.1. Gene Model Annotations

Gene model annotations describe the local transcript and coding context of each variant. The most fundamental is the predicted sequence consequence: whether a variant is synonymous, missense, nonsense, frameshift, splice-site disrupting, or located in untranslated or intronic regions. These consequence categories capture qualitatively different modes of disruption, from silent changes that preserve protein sequence to truncating mutations that eliminate large portions of the gene product.

Beyond simple consequence, CADD incorporates positional features such as distance to exon-intron boundaries and proximity to canonical splice sites. Variants near splice junctions have elevated

potential to disrupt splicing even if they do not directly alter the canonical GT-AG dinucleotides. Distance to the start and stop codons provides additional context, as does the position within the reading frame for coding variants.

Gene-level attributes further enrich the annotation set. Constraint metrics derived from population data, such as the probability of loss-of-function intolerance (pLI) and the loss-of-function observed/expected upper bound fraction (LOEUF), quantify how tolerant each gene is to damaging variation ("The Genome Aggregation Database (gnomAD)" n.d.). Variants in highly constrained genes receive elevated deleteriousness scores, reflecting the empirical observation that such genes are enriched for disease associations. Known disease gene status from OMIM and similar resources provides complementary information about genes with established pathogenic roles.

These gene model features allow CADD to make biologically meaningful distinctions. A synonymous variant in a tolerant gene with high LOEUF receives a very different score than a truncating variant in a highly constrained developmental regulator. The model learns these distinctions from the differential representation of variant types across the proxy-neutral and proxy-deleterious training classes.

4.3.2. Conservation and Constraint

Evolutionary conservation provides some of the strongest signals for variant deleteriousness, particularly in non-coding regions where direct functional labels are scarce. CADD incorporates multiple conservation metrics computed from multi-species alignments spanning mammals, vertebrates, and more distant taxa.

Base-level conservation scores such as GERP (Genomic Evolutionary Rate Profiling) and phyloP quantify the deviation of observed substitution rates from neutral expectation (Davydov et al. 2010; Siepel et al. 2005). Positions with strong negative GERP scores show substitution rates far below neutral, indicating purifying selection has maintained these bases across tens or hundreds of millions of years of evolution. PhastCons provides a complementary view by identifying conserved elements, contiguous regions with elevated conservation that likely correspond to functional units. PhyloP scores individual positions without the smoothing implicit in element-based approaches, capturing both conservation (slow evolution) and acceleration (fast evolution) relative to neutral models.

Regional measures of constraint complement base-level scores. These capture broader patterns of evolutionary pressure that may not be evident at single positions but emerge when considering larger windows. Mutation rate estimates, derived from substitution patterns in presumably neutral regions such as ancestral repeats, allow the model to distinguish true constraint from low mutation rate.

Conservation features are particularly valuable for non-coding variant interpretation, where biochemical annotations are often incomplete or absent. A deeply conserved non-coding position is likely functional even if no enhancer or promoter annotation overlaps it. Conversely, lack of conservation provides evidence (though not proof) that a position is tolerant to variation.

4.3.3. Epigenetic and Regulatory Activity

CADD incorporates regulatory annotations derived from functional genomics assays to capture the chromatin and regulatory context of each variant. These features draw primarily from large-scale

consortium efforts including ENCODE and the Roadmap Epigenomics Project, which have profiled chromatin accessibility, histone modifications, and transcription factor binding across hundreds of cell types and tissues.

DNase I hypersensitivity and ATAC-seq peaks identify regions of open chromatin, marking active regulatory elements including promoters, enhancers, and insulators. ChIP-seq signals for histone modifications provide additional context: H3K4me3 marks active promoters, H3K27ac marks active enhancers, H3K36me3 spans transcribed gene bodies, and H3K27me3 marks polycomb-repressed regions. Transcription factor ChIP-seq directly identifies binding sites for specific regulators, though coverage varies considerably across factors and cell types.

Chromatin state segmentations integrate multiple histone marks and accessibility signals into discrete functional categories. These segmentations, produced by algorithms such as ChromHMM, assign each genomic position to states like "active promoter," "strong enhancer," "weak enhancer," "transcribed region," or "heterochromatin." By including these aggregate states alongside raw signals, CADD can capture combinatorial patterns that distinguish functional regulatory elements from background.

These epigenomic features help prioritize non-coding variants that disrupt active regulatory regions. A variant falling within an active enhancer marked by H3K27ac and DNase hypersensitivity in a relevant tissue receives elevated deleteriousness scores, even if its conservation is modest. The tissue specificity of regulatory annotations presents both opportunity and challenge: a variant may be highly consequential in one cellular context while neutral in another, and CADD's genome-wide scores necessarily average across this heterogeneity.

4.3.4. Additional Features

Beyond the major annotation categories, CADD incorporates features capturing local sequence context and genomic architecture. GC content and CpG dinucleotide density affect mutation rates, chromatin structure, and gene regulation. Segmental duplications and low-complexity regions flag positions where mapping uncertainty may confound variant calls or where duplicated sequences complicate interpretation. Distance to telomeres and centromeres provides coarse chromosomal context.

For coding variants, CADD includes protein-level features beyond simple consequence. Amino acid physicochemical properties, including size, charge, hydrophobicity, and polarity, inform predictions about whether a substitution is conservative or radical. Legacy variant effect scores such as SIFT and PolyPhen are included as features, allowing CADD to leverage decades of prior work on protein variant interpretation (Ng and Henikoff 2003; Adzhubei et al. 2010). Grantham distances quantify the biochemical dissimilarity between amino acid pairs, while secondary structure and domain annotations from databases like Pfam provide structural context.

Not every individual annotation is informative in isolation. Many features are noisy, incomplete, or redundant with one another. The power of CADD lies in learning how to weight and combine these heterogeneous signals, up-weighting annotations that distinguish proxy-deleterious from proxy-neutral variants and down-weighting those that do not. This learned integration is more powerful than any manually specified combination rule and adapts automatically as new features are added in subsequent versions.

4.4. Model Architecture and Scoring

4.4.1. Machine Learning Framework

CADD's classifier operates on a high-dimensional feature vector assembled for each variant. The input representation concatenates all annotations described in the previous section: gene model features, conservation scores, epigenomic signals, sequence context, and protein-level predictions where applicable. For a typical variant, this yields a vector of several dozen to over a hundred features, depending on the CADD version and whether the variant falls in coding or non-coding sequence.

Early CADD versions employed a linear support vector machine (SVM) trained on approximately 30 million observed and simulated variants with 63 annotation features plus selected interaction terms (Rentzsch et al. 2019). The choice of a linear model was deliberate: with tens of millions of training examples and dozens of features, a linear SVM is computationally tractable while still capturing the main structure of the classification problem. The linearity also provides some interpretability, as feature weights indicate which annotations most strongly distinguish the proxy classes.

In CADD v1.7, the framework transitions to a logistic regression-style model with an expanded annotation set exceeding 100 features (Schubach et al. 2024). The fundamental paradigm remains unchanged: contrast simulated and observed variants using a discriminative classifier. The expanded feature set incorporates new annotations including protein language model scores and regulatory CNN predictions (discussed in the following section), while the logistic formulation provides well-calibrated probability estimates that facilitate downstream score transformations.

Conceptually, the classifier learns a scoring function $s(x)$ such that large positive values indicate variants whose annotation profiles resemble the proxy-deleterious class, while large negative values indicate profiles resembling the proxy-neutral class. Variants with intermediate scores occupy an ambiguous middle ground where the annotation evidence does not clearly favor either class. The raw output of this classifier is often referred to as the C-score or raw CADD score.

4.4.2. PHRED-Scaled Scores

Raw CADD scores are not directly interpretable as probabilities or biological effect sizes. The scale depends on model architecture, feature normalization, and training set composition, all of which vary across CADD versions. To provide a more intuitive and stable scoring system, CADD defines PHRED-like scaled scores based on the rank of each variant among all possible single-nucleotide substitutions in the reference genome (Rentzsch et al. 2019; Schubach et al. 2024).

The PHRED scaling follows the same logarithmic convention used in sequencing quality scores. A scaled score of 10 indicates that a variant falls in the top 10% of predicted deleteriousness among all possible substitutions. A score of 20 indicates the top 1%, and a score of 30 indicates the top 0.1%. More generally, a scaled score of n corresponds to the top $10^{-n/10}$ fraction of the deleteriousness distribution. This transformation compresses the raw scores into a 1-99 range that reflects percentile rank rather than absolute effect size.

This rank-based transformation has several practical consequences. First, it provides a simple interpretation: users can immediately understand that a variant with scaled score 20 is predicted to be more deleterious than 99% of possible substitutions. Second, it ensures comparability across

CADD versions. Because the scaled score is defined relative to the full distribution of possible variants, a score of 20 always means "top 1%" even as the underlying model, features, and raw score distributions change between releases. Third, it sacrifices resolution in the bulk of the distribution. Most variants are predicted to be relatively benign, and these cluster in the low-score range where differences are difficult to interpret. The scaling concentrates dynamic range in the high-score tail where clinical interpretation typically focuses.

In rare disease pipelines, CADD scaled scores are commonly used as filters to enrich for potentially pathogenic variants before detailed interpretation. Typical thresholds range from 15 (top 3%) to 20 (top 1%) or higher, depending on the stringency required and the downstream analysis workflow. These filters are not intended as definitive pathogenicity calls but rather as prioritization tools that reduce the variant burden to a manageable number for expert review.

4.5. CADD v1.7: Integration of Deep Learning Predictions

CADD v1.7 demonstrates how the original annotation-integration framework naturally accommodates deep learning outputs and modern sequence models (Schubach et al. 2024). Rather than replacing CADD's architecture with an end-to-end neural network, the developers adopted a pragmatic strategy: treat deep learning predictions as additional features within the existing integrative framework. This approach preserves CADD's interpretable structure while benefiting from the representational power of large pretrained models.

4.5.1. Protein Language Model Features

For protein-coding variants, CADD v1.7 integrates variant effect scores from protein language models (PLMs), particularly ESM-1v (Meier et al. 2021). These models represent a paradigm shift in protein sequence analysis. Trained self-supervised on hundreds of millions of protein sequences using masked language modeling objectives, PLMs learn contextual embeddings that capture the evolutionary constraints and functional requirements shaping protein sequences. The resulting representations encode information about secondary structure, domain boundaries, binding interfaces, and catalytic sites without explicit supervision on any of these properties.

ESM-1v provides per-variant scores by comparing the log-likelihood of the reference and alternate amino acids at each position. Positions where the model confidently predicts the reference residue and assigns low probability to the alternate receive large effect scores, indicating the substitution violates learned sequence constraints. These scores correlate strongly with experimental measurements of variant effects from deep mutational scanning assays, demonstrating that PLMs capture genuine functional information.

By embedding ESM-1v-derived features into its annotation set, CADD v1.7 effectively delegates part of the representation learning to a large foundational protein model, then uses its own classifier to recalibrate and integrate these signals with other annotations (Schubach et al. 2024). This division of labor plays to each model's strengths: the PLM learns rich sequence representations from massive protein databases, while CADD's integrative framework combines these representations with genomic context, conservation, and regulatory features that protein-only models cannot access.

4.5.2. Regulatory CNN Predictions

For non-coding variants, CADD v1.7 incorporates regulatory variant effect predictions from sequence-based convolutional neural networks trained on chromatin accessibility and related assays (J. Zhou and Troyanskaya 2015; Schubach et al. 2024). These CNNs, exemplified by DeepSEA and similar architectures covered in Chapters 5 and 6, take raw DNA sequence as input and predict a battery of chromatin features including transcription factor binding, histone modifications, and DNase hypersensitivity across diverse cell types.

The variant effect predictions are computed as delta scores: the difference in predicted regulatory activity between reference and alternate alleles. Large magnitude deltas indicate variants predicted to substantially alter local chromatin state or transcription factor occupancy. These predictions provide a learned, sequence-based view of regulatory impact that complements the annotation-based epigenomic features derived from experimental data.

By incorporating CNN-derived regulatory predictions, CADD v1.7 uses early sequence-to-function deep learning models as feature generators within its broader integrative framework. This represents an important architectural pattern that recurs throughout genomic deep learning: pretrained sequence models provide representations or predictions that are then combined with other information sources in downstream tasks. The pretrained models capture sequence-intrinsic patterns, while the integrative framework adds genomic context and cross-annotation calibration.

4.5.3. Extended Conservation Scores

CADD v1.7 updates its conservation and mutation-rate features to incorporate advances in comparative genomics and population genetics. Deeper mammalian alignments from projects like Zoonomia, which sequenced over 200 mammalian species, provide substantially improved resolution for identifying constrained positions, particularly in non-coding regions where earlier alignments had limited power (Schubach et al. 2024). The expanded phylogenetic scope allows detection of constraint that is specific to mammals or particular mammalian clades, complementing the broader vertebrate and eukaryotic conservation captured by earlier alignments.

Improved models of genome-wide mutation rates sharpen the distinction between true evolutionary constraint and regions with inherently low mutation rates. Earlier approaches sometimes conflated these signals: a region might appear conserved simply because few mutations arise there rather than because mutations are selectively removed. By incorporating refined mutation rate estimates derived from de novo mutation studies and population polymorphism patterns, CADD v1.7 can better distinguish these scenarios and assign appropriate deleteriousness scores.

These updates are particularly valuable for non-coding variant interpretation, where conservation signals are often the strongest available evidence for function. Improved detection of mammal-specific regulatory elements and better calibration against local mutation rates help identify pathogenic non-coding variants that earlier versions might have missed.

4.5.4. Performance Improvements

CADD v1.7 is evaluated on several benchmark datasets that span different variant types and functional readouts (Schubach et al. 2024). Clinical variant benchmarks drawn from ClinVar and

gnomAD compare pathogenic and benign variant sets, providing a coarse approximation of the clinical classification task that motivates CADD's development. Deep mutational scanning (DMS) assays, summarized in resources like ProteinGym, offer experimentally measured variant effects for thousands of mutations across dozens of proteins, enabling evaluation against direct functional measurements rather than clinical labels (Notin et al. 2023). Saturation mutagenesis reporter assays for promoters and enhancers capture regulatory variant effects with nucleotide resolution, testing CADD's performance on the non-coding variants that are often most challenging to interpret.

Across these benchmarks, incorporating PLM scores, regulatory CNN predictions, and updated conservation features yields consistent improvements in classification and ranking performance compared to earlier CADD versions. The gains are particularly pronounced for missense variants, where ESM-1v features provide substantial additional signal, and for non-coding variants in active regulatory regions, where CNN predictions complement annotation-based features. These improvements validate the strategy of incorporating deep learning outputs as features while maintaining CADD's interpretable integrative framework.

4.6. Benchmarking Against Alternative Approaches

4.6.1. Coding Variants

For coding variants, CADD exists within a crowded landscape of deleteriousness predictors spanning four decades of methodological development. Legacy tools such as SIFT and PolyPhen pioneered sequence-based and structure-based prediction of amino acid substitution effects, using evolutionary conservation and physicochemical properties to identify potentially damaging missense variants (Ng and Henikoff 2003; Adzhubei et al. 2010). Ensemble methods such as REVEL, MetaLR, and M-CAP combine predictions from multiple individual tools, using machine learning to weight and integrate their outputs. **Modern deep learning approaches exploit protein language models, structure prediction from AlphaFold, and end-to-end neural architectures that learn directly from sequence.**

In systematic benchmarks across clinically annotated variants and deep mutational scanning datasets, CADD's combination of evolutionary, protein-level, and gene-context features yields performance that is competitive with or superior to many specialized scores for Mendelian disease variant prioritization (Rentzsch et al. 2019; Schubach et al. 2024). The integration of ESM-1v features in version 1.7 closes much of the gap with pure PLM-based methods while retaining CADD's advantages in interpretability and genome-wide coverage. CADD's performance is particularly strong when variants must be ranked across diverse genes and consequence types, a setting that favors integrative approaches over methods tuned for specific protein families or variant classes.

However, for focused applications within specific protein families or functional classes, specialized methods may outperform CADD. Tools optimized for loss-of-function variant interpretation may capture nuances that CADD's genome-wide training misses. Structure-based methods incorporating AlphaFold predictions can model three-dimensional context that sequence-based features cannot fully capture. The appropriate choice of variant effect predictor depends on the specific application, available data, and interpretability requirements.

4.6.2. Non-coding Variants

Non-coding variant interpretation presents fundamentally greater challenges than coding variant prediction. Ground-truth pathogenic non-coding variants are far rarer in clinical databases and heavily biased toward a small number of well-studied regulatory elements, particularly canonical splice sites and a handful of characterized enhancers. The vast majority of the non-coding genome lacks reliable pathogenicity labels, making supervised approaches difficult and benchmark construction problematic.

Functional genomics assays provide an alternative view of non-coding function, but their interpretation is complicated by noise, cell-type specificity, and the uncertain relationship between biochemical activity and phenotypic consequence. A variant may alter transcription factor binding in a reporter assay yet have no detectable effect on gene expression or organismal phenotype. Conversely, subtle regulatory perturbations may have profound effects in specific developmental contexts that are not captured by standard assays.

Within this challenging landscape, CADD's integration of regulatory annotations and conservation allows it to rank plausible non-coding candidates genome-wide, particularly in promoters and enhancers covered by ENCODE and Roadmap data (Rentzsch et al. 2019). The addition of regulatory CNN predictions in version 1.7 provides learned sequence-based features that extend beyond annotation coverage. However, CADD's performance on non-coding variants depends heavily on the availability and quality of underlying annotations. Variants in poorly annotated regions, including many distal enhancers and non-coding RNAs, receive scores driven primarily by conservation, which may miss recently evolved or lineage-specific functional elements.

4.6.3. Population Frequency Correlation

Because CADD uses evolutionary depletion as its training signal, its scores naturally correlate with population allele frequencies. Common variants in gnomAD tend to have low CADD scores, reflecting the expectation that alleles reaching high frequency have survived purifying selection. Very rare variants, particularly singletons observed in only one individual, show a broad distribution of scores with a substantial fraction in the high-score tail (Rentzsch et al. 2019; "The Genome Aggregation Database (gnomAD)" n.d.).

This correlation is useful for many applications. High CADD scores often highlight variants under purifying selection, which are enriched for functional and potentially pathogenic alleles. The relationship provides a sanity check: if CADD assigned high scores to common variants, something would be wrong with either the model or the frequency data.

However, this correlation also means that CADD partially recapitulates frequency-based filtering. In downstream pipelines, it is important not to double-count this signal by applying both aggressive frequency cutoffs and strict CADD thresholds. Such redundant filtering can exclude variants that fail one criterion but might be genuinely pathogenic. The optimal strategy depends on the application: for highly penetrant Mendelian variants, frequency filtering alone may suffice; for variants with incomplete penetrance or population-specific effects, CADD provides complementary information beyond frequency.

4.6.4. Limitations and Circularity with ClinVar

CADD is now deeply embedded in variant interpretation workflows worldwide, used by clinical laboratories, research groups, and diagnostic pipelines as a standard prioritization tool. This success raises an important methodological concern: potential circularity between CADD scores and clinical databases such as ClinVar.

Two forms of circularity are particularly relevant. First, evaluation circularity arises when CADD is assessed on benchmark datasets that were themselves influenced by CADD. ClinVar submissions increasingly incorporate in silico evidence, including CADD scores, as part of their classification process. When we evaluate CADD on post-2014 ClinVar variants after clinical curation has already used CADD, we risk overestimating performance because the model is partially being judged against labels it helped create (Schubach et al. 2024). Variants with high CADD scores are more likely to be classified as pathogenic, and variants classified as pathogenic form the positive evaluation set, creating a feedback loop that inflates apparent performance.

Second, broader sociotechnical feedback affects model development even if CADD's core training labels derive from simulated versus observed variants rather than clinical databases. ClinVar and related resources still influence feature engineering, threshold selection, and choice of evaluation benchmarks. Over time, variants consistently prioritized by CADD are more likely to receive follow-up investigation, be published, and enter ClinVar as likely pathogenic, reinforcing the underlying signal. This feedback is not unique to CADD but affects any widely used predictive tool in genomics and medicine.

These circularity concerns motivate several best practices for evaluation. Benchmarks should include datasets independent of clinical curation pipelines, such as deep mutational scanning experiments, reporter assays, and population-based burden tests where labels derive from experimental measurement rather than clinical judgment. Performance should be reported separately on pre-CADD and post-CADD ClinVar subsets when temporal stratification is possible. ClinVar-based evaluation should be treated as a sanity check confirming that CADD captures clinically relevant signals, not as the primary or sole measure of model quality.

These concerns foreshadow similar issues we will encounter in later chapters when genomic foundation models are evaluated on benchmarks that themselves rely on older predictive tools or clinical databases shaped by those tools.

4.7. Significance for Genomic Deep Learning

CADD occupies an important historical position at the junction between hand-crafted feature integration and modern deep, self-supervised representation learning. Several aspects of its design resonate throughout the models and methods covered in subsequent chapters, making it a valuable conceptual anchor for understanding the field's evolution.

The first connection is between annotation integration and multi-task deep models. CADD's strategy of combining dozens of heterogeneous annotations into a single score anticipates the multi-task learning frameworks that define modern genomic deep learning. Models like DeepSEA, Basset, and Enformer, covered in Chapters 5 through 7 and revisited in Chapter 11, predict hundreds of functional genomics readouts from sequence and then reuse these predictions as building blocks for downstream tasks. The conceptual structure is similar: learn to predict many weak signals, then

combine them for variant interpretation. In CADD v1.7, the boundary between these approaches blurs as deep networks including ESM-1v and regulatory CNNs provide features that CADD integrates (Meier et al. 2021; J. Zhou and Troyanskaya 2015; Schubach et al. 2024). The distinction between "annotation-based" and "deep learning-based" methods becomes one of degree rather than kind.

The second connection is between evolutionary proxy labels and self-supervised learning. CADD's training on simulated versus observed variants uses the signature of selection as a rich, weak supervisory signal available across the entire genome (Rentzsch et al. 2019). This strategy is conceptually parallel to the masked language modeling objectives that define modern protein and DNA language models. In both cases, the labels derive not from expert curation but from statistical regularities in large datasets: which tokens (amino acids, nucleotides, or variants) are observed versus which are plausible but absent. The resulting models learn representations that transfer to diverse downstream tasks, from variant effect prediction to structure determination to regulatory sequence design. Chapters 8 through 10 develop this connection in detail for transformer-based foundation models.

The third connection concerns genome-wide coverage and scalability. By precomputing scores for all possible single-nucleotide substitutions in the reference genome, CADD demonstrated the feasibility and utility of generating genome-wide variant annotations for downstream reuse. Users need not run the full model for each query; they simply look up precomputed scores from distributed files. Many genomic foundation models now follow an analogous pattern, precomputing embeddings or predictions for every base or variant and exposing them as reusable resources. The infrastructure for distributing and querying such precomputed annotations has become a standard component of genomic analysis pipelines.

The fourth connection is composability with deep learning. CADD is not a direct competitor to modern sequence-based deep models but rather an integrative framework that increasingly incorporates them as features. This "deep features plus shallow integrator" pattern appears repeatedly in practical deployments where interpretability, calibration, or computational constraints favor hybrid approaches over end-to-end neural networks. Clinical variant interpretation pipelines, in particular, often combine CADD-style integrative scores with deep learning predictions and expert review, leveraging the strengths of each approach.

As we move into the CNN-based sequence-to-function models of Part II and the transformer-based genomic foundation models of Parts III and IV, it is helpful to remember that CADD solved a difficult problem using tools available at the time. The challenge of variant prioritization under data scarcity and annotation heterogeneity does not disappear with more powerful models. The deep learning systems that follow expand on CADD's core ideas by learning representations directly from sequence and tying those representations to richer experimental readouts. Yet they still rely on many of the same data resources surveyed in Chapter 2 and confront many of the same challenges around evaluation bias, label circularity, and the fundamental difficulty of inferring causality from correlation. Understanding CADD's solutions and limitations provides essential context for appreciating both the advances and the persistent challenges in genomic deep learning.

Part II.

Part II: CNN Seq-to-Function Models

5. Regulatory Prediction

> ⚠ Warning
>
> **TODO:**
>
> - Citations: verify all citations are in bibliography
> - Add figure showing DeepSEA architecture
> - Consider adding ISM visualization example

5.1. The Noncoding Variant Challenge

The vast majority of disease-associated variants identified by GWAS lie in noncoding regions of the genome. Across thousands of loci mapped to complex traits, only a small minority directly alter protein-coding sequences; the remainder fall in introns, intergenic regions, and putative regulatory elements where their functional consequences are far less obvious. This presents both an interpretive challenge and an opportunity. If we could predict how noncoding variants affect gene regulation, we would have a powerful tool for moving from statistical association to biological mechanism.

Yet in 2015, the field lacked systematic methods to predict how noncoding variants affect regulatory activity. Existing approaches relied on overlap with known annotations: if a variant fell within a ChIP-seq peak or DNase hypersensitive site, it might be flagged as potentially functional. This strategy had obvious appeal, since it grounded predictions in experimental observations, but it suffered from fundamental limitations. Overlap-based annotation offered no mechanism for predicting the direction or magnitude of a variant's effect on regulatory activity. A variant might fall within an enhancer, but would it strengthen or weaken the enhancer? By how much? These questions could not be answered by checking whether genomic coordinates intersected. Furthermore, overlap-based methods could not score variants in regions lacking experimental coverage, which was problematic given that functional genomics experiments, despite their scale, still covered only a fraction of cell types and conditions.

DeepSEA, introduced by Zhou and Troyanskaya in 2015, fundamentally changed this paradigm by learning to predict chromatin features directly from DNA sequence (J. Zhou and Troyanskaya 2015). Rather than asking "does this variant overlap a known regulatory element?", DeepSEA asks "what regulatory activities does this sequence encode, and how would a mutation change them?" This shift from annotation lookup to sequence-based prediction opened a new chapter in computational genomics, one where deep neural networks could learn the relationship between DNA sequence and molecular function without requiring hand-crafted features or explicit motif definitions.

5.2. Learning Regulatory Code from Sequence

DeepSEA's central insight was that deep convolutional networks could learn the sequence patterns underlying regulatory activity without explicit feature engineering. This represented a departure from earlier computational approaches to regulatory genomics, which typically required defining sequence features a priori. Methods like gapped k-mer SVMs (gkm-SVM) required specifying which k-mers to count and how to weight them. Position weight matrices for transcription factor binding sites required curating motif databases like JASPAR or TRANSFAC. These approaches worked, but they encoded human assumptions about what sequence features mattered and could not easily discover novel patterns or complex dependencies.

DeepSEA instead learned relevant sequence features automatically from data. The convolutional layers of the network function analogously to motif scanners, detecting local sequence patterns that correlate with regulatory activity. But unlike predefined motif scanners, these filters are learned during training, allowing the network to discover whatever patterns best predict the training labels. Deeper layers in the network can then learn combinations of these patterns, capturing regulatory "grammar" such as motif spacing, orientation preferences, and cooperative binding arrangements. The network does not know in advance which patterns matter; it learns them by optimizing predictions on hundreds of thousands of genomic sequences with experimentally measured chromatin profiles.

5.2.1. Architecture

The original DeepSEA architecture was deliberately simple by modern standards, comprising a stack of convolutional layers followed by fully connected layers that integrate information across the sequence.

The input to the network is a 1000 bp DNA sequence, one-hot encoded into a binary matrix with four channels (one per nucleotide) and 1000 positions. This representation treats sequence as a signal to be processed by convolution, analogous to how image recognition networks process pixel values. Each position in the sequence is represented by exactly one active channel, encoding which nucleotide (A, C, G, or T) is present.

The network processes this input through three convolutional layers, each followed by ReLU activation and max pooling. The first convolutional layer uses 320 filters of width 8, scanning the sequence for local patterns roughly the size of transcription factor binding sites. Max pooling after each convolution reduces the spatial dimension, progressively compressing the 1000-position input into a more compact representation. The second and third convolutional layers use 480 and 960 filters respectively, with narrower widths (8 and 8) applied to the already-pooled representation. These deeper layers can learn combinations of the patterns detected by earlier layers, building increasingly abstract representations of sequence features.

After the convolutional stack, a fully connected layer with 925 units integrates information across all positions in the compressed representation. This layer allows the network to learn relationships between sequence features at different positions, capturing spatial dependencies that pure convolution cannot represent. Finally, an output layer with 919 sigmoid units produces independent probability predictions for each chromatin profile.

The total number of parameters is modest by contemporary standards, approximately 60 million, but was substantial for genomics applications at the time. Training used stochastic gradient descent with momentum on sequences sampled from the human genome, with chromosome 8 held out for testing.

5.2.2. Training Data

DeepSEA was trained on 919 chromatin profiles compiled from ENCODE and Roadmap Epigenomics, two consortium efforts that had systematically mapped the epigenomic landscape across diverse human cell types and tissues (**encode_integrated_2012?**; **roadmap_integrative_2015?**). These profiles represented three major categories of regulatory annotation.

Transcription factor binding profiles, numbering 690 in total, captured the genomic locations where specific proteins bind DNA. These were derived from ChIP-seq experiments targeting factors like CTCF (a ubiquitous insulator protein), p53 (a tumor suppressor), and GATA1 (a hematopoietic transcription factor). Each profile represents a binary classification problem: for a given sequence, is the central region bound by this factor in this cell type?

Histone modification profiles, numbering 104, captured the locations of specific chemical modifications to histone proteins. Marks like H3K4me3 (trimethylation of lysine 4 on histone H3) are associated with active promoters, while H3K27ac (acetylation of lysine 27) marks active enhancers. H3K27me3 marks repressed regions through Polycomb-mediated silencing. These modifications do not directly encode regulatory logic but reflect the functional state of chromatin and correlate with gene expression.

DNase I hypersensitivity profiles, numbering 125, captured regions of open chromatin across cell types. DNase hypersensitive sites mark regions where DNA is accessible to regulatory proteins, identifying potential regulatory elements regardless of which specific factors bind there. Unlike transcription factor ChIP-seq, DNase-seq provides a relatively unbiased view of regulatory potential.

For each 1000 bp input sequence, the model predicts the probability that the central 200 bp region exhibits each of these 919 chromatin features. The narrower prediction window relative to the input window allows the network to use flanking sequence as context for predicting the central region's activity. Training used sequences sampled from the human genome, excluding chromosome 8 which was reserved for evaluation. This chromosome-level holdout prevents overfitting to sequence homology or LD patterns that might leak between training and test sets.

5.2.3. Multi-Task Learning

A key architectural decision was predicting all 919 features simultaneously rather than training separate models for each. This multi-task learning approach offers several advantages that compound as the number of tasks increases.

Shared representations in early layers benefit all tasks. The first convolutional layer learns general sequence features such as GC content, dinucleotide frequencies, and common motifs that are useful across many prediction problems. By sharing these representations, the network amortizes the cost of learning basic sequence features across all tasks rather than relearning them independently.

Joint prediction provides regularization. Predicting many correlated features simultaneously prevents overfitting to any single task. If a convolutional filter becomes overly specific to one transcription factor, it will harm predictions for other related factors, providing a pressure toward learning generalizable representations. This implicit regularization is particularly valuable when some tasks have limited training data.

Efficiency gains are substantial. One model serving all 919 prediction tasks requires far less computation than training and maintaining 919 separate models. This matters not only for initial training but for deployment, where a single forward pass produces all predictions.

The multi-task framework also reveals relationships between chromatin features. Weights connecting shared representations to different output tasks can be analyzed to understand which features rely on similar sequence patterns. This provides a form of interpretability that separate models would not offer.

5.3. Predicting Variant Effects

With a trained model that maps sequence to chromatin profiles, variant effect prediction becomes straightforward in principle: predict chromatin profiles for both reference and alternative allele sequences, then compute the difference. This produces a 919-dimensional vector describing how the variant is predicted to alter regulatory activity across all profiled features. A variant might be predicted to increase CTCF binding while decreasing DNase accessibility, or to have no effect on any chromatin feature, depending on where it falls and what sequence context it disrupts or creates.

This approach has a crucial property: it requires no training on variant data. The model learns to predict chromatin profiles from sequence during training, using only reference genome sequences and their experimentally measured chromatin states. Variant effect prediction is then a form of transfer: the model applies what it learned about sequence-function relationships to score mutations it has never seen. This ab initio capability distinguishes sequence-based models from approaches that learn directly from observed variant effects, which are inevitably biased toward common variants where statistical power exists.

5.3.1. Single-Nucleotide Sensitivity

For the approach to work, the model must achieve single-nucleotide sensitivity: changing one base must be capable of substantially altering predictions. This is not guaranteed. A model could achieve good performance on chromatin prediction by learning only coarse sequence features (GC content, repeat density) that are insensitive to point mutations. Such a model would be useless for variant interpretation.

DeepSEA achieves genuine single-nucleotide sensitivity, and the authors validated this using allelic imbalance data from digital genomic footprinting. For 57,407 variants showing allele-specific DNase I sensitivity across 35 cell types, DeepSEA predictions correlated strongly with the experimentally observed allelic bias. Variants predicted to increase chromatin accessibility tended to show higher accessibility on the corresponding allele, and vice versa. This correlation would not exist if the model were insensitive to point mutations.

The validation is particularly compelling because allelic imbalance represents an independent experimental readout. The model was not trained to predict allelic imbalance; it was trained to predict chromatin profiles from reference sequences. That it correctly predicts the direction of allelic effects demonstrates that the learned sequence-function relationships capture genuine biology rather than spurious correlations.

5.3.2. In Silico Saturation Mutagenesis

Beyond scoring individual variants, DeepSEA enables a powerful computational experiment: in silico saturation mutagenesis (ISM). By systematically predicting effects of all possible single-nucleotide substitutions within a sequence, one can identify which positions are most critical for regulatory function. At each position, three alternative nucleotides can be substituted, and the predicted change in chromatin profiles can be computed for each. Positions where substitutions produce large predicted effects are presumably functionally constrained, while positions tolerant of substitution are less critical.

ISM analysis of regulatory elements reveals sequence positions where mutations would most strongly perturb function. These critical positions often correspond to transcription factor binding motifs learned by the model. When the predicted effects are visualized along a regulatory sequence, clear patterns emerge: core motif positions show strong predicted effects, while flanking positions are more tolerant. This provides a form of motif discovery that emerges from the model's learned representations rather than from explicit motif searching.

The computational cost of ISM is linear in sequence length: for a 1000 bp sequence, 3000 forward passes are required (three substitutions per position). This is tractable for individual regions of interest, and precomputed ISM scores for the entire genome can be generated with sufficient computational resources.

5.4. Functional Variant Prioritization

Beyond predicting chromatin effects for individual variants, DeepSEA introduced a framework for prioritizing likely functional variants among large sets of candidates. This addresses a practical problem in human genetics: GWAS and sequencing studies identify many variants in a region, most of which are not causal. Which variants should be prioritized for follow-up?

5.4.1. eQTL Prioritization

Expression quantitative trait loci (eQTLs) represent variants statistically associated with gene expression changes. However, most eQTL signals reflect linkage disequilibrium rather than direct causation. A lead eQTL variant may simply be correlated with the true causal variant, which could be any of dozens of SNPs in the same LD block. Distinguishing causal variants from their correlated neighbors is essential for understanding regulatory mechanisms and for transferring findings across populations where LD patterns differ.

DeepSEA demonstrated improved ability to distinguish likely causal eQTL variants from nearby non-causal variants compared to overlap-based methods. The intuition is straightforward: if a variant

is truly causal, it should disrupt a sequence feature that matters for gene regulation. A variant that happens to be in LD with the causal variant but does not itself disrupt regulatory sequences should have minimal predicted effect. By ranking variants according to predicted regulatory impact, DeepSEA can prioritize those most likely to be causal.

5.4.2. GWAS Variant Prioritization

Similarly, for GWAS-identified disease associations, DeepSEA helped prioritize which variants in LD blocks were most likely causal. The model outperformed contemporary methods including GWAVA (which was trained on known regulatory mutations) on held-out benchmarks. This was notable because DeepSEA was not trained on variant data at all; its variant prioritization ability emerged from learning sequence-chromatin relationships, not from learning which variants are pathogenic.

5.4.3. Comparison to Prior Methods

DeepSEA's performance advantage over gkm-SVM was particularly notable for transcription factor binding prediction. The deep CNN achieved higher AUC for nearly all transcription factors tested. More revealing was the pattern with respect to sequence context: gkm-SVM showed no improvement when given longer input sequences (extending context from 200 bp to 500 bp to 1000 bp), while DeepSEA performance improved substantially with additional context.

This difference reflects the fundamental limitation of gapped k-mer methods. By counting k-mers and learning weights for them, gkm-SVM can capture the presence of individual motifs but struggles to learn relationships between motifs at different positions. The same k-mers in different spatial arrangements contribute identically to the score. Deep convolutional networks, by contrast, learn hierarchical representations where deeper layers can capture spatial dependencies between features detected by earlier layers. Additional sequence context provides more opportunities for these dependencies to inform predictions.

5.5. Evolution of the DeepSEA Framework

The original DeepSEA established the sequence-to-chromatin prediction paradigm. Subsequent work from the same research group expanded and refined this approach, building a lineage of models with progressively greater scope and sophistication.

5.5.1. DeepSEA Beluga (2018)

ExPecto, published in 2018, included an updated chromatin prediction model nicknamed "Beluga" that served as the foundation for tissue-specific expression prediction (J. Zhou et al. 2018). Beluga incorporated several architectural improvements over the original DeepSEA. The number of predicted chromatin profiles expanded from 919 to 2,002, covering additional transcription factors and histone modifications across more cell types. The architecture deepened, adding additional convolutional layers with residual connections that facilitated training and improved gradient flow. The input context expanded from 1000 bp to 2000 bp, allowing the model to capture longer-range sequence dependencies.

These improvements were motivated by the downstream application: predicting gene expression requires integrating regulatory signals across tens of kilobases around each transcription start site. A more capable chromatin prediction model, applied at multiple positions around a gene, provides richer features for expression prediction. The Beluga chromatin model is discussed further in Chapter 6, where it forms the first component of the ExPecto expression prediction framework.

5.5.2. Sei (2022)

Sei represents the current state of the DeepSEA lineage, predicting 21,907 chromatin profiles, a 24-fold expansion over the original (Chen et al. 2022). This dramatic scaling required both more training data (from expanded ENCODE and Roadmap datasets) and architectural innovations to handle the increased output dimensionality efficiently.

The Sei architecture introduces dual linear and nonlinear paths: parallel convolution blocks, one with activation functions and one without, allowing the model to learn both complex nonlinear patterns and simpler linear relationships. This design reflects the observation that some chromatin features depend on subtle nonlinear combinations of sequence features, while others are well predicted by simpler linear combinations. Dilated convolutions expand the receptive field without reducing spatial resolution, allowing the network to integrate information across longer distances without aggressive pooling. Spatial basis functions provide a memory-efficient mechanism for integrating information across positions, reducing the parameter count that would otherwise grow prohibitively with the number of output features.

Sei improved over Beluga by 19% on average (measured by AUROC/(1-AUROC), a metric that emphasizes improvements at high performance levels) on the 2,002 profiles predicted by both models. Beyond raw prediction performance, Sei introduced sequence class annotations that cluster the 21,907 chromatin predictions into interpretable regulatory categories, facilitating biological interpretation of model outputs.

Model	Year	Chromatin Targets	Input Length	Architecture
DeepSEA	2015	919	1000 bp	3 conv + FC
Beluga	2018	2,002	2000 bp	Deep residual CNN
Sei	2022	21,907	4000 bp	Dual-path + dilated conv

5.6. What DeepSEA Learns

Analyzing what neural networks learn is notoriously difficult, but several approaches have been applied to DeepSEA and its successors, revealing that the models capture biologically meaningful sequence patterns.

5.6.1. Motif Discovery

The first convolutional layer of DeepSEA contains filters that scan the input sequence for local patterns. By visualizing these filters as position weight matrices (treating the learned weights as

log-odds scores) or by identifying sequences that maximally activate each filter, researchers can examine what patterns the network has learned to detect.

Analysis of DeepSEA's first-layer filters reveals learned sequence patterns corresponding to known transcription factor binding motifs. Many filters match canonical motifs from databases like JASPAR, indicating that the network has independently discovered the sequence preferences of well-characterized transcription factors. This is reassuring: it confirms that the network is learning biologically relevant patterns rather than spurious correlations.

Deeper layers capture more complex patterns that do not correspond to individual motifs. These representations are harder to interpret but presumably encode combinations of motifs and spatial arrangements that predict chromatin state.

5.6.2. Regulatory Grammar

Beyond individual motifs, DeepSEA implicitly learns aspects of regulatory "grammar," the rules governing how motifs combine to produce regulatory activity. This includes motif spacing requirements (some transcription factor pairs require specific distances between their binding sites for cooperative function), motif orientation preferences (certain motifs function only in specific orientations relative to each other or to the gene), and combinatorial logic (multiple weak motifs can synergize, or overlapping sites can create competition between factors).

These grammatical rules are not explicitly represented in the model architecture; they emerge from learning to predict chromatin profiles from sequence. The deep architecture provides the representational capacity to encode complex dependencies, and the training procedure discovers whatever dependencies best predict the training labels.

However, the original DeepSEA architecture's limited receptive field constrained its ability to learn long-range dependencies. Max pooling after each convolutional layer progressively reduces spatial resolution, and the fully connected layer can only integrate information from the resulting compressed representation. Dependencies spanning hundreds or thousands of base pairs, such as enhancer-promoter communication, are difficult to capture in this framework. This limitation motivated later architectures with expanded context windows, culminating in models like Enformer (Chapter 11) with effective receptive fields spanning hundreds of kilobases.

5.7. Limitations and Considerations

DeepSEA represented a major advance, but understanding its limitations is essential for appropriate application and for appreciating the motivations behind subsequent developments.

5.7.1. Cell Type Specificity

DeepSEA predicts chromatin profiles for specific cell types included in training, but the same sequence may have different regulatory activity in cell types not represented. The model cannot extrapolate to novel cell types without relevant training data. If a user wants to predict regulatory activity in a cell type that was not profiled by ENCODE or Roadmap, DeepSEA provides no

principled way to do so. The prediction would either be unavailable or would require assuming that a related profiled cell type is a reasonable proxy.

This limitation is intrinsic to the supervised learning framework: the model learns input-output mappings for the cell types present in training data. Extending predictions to new cell types would require either profiling those cell types experimentally (creating new training labels) or developing methods that transfer learned representations across cell types, an active area of research in subsequent work.

5.7.2. Context Independence

The model treats each input sequence independently, without considering the broader genomic or cellular context in which that sequence operates. Three important contextual factors are absent from the model.

Three-dimensional chromatin structure brings distant genomic sequences into spatial proximity, allowing enhancers to regulate promoters located hundreds of kilobases away on the linear chromosome. DeepSEA sees only the linear sequence within its 1000 bp window; it cannot know whether distant regulatory elements are spatially proximal in the nucleus.

The current transcriptional state of the cell affects chromatin accessibility and transcription factor availability. A sequence might have regulatory potential that is realized only when certain factors are expressed. DeepSEA predicts potential regulatory activity based on sequence alone, not actual activity conditioned on cellular state.

Other variants in the same individual (epistasis) may modify the effect of any single variant. DeepSEA predicts effects for each variant in isolation, against the reference genome background. In reality, individuals carry thousands of variants, some of which may interact.

5.7.3. Quantitative Accuracy

While DeepSEA accurately predicts the binary presence or absence of chromatin features, its quantitative predictions of signal strength are less reliable. The model outputs probabilities that a region exhibits each feature, but these probabilities do not directly correspond to the magnitude of ChIP-seq or DNase-seq signal intensity. A region might be correctly predicted as bound by a transcription factor, but the model provides limited information about whether binding is strong or weak.

Later models addressed this limitation by predicting continuous coverage tracks rather than binary peaks. Basenji, introduced in 2018, predicted normalized read coverage across the genome, providing quantitative predictions that could be directly compared to experimental measurements (**kelley_basenji_2018?**). This shift from classification to regression enabled more nuanced variant effect predictions, where the question becomes not just "does this variant disrupt binding?" but "by how much does this variant change binding affinity?"

5.8. Significance for the Field

DeepSEA established several paradigms that shaped subsequent genomic deep learning. These contributions extend beyond the specific model to influence how the field approaches sequence-to-function prediction more broadly.

The "sequence-in, function-out" paradigm treats DNA sequence as the sole input and molecular function as the output, learning the mapping without hand-engineered features. This end-to-end learning approach allows the model to discover relevant patterns from data rather than encoding assumptions about what patterns matter. Subsequent models have extended this paradigm to predict increasingly complex functional readouts, from expression levels to splicing outcomes to three-dimensional chromatin organization.

Multi-task chromatin prediction, jointly modeling many related tasks, proved both more efficient and more effective than training separate models. The shared representations and implicit regularization that emerge from multi-task learning have become standard in genomic deep learning. Modern models routinely predict hundreds or thousands of outputs simultaneously, leveraging correlations between tasks to improve predictions for each.

Variant effect prediction via sequence comparison, scoring variants by comparing predictions for reference and alternative alleles, provided a general framework for interpreting genetic variation. This approach extends naturally to any sequence-based model: if the model predicts molecular function from sequence, it can predict how mutations alter that function. The ab initio nature of this prediction, requiring no training on variant data, enables scoring of rare variants where population data is sparse.

The approach demonstrated that deep learning could extract biologically meaningful patterns from raw sequence data at scale. Convolutional filters learn motifs, deeper layers learn combinations, and the resulting representations support accurate prediction and variant interpretation. This opened the door to increasingly sophisticated sequence-to-function models predicting not just chromatin state, but gene expression (ExPecto, Chapter 6), splicing (SpliceAI, Chapter 7), and eventually long-range regulatory interactions (Enformer, Chapter 11).

DeepSEA's public web server (http://deepsea.princeton.edu/) and code release also established a model for making genomic deep learning tools accessible to the broader research community. Rather than keeping trained models proprietary, the authors provided both a web interface for casual users and downloadable code and weights for computational researchers. This practice of open release has become standard in the field, accelerating progress by allowing others to build on published work rather than reimplementing from scratch.

The model's success also catalyzed interest in deep learning among genomics researchers who had previously worked with simpler statistical methods. By demonstrating that neural networks could learn interpretable and useful representations of regulatory sequence, DeepSEA helped establish genomics as a legitimate application domain for deep learning and attracted researchers from both communities to the intersection.

6. Transcriptional Effects

> ⚠ Warning
>
> **TODO:**
>
> - Citations: verify all citations are in bibliography
> - Consider adding figure showing ExPecto architecture diagram

6.1. From Chromatin to Expression

DeepSEA (Chapter 5) demonstrated that deep learning could predict chromatin features from DNA sequence alone. Yet chromatin accessibility and transcription factor binding are intermediate phenotypes. The ultimate functional readout for most regulatory variants is their effect on gene expression. A variant might disrupt a transcription factor binding site, but does that binding site actually regulate a nearby gene? In which tissues? By how much?

ExPecto, introduced by Zhou et al. in 2018, addressed these questions by extending the sequence-to-chromatin paradigm to predict tissue-specific gene expression levels (J. Zhou et al. 2018). The framework's name reflects its core capability: expression prediction. Rather than stopping at chromatin predictions, ExPecto integrates predicted regulatory signals across a 40 kb promoter-proximal region to predict absolute expression levels in 218 tissues and cell types.

Critically, ExPecto predicts expression effects ab initio from sequence, without training on any variant data. This enables scoring of rare variants, de novo mutations, and even hypothetical mutations never observed in any population.

6.2. The Modular Architecture

ExPecto comprises three sequential components, each addressing a distinct computational challenge.

6.2.1. Component 1: Epigenomic Effects Model (Beluga CNN)

The first component is an enhanced version of DeepSEA, predicting 2,002 chromatin profiles (histone marks, transcription factor binding, and DNase hypersensitivity) across more than 200 cell types. Key architectural improvements over the original DeepSEA include expanded chromatin targets (from 919 to 2,002), a wider input window (from 1,000 bp to 2,000 bp), deeper architecture (six

convolutional layers with residual connections rather than three), and broader cell type coverage (over 200 cell types compared to approximately 125).

Feature	DeepSEA (2015)	ExPecto/Beluga (2018)
Chromatin targets	919	2,002
Input window	1,000 bp	2,000 bp
Convolution layers	3	6 (with residual connections)
Cell types	~125	>200

The CNN scans the 40 kb region surrounding each transcription start site (TSS) with a moving window (200 bp step size), generating chromatin predictions at 200 spatial positions. For each gene, this produces 2,002 × 200 = 400,400 features representing the predicted spatial chromatin organization around the TSS.

6.2.2. Component 2: Spatial Feature Transformation

The 400,400-dimensional feature space poses optimization challenges for downstream expression prediction. ExPecto addresses this through spatial transformation, a biologically motivated dimensionality reduction that captures the known distance-dependent relationship between regulatory elements and their target promoters.

The transformation applies ten exponential decay functions separately to upstream and downstream regions. The full model specification is:

$$\text{expression} = \sum_{i,k} \left(\beta_{ik}^{\text{up}} \cdot \mathbf{1}(t_d < 0) + \beta_{ik}^{\text{down}} \cdot \mathbf{1}(t_d > 0)\right) \cdot \sum_{d \in D} p_{id} \cdot e^{-a_k \cdot |t_d|}$$

where p_{id} is the predicted probability for chromatin feature i at spatial bin d, t_d is the mean distance to TSS for bin d, and a_k represents decay constants (0.01, 0.02, 0.05, 0.1, 0.2). The indicator functions $\mathbf{1}(\cdot)$ allow separate coefficients for upstream (β^{up}) and downstream (β^{down}) regions.

This transformation reduces dimensionality 20-fold (to 20,020 features) while preserving spatial information. Features with higher decay rates are concentrated near the TSS, while lower decay rates capture more distal signals. The transformation is not learned but prespecified, equivalent to constraining the model to learn smooth spatial patterns as linear combinations of basis functions.

6.2.3. Component 3: Tissue-Specific Linear Models

The final component comprises 218 L2-regularized linear regression models (one per tissue), each predicting log RPKM expression from spatially-transformed features. Linear models were chosen deliberately: they provide interpretability, prevent overfitting given the high-dimensional feature space, and enable straightforward coefficient analysis to identify which chromatin features drive expression in each tissue.

Training used gradient boosting with L2 regularization (=100, shrinkage =0.01), with chromosome 8 held out for evaluation (990 genes). The chromosome-level holdout prevents data leakage through overlapping regulatory regions and sequence homology.

6.3. Expression Prediction Performance

ExPecto achieved 0.819 median Spearman correlation between predicted and observed expression (log RPKM) across 218 tissues and cell types, a substantial improvement over prior sequence-based expression models, which were typically limited to narrower regulatory regions (<2 kb) and fewer cell types.

6.3.1. Tissue Specificity

Beyond predicting absolute expression levels, ExPecto captures tissue-specific expression patterns. Expression predictions correlate more strongly with experimental measurements from the matching tissue than from other tissues, indicating the model learns tissue-specific regulatory logic rather than generic sequence features.

Analysis of model coefficients reveals automatic learning of cell-type-relevant features without explicit tissue labels. The liver expression model assigns top weights to seven transcription factors profiled in HepG2 (liver-derived) cells. The breast tissue model weights estrogen receptor (ER-) and glucocorticoid receptor (GR) features from breast cancer cell lines T-47D and ECC-1 most heavily among its positive coefficients. Blood cell expression models derive their top five predictive features from blood cell lines and erythroblast cells. These patterns emerge purely from learning to predict expression, without any tissue identity information provided to the chromatin features.

6.3.2. Feature Importance

Model coefficients also reveal the relative contributions of different chromatin feature types to expression prediction. Transcription factors and histone marks receive consistently higher weights, reflecting their direct mechanistic roles in transcriptional regulation. DNase I features receive significantly lower weights ($p = 6.9\times10^{-2}$, Wilcoxon rank sum test) despite indicating regulatory activity. This discrepancy likely reflects that DNase hypersensitivity marks the presence of regulatory activity without specifying its type (activating versus repressing) or its causal relationship to expression.

6.4. Variant Effect Prediction

ExPecto's expression predictions enable scoring variant effects through in silico mutagenesis: predict expression with reference allele, predict with alternative allele, and compute the difference. Because the model never trains on variant data, predictions are unconfounded by linkage disequilibrium, a fundamental advantage over statistical eQTL approaches.

6.4.1. Computing Variant Effects

For any variant, ExPecto computes effects by comparing predictions:

$$\Delta\text{expression} = f(\text{sequence}_{\text{alt}}) - f(\text{sequence}_{\text{ref}})$$

This approach predicts the direction and magnitude of expression change in each of 218 tissues for any single nucleotide variant within the 40 kb promoter region.

6.4.2. eQTL Validation

ExPecto correctly predicted the direction of expression change for 92% of the top 500 strongest-effect GTEx eQTL variants. Prediction accuracy increases with predicted effect magnitude: variants with stronger predicted effects show higher eQTL direction concordance, consistent with the expectation that true causal variants should have larger predicted effects.

Unlike traditional eQTL studies, which are biased toward common variants with sufficient statistical power, ExPecto predictions work equally well across the allele frequency spectrum. This makes the framework particularly valuable for rare variant interpretation where population data is sparse.

6.4.3. Advantages Over eQTL Mapping

Traditional eQTL studies face fundamental limitations. Linkage disequilibrium confounds causal inference: only 3.5 to 11.7% of GTEx lead variants are estimated to be truly causal, meaning fewer than 1% of all reported eQTL variants directly affect expression. Allele frequency creates power imbalances, as rare variants lack the sample sizes required for detection. Tissue availability constrains what can be studied, since eQTL mapping requires large sample sizes in the tissue of interest.

ExPecto's sequence-based predictions sidestep all three limitations. The model scores variants based on predicted functional impact rather than population associations, works identically for any allele frequency, and leverages expression training data from many tissues even when eQTL data is unavailable.

6.5. GWAS Causal Variant Prioritization

A major application of ExPecto is prioritizing causal variants within GWAS-identified loci, where LD typically prevents identification of the true functional variant.

6.5.1. Systematic Prioritization

Zhou et al. applied ExPecto to prioritize variants from approximately 3,000 GWAS studies. GWAS loci with stronger predicted effect variants were significantly more likely to replicate in independent studies ($p = 6.3\times10^{-1}$, Wald test with logistic regression). Stronger predicted effect variants were also more likely to be the exact replicated variant ($p = 5.6\times10^{-1}$).

The framework can identify causal variants that statistical association alone cannot distinguish. For example, an early venous thromboembolism GWAS identified rs3756008 as the lead variant near the F11 locus. ExPecto prioritized a different LD variant, rs4253399, which was subsequently discovered as the true association in a larger cohort study.

6.5.2. Experimental Validation

The authors experimentally validated three top-ranked ExPecto predictions for immune-related diseases using luciferase reporter assays. In all cases, the ExPecto-prioritized variants showed significant allele-specific regulatory activity, while the original GWAS lead variants showed no differential activity.

Disease	ExPecto-Prioritized SNP	Gene	Reporter Effect	p-value	GWAS Lead SNP
Crohn's disease / IBD	rs1174815	IRGM	Decreased expression	3×10	Not significant
Behçet's disease	rs147398495	CCR1	Changed activity	7×10^{-1}	Not significant
Chronic HBV infection	rs381218	HLA-DOA	4-fold change	1×10	Not significant

ExPecto correctly predicted the direction of expression change for all three validated variants. These results demonstrate that sequence-based expression models can identify functional variants that statistical association studies cannot distinguish from linked non-functional variants.

6.6. In Silico Saturation Mutagenesis

The computational efficiency of ExPecto enables exhaustive characterization of the regulatory mutation space. The authors computed predicted effects for all possible single nucleotide substitutions within ±1 kb of each TSS, covering over 140 million mutations across 23,779 human Pol II-transcribed genes. This identified more than 1.1 million mutations with strong predicted expression effects.

6.6.1. Variation Potential

For each gene, the comprehensive mutagenesis profile defines its "variation potential" (VP), the collective effects of all possible mutations on that gene's expression. VP reflects the regulatory sensitivity of each gene. Genes with high VP have expression that is easily perturbed by sequence changes, with regulatory regions densely packed with functional elements. Genes with low VP show expression robust to mutations, potentially indicating fewer regulatory constraints or more redundant regulatory architecture.

VP correlates with known biological properties: tissue-specific genes show lower VP than broadly expressed genes, and genes under stronger evolutionary constraint tend to have higher VP.

6.6.2. Constraint Violation Scores

By comparing predicted mutational effects to observed population variation, ExPecto enables inference of evolutionary constraints. A "constraint violation score" measures whether observed variants push expression in the "wrong" direction relative to inferred evolutionary constraint. Genes with negative VP directionality (mutations tend to reduce expression) are typically actively expressed, where loss-of-function mutations are deleterious. Genes with positive VP directionality (mutations tend to increase expression) are typically repressed, where gain-of-expression mutations are deleterious.

This framework successfully predicts GWAS risk alleles without any prior variant-disease association data. Positive violation scores are significantly associated with alternative alleles being risk alleles ($p = 0.002$, Wilcoxon rank sum test, AUC = 0.67), demonstrating potential for ab initio disease variant identification.

6.7. The 40 kb Regulatory Window

ExPecto's ±20 kb window around each TSS represents an empirically optimized trade-off. Smaller windows decreased prediction performance, while larger windows (50 to 200 kb) showed negligible performance improvement.

This suggests that most regulatory information for promoter-proximal expression lies within 40 kb of the TSS, at least within the linear modeling framework employed by ExPecto. Distal enhancers beyond this window, while biologically important, likely require more sophisticated integration approaches to capture. Enformer (Chapter 11), with its 200 kb effective receptive field, addresses this limitation.

6.8. Relationship to the DeepSEA Lineage

ExPecto represents a conceptual extension of the DeepSEA framework. DeepSEA (2015) predicts 919 chromatin profiles from a 1 kb context window. ExPecto/Beluga (2018) predicts gene expression across 218 tissues from a 40 kb context window. Sei (2022) predicts 21,907 chromatin profiles plus sequence classes from a 4 kb window.

Model	Year	Primary Output	Context Window
DeepSEA	2015	919 chromatin profiles	1 kb
ExPecto/Beluga	2018	Gene expression (218 tissues)	40 kb
Sei	2022	21,907 chromatin profiles + sequence classes	4 kb

While DeepSEA predicts regulatory intermediate phenotypes, ExPecto predicts the downstream transcriptional consequence. For GWAS variant prioritization, ExPecto predictions proved more effective than DeepSEA alone. Variants may alter chromatin features without affecting expression, but expression effects are more directly tied to phenotypic consequences.

The chromatin prediction component of ExPecto (Beluga) became the foundation for Sei (discussed in Chapter 5), which expanded chromatin targets to 21,907 profiles and introduced sequence class annotations for interpretability.

6.9. Limitations and Considerations

6.9.1. Linear Expression Model

While the chromatin CNN captures nonlinear sequence patterns, the final expression model is linear. This prevents modeling of complex regulatory logic such as synergistic interactions between elements, competitive binding or mutual exclusion, and threshold effects where element contributions are context-dependent. The choice was pragmatic: linear models require less data and offer interpretability, but may sacrifice predictive power for genes with complex regulatory logic.

6.9.2. Context Window Constraints

The 40 kb promoter-proximal window misses distal enhancers operating over hundreds of kilobases, three-dimensional chromatin interactions that bring distant elements into proximity, and enhancer-promoter specificity (which enhancer regulates which gene among nearby alternatives).

6.9.3. TSS-Centric Framework

ExPecto requires a defined TSS for each gene, potentially limiting predictions for genes with multiple alternative promoters, novel or unannotated transcription start sites, and tissue-specific promoter usage.

6.9.4. Training Data Biases

Expression models trained on GTEx, Roadmap, and ENCODE data inherit their biases. These include ancestry composition (GTEx is primarily European), tissue representation (some tissues well-covered, others sparse), and cell line artifacts (immortalized cells may not reflect primary tissue biology).

6.10. Significance for the Field

ExPecto established several paradigms that influenced subsequent genomic deep learning.

The modular sequence-to-expression prediction architecture demonstrated the value of decomposing the problem into chromatin prediction, spatial integration, and expression modeling. This separation enables interpretability and component-wise improvement.

Ab initio variant effect prediction, achieved by training without variant data, avoids LD confounding and enables causal inference rather than mere association. This principle carries forward to later expression and variant effect models.

Scalable in silico mutagenesis showed that computational efficiency enables exhaustive characterization of mutational effects at genome scale, a capability that would be impossible experimentally.

The framework's tissue-specific regulatory learning demonstrated that models can learn tissue-relevant regulatory features without explicit tissue labels for chromatin inputs, relying instead on the structure of expression data.

Finally, the experimental validation standard set by ExPecto, demonstrating functional validation of computational predictions with reporter assays, established expectations for the field.

The framework demonstrated that deep learning could move beyond predicting intermediate molecular phenotypes (chromatin state) to predict cellular phenotypes (expression levels) directly from sequence. This progression from sequence to chromatin to expression to disease prefigured the increasingly ambitious goals of later genomic foundation models.

ExPecto's public web portal (http://hb.flatironinstitute.org/expecto) and code release (https://github.com/FunctionLab/ExPecto) maintained the field's norm of open tool availability established by DeepSEA. The framework continues to serve as a baseline for expression prediction methods and as a component in variant prioritization pipelines.

7. Splicing Prediction

> ⚠ Warning
>
> **TODO:**
>
> - Verify all citations are in bibliography
> - Consider adding figure showing SpliceAI architecture diagram

7.1. The Splicing Challenge

While DeepSEA and ExPecto (Chapter 5; Chapter 6) addressed chromatin state and gene expression, a distinct class of functional variants operates through a different mechanism: disruption of pre-mRNA splicing. The spliceosome achieves remarkable precision, recognizing the correct splice sites among millions of potential candidates in the human transcriptome. Yet the sequence determinants underlying this specificity remained incompletely understood, limiting interpretation of variants that might alter splicing.

The clinical stakes are substantial. As discussed in Chapter 1, variant calling pipelines identify thousands of variants per exome and millions per genome, but annotation frameworks traditionally focus on coding consequences. Variants affecting splicing outside the canonical GT/AG dinucleotides are systematically underascertained, even though splice-disrupting mutations are a major mechanism of Mendelian disease. The ACMG/AMP guidelines (Chapter 2) recognize splicing evidence as supporting pathogenicity, but until recently, computational tools lacked the accuracy to identify cryptic splice variants reliably.

SpliceAI, introduced by Jaganathan et al. in 2019, demonstrated that deep neural networks could learn the sequence rules governing splicing with near-spliceosomal precision (Jaganathan et al. 2019). The model predicts splice site locations directly from pre-mRNA sequence, enabling identification of "cryptic splice" variants that create novel splice sites or disrupt existing ones in ways that evade traditional annotation-based detection.

This chapter examines SpliceAI as the culmination of Part II's CNN-based sequence-to-function models. Where Chapter 5 focused on chromatin accessibility and transcription factor binding, and Chapter 6 extended to gene expression, SpliceAI targets a specific post-transcriptional mechanism with direct clinical relevance. The model illustrates how deep learning can achieve near-expert performance on a well-defined biological problem while revealing mechanistic insights about the underlying biology.

7.2. Prior Approaches and Limitations

Before SpliceAI, splice site prediction relied on methods with limited sequence context, a constraint we have seen repeatedly limit earlier genomic models (Chapter 5; Chapter 6).

MaxEntScan models core splice motifs using maximum entropy, limited to approximately 9 bp context around donor/acceptor sites (**yeo_maxentscan_2004?**). **GeneSplicer** combines Markov models with decision trees. **NNSplice** represents an early neural network approach with narrow receptive fields. These methods captured the essential GT (donor) and AG (acceptor) dinucleotides and surrounding consensus sequences, but could not model the long-range determinants that contribute to splicing specificity.

The limitations parallel those of pre-deep-learning variant effect predictors like CADD (Chapter 4), which aggregate many annotation features but lack the capacity to learn complex sequence dependencies. Just as CADD's logistic regression cannot capture the combinatorial logic of regulatory grammar, MaxEntScan's position weight matrices cannot represent the spatial relationships between distant splicing determinants.

These constraints had practical consequences. Prior methods produced many false positive predictions and missed variants acting through distal mechanisms. A variant that weakens a splice site may only cause pathogenic mis-splicing if no nearby cryptic site can compensate, a judgment requiring integration of information across thousands of nucleotides.

7.3. The SpliceAI Architecture

SpliceAI employs an ultra-deep residual convolutional network that integrates information across 10,000 nucleotides of sequence context. This represents an order of magnitude expansion beyond prior methods and reflects the same architectural intuition that motivated ExPecto's 40 kb regulatory window (Chapter 6): functional genomic predictions often require long-range context that shallow models cannot capture.

7.3.1. Input Representation

Like DeepSEA and ExPecto, SpliceAI uses one-hot encoded nucleotide sequences as input. The four nucleotides (A, C, G, T) are encoded as binary vectors, with no hand-crafted features or annotations. This end-to-end learning approach forces the network to discover relevant sequence patterns from training data rather than relying on prior biological knowledge.

The input window spans 10,000 nucleotides (5,000 on each side of the position of interest), providing context for recognizing distant determinants like branch points, exonic splicing enhancers, and intron/exon length constraints.

7.3.2. Residual Block Design

The architecture's fundamental unit is the residual block, comprising batch normalization, ReLU activation, and dilated convolutions. Residual connections address the vanishing gradient problem that had limited earlier deep networks:

$$\text{output} = \text{input} + F(\text{input})$$

where F represents the transformation learned by the convolutional layers. This design enables training of networks with 32 layers, far deeper than the 3-layer DeepSEA or 6-layer ExPecto/Beluga architectures (Chapter 5; Chapter 6).

Skip connections from every fourth residual block feed directly to the penultimate layer, accelerating training convergence and enabling gradient flow through the full network depth.

7.3.3. Dilated Convolutions for Long-Range Context

Standard convolutions with small kernels (e.g., 3-8 bp) would require many layers to achieve a 10,000 bp receptive field, making training prohibitively expensive. SpliceAI uses dilated convolutions that exponentially expand the receptive field while maintaining computational efficiency.

A dilated convolution with dilation rate d samples input positions at intervals of d rather than consecutively. By stacking convolutions with increasing dilation rates, the network can efficiently integrate information across the full 10 kb window while maintaining sensitivity to local motif patterns.

This approach represents an architectural innovation beyond the standard convolutional designs in DeepSEA and ExPecto. Later models in Part III, including long-context transformers and hybrid architectures (Chapter 11), would explore alternative solutions to the long-range dependency problem.

7.3.4. Output Predictions

For each position in the pre-mRNA sequence, SpliceAI outputs three probabilities summing to one: the probability of being a splice acceptor, splice donor, or neither. This per-position classification enables fine-grained predictions across entire transcripts.

7.4. Training and Evaluation

7.4.1. Training Data

SpliceAI was trained on GENCODE V24 annotations (Chapter 2), using 20,287 protein-coding genes with principal transcripts selected when multiple isoforms existed. The training/test split used odd versus even chromosomes:

Set	Chromosomes	Genes	Donor-Acceptor Pairs
Training	2, 4, 6, 8, 10-22, X, Y	13,384	130,796
Testing	1, 3, 5, 7, 9	1,652	14,289

Critically, genes with paralogs on training chromosomes were excluded from the test set. This prevents information leakage through sequence homology, a form of data leakage we will examine systematically in Chapter 16. Paralog exclusion is particularly important for splicing models because conserved gene families often share splice site architecture.

For variant effect prediction, training was augmented with novel splice junctions commonly observed in GTEx RNA-seq data (Chapter 2), adding approximately 67,000 donor and 63,000 acceptor annotations. This augmentation improved sensitivity for detecting splice-altering variants, particularly in deep intronic regions where GENCODE annotations are incomplete.

7.4.2. Splice Site Prediction Performance

SpliceAI-10k achieved remarkable accuracy:

Metric	SpliceAI-10k	MaxEntScan
Top-k accuracy	95%	57%
PR-AUC	0.98	—

Top-k accuracy measures the fraction of correctly predicted splice sites at the threshold where predicted sites equal actual sites. The dramatic improvement reflects SpliceAI's ability to reject false positive splice sites by considering sequence context beyond the core motif.

Even complex genes exceeding 100 kb, such as CFTR, are often reconstructed perfectly to nucleotide precision. When tested on long noncoding RNAs (which lack protein-coding selective pressures on sequence composition), the network achieved 84% top-k accuracy, confirming it learned genuine splicing determinants rather than artifacts of coding sequence.

7.4.3. Context Length Matters

Performance improved substantially with context length across SpliceAI variants:

Model	Context (each side)	Top-k Accuracy	PR-AUC
SpliceAI-80nt	40 bp	—	0.87
SpliceAI-400nt	200 bp	—	0.93
SpliceAI-2k	1,000 bp	—	0.96
SpliceAI-10k	5,000 bp	95%	0.98

This progression confirms that distal sequence features thousands of nucleotides from splice sites contribute meaningfully to splicing decisions. The pattern echoes findings from ExPecto (Chapter 6), where expression prediction improved with wider regulatory windows, and anticipates the even longer contexts explored by transformer models in Part III.

7.5. Variant Effect Prediction

7.5.1. The Delta Score

SpliceAI predicts variant effects by comparing splice site predictions for reference and alternative sequences:

$$\Delta\text{score} = \max_{|p-v|\leq 50} |P_{\text{alt}}(p) - P_{\text{ref}}(p)|$$

where v is the variant position and p ranges over positions within 50 bp of the variant. The maximum change across all positions captures variants that strengthen existing sites, weaken existing sites, or create entirely new splice sites.

Critically, the model was trained only on reference transcript sequences and splice junction annotations. It never saw variant data during training. Variant effect prediction is thus a challenging test of whether the network learned genuine sequence determinants of splicing, analogous to how ExPecto's variant effect predictions emerged from learning sequence-expression relationships without any variant labels (Chapter 6).

This ab initio approach to variant effect prediction represents a fundamental advantage over annotation-based scores like CADD (Chapter 4), which incorporate variant-level training data and may learn associations rather than causal sequence-function relationships.

7.5.2. Cryptic Splice Variant Classes

SpliceAI detects several classes of splice-altering variants:

Donor/acceptor loss: Disruption of annotated splice sites, either through direct mutation of the GT/AG dinucleotides or through weakening of flanking enhancer sequences.

Donor/acceptor gain: Creation of novel splice sites that compete with canonical sites. These can occur within exons (causing partial exon loss) or within introns (causing inclusion of intronic sequence).

Exon skipping: Variants that weaken a splice site sufficiently that the spliceosome skips the entire exon, joining flanking exons directly.

Intron retention: Variants causing failure to recognize either the donor or acceptor site of an intron, leaving intronic sequence in mature mRNA.

Cryptic exon activation: Deep intronic variants that create both a novel donor and acceptor, activating a "pseudoexon" that inserts into the mature transcript.

Traditional annotation-based methods can identify variants in the essential GT/AG dinucleotides but miss the broader landscape of cryptic splice variants operating through more subtle mechanisms.

7.5.3. RNA-seq Validation

The authors validated predictions using GTEx RNA-seq data from 149 individuals with matched whole-genome sequencing (Chapter 2). Focusing on rare, private mutations (present in only one GTEx individual), they found:

Private mutations predicted to have functional consequences were strongly enriched at private novel splice junctions and at boundaries of skipped exons. Confidently predicted cryptic splice variants (score 0.5) validated at three-quarters the rate of essential GT/AG splice disruptions.

Both validation rate and effect size tracked closely with scores:

Score Threshold	Validation Rate
0.2	~50%
0.5	~75%
0.8	~85%

Validated variants, especially those with lower scores, often showed incomplete penetrance, producing a mixture of aberrant and normal transcripts. This partial effect distinguishes cryptic splice variants from essential GT/AG disruptions and has implications for clinical interpretation.

7.5.4. Population Genetics Evidence

Complementing RNA-seq validation, the authors examined allele frequency spectra in gnomAD (Chapter 2). If predicted cryptic splice variants are truly deleterious, they should be depleted at common allele frequencies relative to neutral variants.

Predicted cryptic splice variants (score 0.8) showed 78% depletion at common frequencies compared to singletons, nearly matching the 82% depletion of frameshift, stop-gain, and essential splice disruptions. This population genetics signature provides orthogonal evidence that predictions identify genuinely functional variants.

The analysis revealed an important nuance: deep intronic variants (>50 nt from exons) showed only 56% depletion, consistent with the lower validation rates observed in RNA-seq data for this category. Predictions farther from annotated exons are more challenging, possibly because deep intronic regions contain fewer of the specificity determinants that have been selected to be present near exons.

7.5.5. Rare Variant Burden

The average human genome carries approximately:

- **11** rare protein-truncating variants (allele frequency <0.1%)
- **5** rare functional cryptic splice variants

Cryptic splice variants outnumber essential GT/AG splice-disrupting variants roughly 2:1, highlighting the substantial mutational target space beyond canonical splice sites. These numbers provide context for clinical interpretation: cryptic splice variants are not exotic curiosities but a common class of potentially pathogenic variation.

7.6. De Novo Mutations in Rare Disease

The central clinical finding of SpliceAI is that cryptic splice mutations constitute a major, previously underappreciated cause of rare genetic disorders. This analysis connects SpliceAI predictions to the clinical variant interpretation frameworks discussed in Chapter 2 and previews the pathogenic variant discovery workflows examined in Chapter 19.

7.6.1. Case-Control Analysis

The authors analyzed de novo mutations in:

- **4,293** individuals with intellectual disability (Deciphering Developmental Disorders cohort)
- **3,953** individuals with autism spectrum disorders (Simons Simplex Collection + Autism Sequencing Consortium)
- **2,073** unaffected sibling controls

De novo mutations predicted to disrupt splicing (Δ 0.1) were significantly enriched in affected individuals:

Cohort	Enrichment vs. Controls	p-value
Intellectual disability (DDD)	1.51-fold	4.2×10
Autism spectrum disorder	1.30-fold	0.020

The enrichment remained significant when restricting to synonymous and intronic mutations, excluding the possibility that results were driven solely by variants with dual protein-coding and splicing effects. This confirms that the splicing mechanism itself, not merely correlation with coding effects, drives disease association.

7.6.2. Fraction of Pathogenic Mutations

Based on the excess of de novo mutations in cases versus controls:

- **9%** of pathogenic de novo mutations in intellectual disability act through cryptic splicing
- **11%** of pathogenic de novo mutations in autism act through cryptic splicing

In absolute terms, approximately 250 cases across the cohorts could be explained by de novo cryptic splice mutations, compared to approximately 909 cases explained by de novo protein-truncating variants.

7.6.3. Clinical Penetrance

Cryptic splice mutations showed roughly 50% of the clinical penetrance of classic protein-truncating mutations (stop-gain, frameshift, essential splice). This reduced penetrance reflects that many cryptic splice variants are hypomorphic, producing a mixture of normal and aberrant transcripts rather than complete loss of function.

Well-characterized examples from Mendelian disease support this interpretation: the c.315-48T>C variant in FECH and c.-32-13T>G in GAA are both hypomorphic cryptic splice alleles associated with milder phenotype or later age of onset.

The reduced penetrance of cryptic splice mutations has implications for clinical interpretation. Variants with moderate Δ scores may contribute to disease risk without being fully penetrant, and the same variant may have different consequences depending on the expression level and splicing factor context in relevant tissues.

7.6.4. Novel Gene Discovery

Including cryptic splice mutations in gene discovery analyses identified:

- **5 additional** candidate genes for intellectual disability
- **2 additional** candidate genes for autism

These genes would have fallen below the discovery threshold (FDR <0.01) when considering only protein-coding mutations. This finding suggests that systematic inclusion of splice predictions in gene burden analyses could accelerate discovery of disease genes with splicing-mediated pathogenic mechanisms.

7.6.5. Experimental Validation

RNA-seq validation in lymphoblastoid cell lines from autism patients confirmed predictions in 21 of 28 cases (75% validation rate). Among confirmed cases:

- 9 showed novel junction creation
- 8 showed exon skipping
- 4 showed intron retention

Seven cases did not show aberrant splicing in lymphoblastoid cells despite adequate transcript expression. These may represent tissue-specific effects not observable in the available cell type, highlighting a limitation of using accessible tissues for validation.

7.7. What SpliceAI Learned

Beyond prediction accuracy, SpliceAI provides insights into splicing mechanisms through interpretability analyses. These findings demonstrate that deep learning models can serve as hypothesis-generating tools for understanding biology, not merely black-box classifiers.

7.7.1. Long-Range Specificity Determinants

Comparison of models trained on different context lengths revealed that apparent "degeneracy" in splice motifs (the observation that many sequences matching consensus motifs are not used as splice sites) is explained by long-range determinants:

The 80 bp context model (SpliceAI-80nt) assigned lower scores to splice sites flanking average-length exons and introns, favoring sites adjacent to unusually short or long elements. This reflects that such unusual sites tend to have stronger local motifs to compensate for suboptimal geometry.

The 10 kb context model (SpliceAI-10k) showed the opposite pattern, preferring splice sites flanking typical-length elements. With access to full context, the model learned that typical geometries provide favorable long-range determinants that compensate for weaker local motifs.

This finding resolves a longstanding puzzle in splicing biology: why do most functional splice sites have degenerate motifs when consensus sequences can be much more information-rich? The answer is that motif strength is only one component of recognition; geometric and contextual features provide additional specificity that makes strong motifs unnecessary.

7.7.2. Branch Point Recognition

In silico mutagenesis experiments confirmed that SpliceAI learned canonical splicing elements without explicit training on them. Introducing the optimal branch point sequence (TACTAAC) at various distances from splice acceptors increased predicted splice strength specifically when placed 20-45 nucleotides upstream, matching the known functional range for branch points in mammals.

At distances less than 20 nucleotides, the branch point sequence disrupted the polypyrimidine tract, decreasing predicted acceptor strength. This context-dependent effect demonstrates that SpliceAI learned the spatial relationships between splicing elements, not just their individual contributions.

7.7.3. Exonic Splicing Enhancers

The SR-protein binding motif GAAGAA, introduced at various positions, enhanced splice site strength when placed in expected locations within exons. This confirms that SpliceAI learned the contribution of exonic splicing enhancers to splice site recognition.

7.7.4. Nucleosome Positioning

Novel exon-creation events (where variants activate cryptic exons in introns) were significantly associated with existing nucleosome positioning ($p = 0.006$ by permutation test), even in cell lines that likely lack the corresponding genetic mutations. This supports a causal role for nucleosome occupancy in exon definition and demonstrates that SpliceAI implicitly captures chromatin-related effects despite not being trained on chromatin data.

The association with nucleosome positioning connects to the chromatin-centric models in Chapter 5 and Chapter 6, suggesting that splicing and chromatin state are not independent regulatory layers but interact in ways that sequence models can detect.

7.8. Relationship to Other Sequence-to-Function Models

SpliceAI fits within the broader trajectory of CNN-based genomic models while illustrating the value of task-specific architectures.

7.8.1. Comparison to DeepSEA and ExPecto

Feature	DeepSEA	ExPecto	SpliceAI
Primary task	Chromatin state	Gene expression	Splice sites
Context window	1 kb	40 kb	10 kb
Architecture depth	3 layers	6 layers	32 layers
Output type	Multi-label classification	Regression	Per-position classification
Training data	ENCODE/Roadmap	GTEx expression	GENCODE annotations
Variant interpretation	Allelic imbalance	Expression effect	Δ score

SpliceAI shares the one-hot encoding and convolutional architecture of its predecessors but introduces residual connections, dilated convolutions, and much greater depth. These architectural innovations enable effective training on a narrow, well-defined task while achieving substantially higher accuracy than would be possible with shallower networks.

7.8.2. Task Specificity vs. Foundation Models

SpliceAI represents a different design philosophy from the foundation model approach explored in Parts III and IV. Rather than learning general sequence representations that transfer across tasks, SpliceAI focuses computational capacity on a single problem: predicting splice sites.

This specialization has both advantages and limitations. SpliceAI achieves remarkable accuracy on its target task, but its representations do not obviously transfer to other problems. Later chapters will explore whether self-supervised foundation models can match task-specific performance while providing broader utility (Chapter 10; Chapter 12).

The tension between specialized and general-purpose models remains unresolved. For clinical applications requiring high accuracy on specific tasks, specialized models like SpliceAI may remain preferred. For discovery applications requiring broad coverage of molecular mechanisms, foundation models may prove more valuable.

7.8.3. Integration with Variant Interpretation Pipelines

SpliceAI scores can be incorporated into variant interpretation workflows alongside other evidence types discussed in Chapter 4 and Chapter 13:

Complementarity with CADD: While CADD provides genome-wide deleteriousness scores incorporating many feature types, SpliceAI offers more accurate predictions for the specific mechanism of splice disruption. In clinical practice, both scores contribute independent evidence.

Integration with expression models: ExPecto predicts expression effects of variants but does not model splicing directly. A variant might have minimal ExPecto effect but high SpliceAI Δ score if it disrupts splicing without affecting transcription. Combining predictions provides a more complete view of functional consequences.

ClinVar evidence: SpliceAI Δ scores can support splicing evidence in ACMG/AMP classification, particularly for variants outside canonical splice sites where experimental evidence is often lacking.

7.9. Limitations and Considerations

7.9.1. Tissue Specificity

SpliceAI predicts splice sites based on sequence alone, without modeling tissue-specific alternative splicing. The same variant may have different effects across tissues depending on the expression of splicing factors and regulatory RNAs. Tissue-specific models trained on RNA-seq annotations could address this limitation but would require careful handling of annotation completeness across tissues.

7.9.2. Incomplete Penetrance

Many cryptic splice variants produce partial shifts in splicing (alternative splicing) rather than complete disruption. The Δ score correlates with penetrance, but precise quantification of isoform ratios requires experimental validation. For clinical interpretation, incomplete penetrance may lead to variable expressivity and reduced disease severity.

7.9.3. Deep Intronic Predictions

While SpliceAI substantially improves deep intronic variant prediction over prior methods, sensitivity remains lower than for variants near exons. The 41% sensitivity (Δ 0.5) in deep intronic regions suggests that additional sequence features beyond the 10 kb context, or mechanisms not captured by the model, may contribute to splicing in these regions.

7.9.4. Training on Canonical Transcripts

Training on principal transcripts may not fully capture the diversity of alternative splicing. Augmentation with RNA-seq-derived junctions improved performance, suggesting that expanded training data, including tissue-specific annotations, could further enhance predictions.

7.9.5. Evaluation Circularity

As with all variant effect predictors, evaluation faces potential circularity (Chapter 16). If SpliceAI predictions influence variant classification in clinical databases, using those databases to benchmark newer models creates inflated performance estimates. The RNA-seq and population genetics validations provide more independent assessments but are not immune to selection biases.

7.10. Significance for the Field

SpliceAI established several paradigms that influenced subsequent genomic deep learning:

Clinical impact quantification: The estimate that 9-11% of pathogenic mutations act through cryptic splicing fundamentally changed understanding of the noncoding disease mutation landscape. This finding has practical implications for diagnostic yield: including splice predictions in clinical analysis can identify diagnoses that would otherwise be missed.

Deep context matters: The 32-layer, 10 kb context architecture demonstrated that splicing involves long-range sequence integration, motivating similar approaches in other genomic prediction tasks. This insight carries forward to the transformer and hybrid models in Part III.

Genome-wide variant scoring: Precomputed Δ scores for all possible single nucleotide substitutions (available at https://github.com/Illumina/SpliceAI) enable routine clinical annotation without requiring per-variant model inference. This resource has been widely adopted in clinical and research pipelines.

Validation standards: The combination of RNA-seq validation, population genetics evidence, and case-control analysis established a rigorous framework for evaluating variant effect predictors. This multi-modal validation approach sets expectations for how genomic deep learning models should demonstrate biological validity.

Mechanistic interpretability: The in silico mutagenesis experiments showing learned recognition of branch points, exonic enhancers, and nucleosome positioning demonstrate that deep learning models can provide biological insights, not just predictions. This finding motivates the interpretability approaches examined in Chapter 17.

SpliceAI has become a standard component of clinical variant interpretation pipelines, complementing protein-effect predictors and regulatory variant scores. Its success illustrates both the power of task-specific deep learning and the value of rigorous biological validation for establishing clinical utility.

7.11. Summary

This chapter examined SpliceAI as the culmination of Part II's CNN-based sequence-to-function models. Key themes include:

Architectural progression: From DeepSEA's 3-layer, 1 kb model to SpliceAI's 32-layer, 10 kb model, increasing depth and context enable capture of long-range biological dependencies.

Task-specific versus general models: SpliceAI's focus on a single, well-defined prediction task enables remarkable accuracy, illustrating the trade-off between specialization and generality that Part III will explore through foundation models.

Clinical translation: The disease association analyses demonstrate how sequence models can quantify clinical impact, estimate diagnostic yield, and guide resource allocation in rare disease research.

Mechanistic insight: Deep learning models trained on biological data can reveal genuine biological mechanisms, not just correlations, when combined with careful interpretability analyses.

Validation rigor: Multiple orthogonal validation approaches (RNA-seq, population genetics, case-control) establish confidence that model predictions reflect true biological effects.

Part III now turns to transformer-based architectures and self-supervised learning, approaches that aim to learn general-purpose sequence representations applicable across many tasks rather than optimizing for single prediction targets.

Part III.

Part III: Transformers Models

8. Sequence Representation & Tokens

> ⚠ Warning
>
> **TODO:**
>
> - Verify tradeoffs and general concensus discussion is sufficient
> - ...

8.1. From Sequence to Model: The Representation Problem

Every genomic deep learning model must answer a fundamental question before learning can begin: how should DNA sequence be represented as numerical input? This question might seem purely technical, a preprocessing detail to be settled and forgotten. Yet the choice of representation profoundly shapes what a model can learn, how efficiently it trains, and what biological phenomena it can capture. The previous chapters employed one-hot encoding without much discussion, treating it as the obvious default for CNN-based architectures like DeepSEA (Chapter 5) and SpliceAI (Chapter 7). This approach worked remarkably well for those models, but the emergence of transformer-based language models introduced new considerations around tokenization, vocabulary design, and the fundamental trade-offs between sequence compression and resolution.

The challenge can be understood through an analogy to natural language processing. When training a language model on English text, researchers must decide how to segment the continuous stream of characters into discrete tokens. One could treat each character as a token, preserving maximum resolution but creating very long sequences. Alternatively, one could use words as tokens, compressing the sequence but potentially losing information about word structure. Or one could learn a vocabulary of subword units that balances these concerns. Each choice affects what patterns the model can discover and how efficiently it can process long documents.

DNA presents similar choices but with important differences. The genome has only four letters rather than dozens, no natural word boundaries, and biological structure that operates at multiple scales simultaneously. A transcription factor binding site might span 6-12 nucleotides, but the regulatory grammar linking multiple binding sites can extend over hundreds of base pairs. Coding sequences follow a strict three-nucleotide codon structure, while noncoding regions have no such constraint. Any representation scheme must navigate these biological realities while remaining computationally tractable.

This chapter examines the evolution of sequence representation strategies in genomic deep learning. We trace the progression from one-hot encoding through k-mer tokenization to modern approaches including Byte Pair Encoding, single-nucleotide tokens, and biologically-informed tokenization schemes. Understanding these choices clarifies design decisions in models throughout Parts III

and IV, and illuminates why seemingly minor representation choices can dramatically affect model capabilities.

8.2. One-Hot Encoding: The CNN Foundation

One-hot encoding represents the simplest possible approach to sequence representation: each nucleotide becomes a sparse binary vector with a single active element indicating its identity. Adenine is encoded as [1, 0, 0, 0], cytosine as [0, 1, 0, 0], guanine as [0, 0, 1, 0], and thymine as [0, 0, 0, 1]. A sequence of length L thus becomes a matrix of dimensions $4 \times L$, interpretable as four channels analogous to the RGB channels of an image plus one additional channel.

This representation dominated the CNN era of genomic deep learning for good reason. One-hot encoding is lossless, preserving every nucleotide explicitly without any information compression. It maintains single-nucleotide resolution, enabling detection of effects from individual SNPs, which is critical for variant interpretation. The representation exhibits translation equivariance, meaning that convolutional filters learn position-invariant motifs that can be recognized anywhere in the sequence. And it requires no preprocessing, vocabulary construction, or tokenizer training, making implementation straightforward.

DeepSEA, ExPecto, and SpliceAI all employed one-hot encoding without modification. The convolutional layers in these models learned to detect sequence patterns directly from the binary representation, with first-layer filters discovering motifs corresponding to transcription factor binding sites and deeper layers capturing combinations and spatial arrangements. The representation worked because CNNs process sequences through local operations, with each convolutional filter examining only a small window of positions at a time. The sparse, orthogonal nature of one-hot vectors posed no obstacle to this local processing.

Yet for transformer architectures, one-hot encoding presents significant challenges. Transformers compute attention between all pairs of positions in a sequence, with computational cost scaling as $O(L^2)$ where L is the sequence length. A 10 kb sequence requires 10,000 tokens, meaning 100 million pairwise attention computations per layer. This quickly becomes prohibitive for the long sequences that genomic applications require. Furthermore, transformers typically learn dense embeddings for each token, but with only four possible nucleotides, there is little opportunity for the model to discover rich representations through the embedding layer. The sparse one-hot vectors provide minimal information for the embedding to transform. Most critically, practical transformer context windows of 512 to 4,096 tokens translate to only 512 to 4,096 base pairs when using one-hot encoding, a tiny fraction of genes or regulatory regions and far less than the context that proved valuable for models like Enformer and SpliceAI.

These limitations motivated the search for alternative representations that could compress genomic sequences into fewer tokens while preserving the information needed for biological prediction.

8.3. K-mer Tokenization: The DNABERT Approach

K-mer tokenization treats overlapping subsequences of length k as tokens, drawing an analogy between k-mers and words in natural language. Just as sentences are composed of words that carry meaning through their sequence and combination, genomic sequences might be understood as

composed of k-mer "words" that encode biological function through their arrangement. DNABERT (2021) pioneered this approach for genomic transformers, using 6-mers as tokens and training a BERT-style masked language model on human reference sequences (Ji et al. 2021).

The k-mer vocabulary has a fixed size of 4^k possible tokens. For 6-mers, this yields 4,096 distinct tokens, comparable to the vocabulary sizes used in some natural language models. Each token represents six consecutive nucleotides, creating a direct correspondence between subsequence and token identity. The tokenization proceeds by sliding a window across the sequence and recording each k-mer encountered.

DNABERT used overlapping k-mers, meaning that for a sequence like ACGTACGT, the 6-mer tokens would share five nucleotides with their neighbors. The sequence position advances by one nucleotide at a time, generating one token per position (minus the k-1 positions at the end where a complete k-mer cannot be formed). This overlapping design preserves positional information and ensures that every nucleotide contributes to multiple tokens, potentially providing redundancy that helps the model learn robust representations.

The DNABERT approach provided valuable proof of concept. It demonstrated that self-supervised pretraining on raw DNA sequences could improve performance over training from scratch, that learned embeddings could capture biologically meaningful regularities even when trained only on the reference genome, and that BERT-style architectures could be reused across multiple downstream tasks. DNABERT achieved state-of-the-art performance on prediction of promoters, splice sites, and transcription factor binding sites after fine-tuning with relatively small amounts of task-specific labeled data.

However, subsequent analysis revealed fundamental limitations of k-mer tokenization that stemmed from the overlapping design. DNABERT-2 (2024) articulated these problems clearly (Z. Zhou et al. 2024). First, overlapping k-mers provide no sequence compression. The number of tokens equals the number of nucleotides (minus a small constant), so context window limitations persist unchanged. A 10 kb sequence still requires approximately 10,000 tokens, and the quadratic attention complexity remains prohibitive for long sequences.

Second, overlapping tokenization creates ambiguity in how sequence positions map to tokens. A single nucleotide contributes to k different tokens, complicating interpretation of which token is responsible for any given prediction. This ambiguity becomes particularly problematic for variant effect interpretation, where one wants to understand how changing a specific nucleotide alters model predictions. The effect of a single nucleotide substitution propagates through k different tokens in ways that can be difficult to disentangle.

Third, the overlapping design introduces sample inefficiency. The model must learn that overlapping tokens share nucleotides, a relationship that is obvious from the tokenization scheme but must be discovered through training. This redundancy consumes model capacity that could otherwise be devoted to learning more complex biological patterns.

Fourth, the fixed 4^k vocabulary does not adapt to corpus statistics. Frequent and rare k-mers receive equal representation capacity in the embedding table, even though their importance for prediction may differ substantially. Common motifs that appear throughout the genome receive no more parameters than rare sequences that might represent sequencing errors or unique regulatory elements.

These limitations motivated exploration of alternative tokenization strategies that could achieve genuine sequence compression while preserving the information needed for biological prediction.

8.4. Byte Pair Encoding: Learning the Vocabulary

Byte Pair Encoding offers a fundamentally different approach to tokenization. Rather than defining tokens through a fixed rule (every k consecutive nucleotides), BPE constructs a vocabulary by learning which subsequences appear frequently in the training corpus. The algorithm, originally developed for data compression, iteratively merges the most frequent adjacent token pairs until reaching a desired vocabulary size.

The BPE algorithm begins by initializing the vocabulary with single nucleotides: {A, C, G, T}. It then scans the training corpus to count all adjacent token pairs and identifies the most frequent pair. This pair is merged into a new token, added to the vocabulary, and all instances in the corpus are replaced with the new token. The process repeats, counting pairs again (now including the newly created token) and merging the next most frequent pair. Through many iterations, BPE builds a vocabulary of variable-length tokens that capture frequently occurring sequence patterns.

The key insight is that BPE produces genuine sequence compression. Unlike overlapping k-mers where each nucleotide generates its own token, BPE creates non-overlapping tokens that can span multiple nucleotides. A 10 kb sequence might compress to 2,000 or 3,000 tokens depending on its repetitive structure, enabling transformers to process much longer sequences within the same context window.

DNABERT-2 replaced 6-mer tokenization with BPE and demonstrated dramatic improvements (Z. Zhou et al. 2024). The new model achieved comparable performance to state-of-the-art approaches while using 21 times fewer parameters and requiring approximately 92 times less GPU time in pretraining. The efficiency gains stem directly from non-overlapping tokenization: actual sequence compression enables processing longer sequences with the same computational budget, and eliminating the redundancy of overlapping tokens allows the model to focus capacity on learning biological patterns rather than token relationships.

The BPE vocabulary learns corpus statistics through its construction process. Repetitive elements that appear frequently throughout the genome, such as Alu sequences or common regulatory motifs, receive dedicated tokens that span many nucleotides. These long tokens enable efficient representation of repetitive regions while preserving single-nucleotide resolution for unique sequences. Rare sequences that BPE never encountered during vocabulary construction are represented as concatenations of shorter subunits, maintaining the ability to encode any sequence while allocating more representation capacity to common patterns.

GROVER (Genome Rules Obtained Via Extracted Representations) extended this approach by training BPE specifically on the human genome and selecting vocabulary using a custom next-k-mer prediction task (Sanabria et al. 2024). Analysis of the resulting token embeddings revealed that the learned vocabulary encodes biologically meaningful structure. Common tokens cluster separately from rare ones in embedding space. GC-rich tokens segregate from AT-rich tokens, reflecting the different properties of these sequence compositions. Token length correlates with specific embedding dimensions, allowing the model to represent both the content and extent of each token. Some tokens appear primarily in repetitive regions while others distribute broadly across the genome, and this localization pattern is captured in the learned representations.

Yet BPE introduces its own complications. The variable-length tokens mean that variant positions fall at different locations relative to token boundaries depending on the local sequence context. A SNP might fall in the middle of a long token in one context but at a token boundary in another,

potentially affecting how the model represents and processes the variant. This context-dependence can complicate variant effect interpretation, as the same nucleotide change may alter different numbers of tokens depending on surrounding sequence.

8.5. Single-Nucleotide Tokenization: The HyenaDNA Approach

While k-mer and BPE tokenization compress sequences to enable longer context windows, they sacrifice single-nucleotide resolution in doing so. This trade-off becomes problematic for variant effect prediction, where the precise position and identity of mutations is paramount. A single nucleotide polymorphism can completely alter protein function through mechanisms ranging from amino acid substitution to splice site disruption to regulatory element ablation. Multi-nucleotide tokens obscure exactly where variants fall and how they relate to the boundaries of biological features.

HyenaDNA (2023) took the opposite approach, using single-nucleotide tokens with no compression whatsoever (Nguyen et al. 2023). Each nucleotide (A, C, G, T) is a separate token, maintaining the maximum possible resolution. Every nucleotide is independently represented in the token sequence, SNP effects can be isolated to specific token positions without ambiguity, and there are no tokenization artifacts that depend on surrounding sequence context.

The challenge with single-nucleotide tokens is sequence length. A 1 Mb region requires 1 million tokens, far beyond the capacity of any standard transformer. The quadratic attention complexity would require a trillion pairwise computations per layer, rendering the approach computationally infeasible with conventional architectures.

HyenaDNA addressed this challenge through a fundamental architectural innovation rather than a tokenization compromise. The Hyena architecture replaces the attention mechanism with implicit convolutions that scale sub-quadratically with sequence length. Where attention computes explicit pairwise interactions between all positions, Hyena uses long convolutions parameterized by a small neural network, achieving similar representational power with $O(L \log L)$ complexity rather than $O(L^2)$. This enables processing of sequences hundreds of times longer than attention-based transformers within the same computational budget.

The result was a 500-fold increase in context length over dense attention models while maintaining single-nucleotide resolution. HyenaDNA could process 1 Mb sequences where DNABERT was limited to approximately 500 bp and the Nucleotide Transformer to approximately 6 kb. On the Nucleotide Transformer benchmarks, HyenaDNA reached state-of-the-art performance on 12 of 18 datasets with orders of magnitude fewer parameters and less pretraining data. On GenomicBenchmarks, it surpassed prior state-of-the-art on 7 of 8 datasets by an average of 10 accuracy points.

Perhaps most notably, HyenaDNA demonstrated the first use of in-context learning in genomics. The model could perform tasks based on examples provided in the context window without any fine-tuning, simply by conditioning on demonstration sequences. This capability, familiar from large language models, had not previously been shown for genomic sequences and suggests that very long context combined with high resolution enables qualitatively new forms of biological reasoning.

8.6. Biologically-Informed Tokenization

Standard tokenization schemes treat DNA as a homogeneous string of characters, ignoring the biological reality that different genomic regions serve fundamentally different functions and follow different structural rules. Coding sequences obey a strict codon structure where every three nucleotides encode an amino acid, while noncoding regions have no such constraint. Treating these regions identically wastes an opportunity to build biological knowledge directly into the representation.

Life-Code (2025) proposed codon-aware tokenization that respects the central dogma of molecular biology (Liu et al. 2025). The approach uses different tokenization strategies for different genomic regions based on their biological function. Coding regions are tokenized by codons, with each three-nucleotide unit encoding an amino acid becoming a single token. This aligns the token boundaries with the fundamental unit of protein translation, enabling the model to learn directly about amino acid sequences and protein structure. Noncoding regions, lacking codon structure, are tokenized by learned patterns that capture regulatory motifs and other functional elements.

This biologically-informed design enables Life-Code to learn protein structure through knowledge distillation from protein language models, capture interactions between coding and noncoding regions within a unified framework, and achieve state-of-the-art results across tasks involving DNA, RNA, and protein. The approach demonstrates that tokenization need not be uniform across the genome, and that encoding biological knowledge in the representation itself can improve model capabilities.

BioToken (2025) extends tokenization even further beyond sequence content to include explicit genomic structural annotations (Medvedev et al. 2025). Rather than treating variants as implicit changes in the sequence string, BioToken creates tokens that explicitly represent SNPs, insertions, and deletions. Known regulatory elements receive dedicated tokens encoding their presence and type. Gene structure, chromatin state, and other functional annotations are integrated directly into the token representation.

By incorporating biological inductive biases directly into tokenization, BioToken's associated model (BioFM) achieves competitive or superior performance to specialized models like Enformer and SpliceAI with significantly fewer parameters, approximately 265 million compared to the billions in some contemporary models. This efficiency suggests that appropriate representation can substitute for model scale, at least partially, by making the learning problem easier through informed structure.

8.7. The Context Length Evolution

Examining the history of genomic deep learning reveals a consistent trend toward longer sequence context, reflecting growing appreciation for the importance of distal regulatory interactions.

The earliest CNN models from 2015 to 2017, including DeepSEA and DeepBind, operated on sequences of approximately 1 kb, sufficient to capture local motifs and their immediate context. The next generation of models from 2018 to 2020, including ExPecto and SpliceAI, expanded to 10-40 kb windows, enabling capture of promoter-proximal regulatory elements and the extended context needed for accurate splice site prediction.

The transformer era beginning in 2021 brought divergent approaches. DNABERT with its overlapping k-mers was limited to approximately 512 bp of effective context, while Enformer combined CNN preprocessing with attention to achieve 200 kb contexts. The Nucleotide Transformer (2022-2023) pushed transformer-based models to 6 kb using k-mer tokenization. Then HyenaDNA and Caduceus (2023-2024) demonstrated that sub-quadratic architectures could reach 1 Mb while maintaining single-nucleotide resolution through character-level tokenization. Most recently, Evo 2 (2025) has achieved similar million-base-pair contexts using single-nucleotide tokens with BPE-style learned embeddings.

This progression reflects biological reality. Enhancers can regulate genes from hundreds of kilobases away. TAD boundaries and loop anchors create long-range dependencies in chromatin organization. Understanding genome function requires integrating information across these distances, and representation schemes must enable architectures capable of capturing such interactions.

8.8. Trade-offs and Practical Considerations

The choice between tokenization strategies involves multiple competing considerations that depend on the intended application.

Compression and resolution exist in fundamental tension. Higher compression enables longer context windows within fixed computational budgets, but loses precision for identifying exactly where variants fall and how they relate to biological features. One-hot encoding and single-nucleotide tokenization provide no compression but maintain full resolution. Non-overlapping k-mers achieve approximately k-fold compression at the cost of k-nucleotide resolution. BPE provides variable compression depending on sequence repetitiveness, with corresponding variable resolution. For variant effect prediction, where single nucleotide changes can have dramatic phenotypic consequences, resolution is paramount and the computational costs of long single-nucleotide sequences are often justified.

Vocabulary size affects both model capacity and efficiency. Larger vocabularies require bigger embedding tables but may capture more complex patterns directly. Smaller vocabularies are parameter-efficient but require the model to learn compositional structure through multiple layers. The vocabulary size of one-hot encoding (4 tokens plus special tokens) minimizes embedding parameters but maximizes the compositional learning burden. K-mer vocabularies scale exponentially with k, reaching 4,096 for 6-mers. BPE vocabularies are tunable, typically ranging from 4,096 to 32,000 tokens for genomic applications. Codon-aware approaches use approximately 64 codons plus additional tokens for noncoding regions.

Computational efficiency depends on both tokenization and architecture. For standard attention with $O(L^2)$ complexity, any compression directly reduces cost: non-overlapping k-mers reduce attention cost by a factor of k^2, and BPE with average compression c reduces cost by c^2. But sub-quadratic architectures like Hyena change this calculus, making single-nucleotide tokenization computationally feasible at long contexts and eliminating the need to trade resolution for efficiency.

For variant effect prediction specifically, tokenization choice has direct implications. Single-nucleotide tokens (as in HyenaDNA) enable clean comparison of reference and alternate alleles at the same token position with no ambiguity about effect localization. K-mer tokens complicate matters because a single SNP changes k overlapping tokens, requiring aggregation across affected tokens and introducing potential boundary effects. BPE tokens create context-dependent effects where the

same variant may fall at different positions relative to token boundaries depending on surrounding sequence, and where re-tokenization may be needed to properly represent the alternate allele.

8.9. The Emerging Consensus

Recent developments in the field suggest convergence toward several principles, though the optimal approach continues to evolve.

First, single-nucleotide resolution has become the preferred choice for applications requiring precise variant interpretation. The development of sub-quadratic architectures like Hyena, Mamba, and state space models has eliminated the computational barriers that previously forced researchers to accept resolution trade-offs. When long context and high resolution can both be achieved, there is little reason to sacrifice resolution through compression.

Second, learned embeddings rather than fixed representations have become standard. Even single-nucleotide tokenization now typically involves trainable embeddings that transform the four nucleotide identities into dense vectors. This allows the model to discover meaningful representations of nucleotide properties rather than treating all positions equivalently.

Third, biologically-informed augmentation has emerged as a promising direction for incorporating domain knowledge. Encoding codons in coding regions, incorporating functional annotations, or using species-specific vocabularies can provide useful inductive biases that improve learning efficiency and model interpretability.

Fourth, hybrid approaches that combine multiple representation strategies show promise for different genomic contexts. A model might use codon-level tokenization within genes while employing single-nucleotide tokens in regulatory regions, adapting the representation to the structure of each region.

The choice ultimately depends on the task at hand. Variant effect prediction demands high resolution and benefits most from single-nucleotide approaches. Species classification or repeat annotation may benefit from compression that enables comparison across longer regions. Expression prediction requires sufficient context to capture distal enhancers while maintaining resolution to identify causal variants. Understanding these trade-offs is essential for selecting or designing appropriate representations for specific applications.

8.10. Implications for Subsequent Chapters

The tokenization choices examined in this chapter set the stage for the genomic language models covered in Chapter 10. Understanding why models like the Nucleotide Transformer use 6-mers (Dalla-Torre et al. 2023), why DNABERT-2 switched to BPE, and why HyenaDNA's single-nucleotide approach enabled unprecedented context lengths clarifies the design space these models navigate. The hybrid architectures of Chapter 11, including Enformer and Borzoi, largely retained one-hot encoding for its precision in variant effect prediction, while the foundation models of Chapter 12 explore how sub-quadratic architectures enable single-nucleotide tokenization at truly genomic scale.

The representation problem remains an active area of research. As models grow larger and contexts extend further, new tokenization strategies may emerge that better balance compression, resolution, and biological structure. The field has moved from treating tokenization as a fixed preprocessing step to recognizing it as a fundamental design decision that shapes what models can learn and how they can be applied.

9. Protein Language Models

> ⚠ Warning
>
> **TODO:**
>
> - Add figure: ESM architecture diagram showing transformer layers, attention heads, and masked token prediction
> - Add figure: ESMFold pipeline diagram showing embedding extraction → structure module
> - Add figure: AlphaMissense workflow showing integration of PLM embeddings with structural context
> - Consider adding visualization of attention patterns capturing residue contacts
> - Add table comparing PLM architectures (ESM, ProtTrans variants, ESM-2 scaling)
> - Add discussion somewhere (here or VEP chapter) on marginal VEP calculations of log likelihoods

9.1. Evolutionary Sequences as Natural Language

Before transformers revolutionized genomic sequence modeling, they first transformed our ability to model proteins. The success of protein language models (PLMs) established a paradigm that would later inspire genomic foundation models: treat biological sequences as a form of natural language, train large transformer models on massive unlabeled sequence databases, and extract functional knowledge through self-supervised learning.

The analogy between protein sequences and natural language runs deeper than mere metaphor. Both encode complex information in linear strings of discrete tokens, whether amino acids or words. Both exhibit hierarchical structure, with motifs combining into domains as words combine into phrases. Both have syntax in the form of structural constraints and semantics in the form of functional meaning. And crucially, both are shaped by evolutionary pressure: natural selection filters protein sequences just as cultural selection shapes language.

This chapter examines how protein language models pioneered biological foundation modeling, from the ESM family's demonstration that transformers can learn protein structure and function from sequence alone, to their application in variant effect prediction and structure determination. Understanding PLMs provides essential context for the genomic language models covered in subsequent chapters, as many architectural choices and training strategies transfer directly from proteins to DNA.

9.2. The ESM Model Family

9.2.1. ESM-1b: Establishing the Paradigm

The Evolutionary Scale Modeling (ESM) project, developed at Meta AI Research, demonstrated that transformer language models trained on protein sequences learn biologically meaningful representations without explicit supervision (Rives et al. 2021). The key insight was that masked language modeling, the same objective that powers BERT in natural language processing, could be applied directly to amino acid sequences.

ESM-1b was trained on UniRef50, a clustered database of approximately 33 million protein sequences covering the known diversity of protein families. UniRef50 clusters sequences at 50% identity, providing broad coverage while reducing redundancy. This curation strategy ensures the model sees diverse evolutionary solutions to protein function rather than memorizing overrepresented families.

The architecture follows the BERT-style bidirectional transformer design with 650 million parameters distributed across 33 layers, a hidden dimension of 1,280, and 20 attention heads. The maximum sequence length of 1,024 amino acids accommodates most individual protein domains and many complete proteins. The training objective is masked language modeling: the model learns to predict randomly masked amino acids given surrounding context. This is analogous to BERT's masked token prediction, but operates on amino acids rather than words.

9.2.2. Emergent Biological Knowledge

Despite never seeing structural or functional labels during training, ESM learns representations that capture fundamental biological properties. This emergent knowledge manifests across multiple levels of protein organization.

At the level of secondary structure, attention patterns in ESM correlate with alpha helices and beta sheets. The model implicitly learns that certain amino acid patterns form specific structural elements, encoding this knowledge in its internal representations without any explicit supervision on structure labels.

ESM's attention heads also capture residue-residue contacts, identifying amino acids that are distant in sequence but close in three-dimensional space. This emergent capability suggests the model learns aspects of protein folding from sequence statistics alone. When researchers analyzed which sequence positions attend to each other in trained ESM models, they found strong correspondence with experimentally determined contact maps.

The model's masked token predictions correlate with position-specific conservation scores from multiple sequence alignments. ESM effectively learns which positions tolerate variation and which are evolutionarily constrained, extracting this information from the statistical patterns in sequence databases rather than from explicit conservation annotations.

Attention also concentrates on catalytic residues, binding sites, and other functionally important positions, even without explicit functional annotation in the training data. The model discovers that certain sequence positions are more informative about surrounding context, and these positions frequently correspond to sites of biological importance.

9.2.3. ESM-2: Scaling Up

ESM-2 extended the ESM approach with larger models and improved training (Lin et al. 2022). The model family spans several orders of magnitude in scale, from 8 million to 15 billion parameters, enabling systematic study of how biological knowledge scales with model capacity.

Model	Parameters	Layers	Contact Prediction Performance
ESM-2 (8M)	8M	6	Baseline
ESM-2 (35M)	35M	12	+5%
ESM-2 (150M)	150M	30	+8%
ESM-2 (650M)	650M	33	+12%
ESM-2 (3B)	3B	36	+15%
ESM-2 (15B)	15B	48	State-of-the-art

Performance scales smoothly with model size across structure prediction, contact prediction, and variant effect tasks. This phenomenon mirrors the scaling laws observed in natural language processing, where larger models consistently capture more nuanced patterns and achieve better downstream performance. The predictable scaling relationship suggests that continued investment in model size yields reliable returns in biological accuracy.

9.3. Alternative Architectures: The ProtTrans Family

The ProtTrans family explored multiple transformer architectures for protein sequences, demonstrating that the protein language modeling paradigm generalizes beyond the specific design choices of ESM.

ProtBERT applies the BERT-style bidirectional encoder to protein sequences, trained on the Big Fantastic Database (BFD) comprising approximately 2.1 billion protein sequences. This massive training corpus, substantially larger than UniRef50, provides even broader coverage of protein sequence space.

ProtT5 adapts the encoder-decoder architecture from T5, enabling both understanding and generation tasks. The encoder processes input sequences to produce contextual representations, while the decoder can generate output sequences conditioned on those representations. This architecture is particularly valuable for tasks that require sequence generation, such as protein design or sequence completion.

ProtXLNet explores permutation language modeling based on XLNet, capturing bidirectional context without the artificial [MASK] token that BERT-style models require during training. By training on all possible token orderings, XLNet-style models learn to predict each token from any subset of context tokens, potentially capturing richer dependencies.

These architectural variants demonstrate that the protein language modeling paradigm generalizes across architectures. The choice between encoder-only (BERT-style) and encoder-decoder (T5-style) models depends on the downstream application: encoders excel at classification and embedding tasks, while encoder-decoders enable sequence generation.

9.4. Zero-Shot Variant Effect Prediction

A critical application of protein language models is predicting the effects of amino acid substitutions. Missense variants are the most common type of protein-coding mutation, and clinical genetics pipelines must routinely assess whether specific substitutions are likely to be pathogenic or benign. Traditionally, this required either direct experimental characterization or computational methods trained on labeled pathogenicity data.

9.4.1. The Zero-Shot Paradigm

ESM-1v demonstrated that PLMs can predict variant effects without any training on variant labels (Meier et al. 2021). The approach exploits the masked language modeling objective: for a variant at position i changing amino acid a to amino acid b, compute the log-likelihood ratio:

$$\Delta\text{score} = \log P(b \mid \text{context}) - \log P(a \mid \text{context})$$

If the model assigns higher probability to the mutant amino acid than the wild-type, the variant is predicted benign; if lower, deleterious. This zero-shot prediction requires no labeled training data. The model's evolutionary knowledge, learned from sequence databases, directly informs variant interpretation.

The intuition is straightforward. If evolution has shaped protein sequences such that certain positions strongly prefer certain amino acids, substitutions that violate these preferences are likely to disrupt function. The language model captures these preferences through its training on millions of evolutionarily successful sequences. Variants that the model finds surprising, in the sense of assigning low probability, are more likely to be functionally disruptive.

9.4.2. Genome-Wide Application

Brandes and colleagues applied ESM-1b to predict effects for all approximately 450 million possible missense variants in the human genome (Brandes et al. 2023). This comprehensive annotation covers every position in every human protein multiplied by every possible amino acid substitution, providing precomputed effect scores that can be queried for any missense variant without running the model.

On ClinVar, the database of clinically annotated variants, ESM-1b outperformed existing methods in classifying approximately 150,000 missense variants as pathogenic or benign. The model achieved strong correlation with experimental measurements across 28 deep mutational scanning datasets, demonstrating that PLM predictions capture genuine functional information rather than merely correlating with annotation artifacts.

The analysis also identified approximately 2 million variants annotated as damaging only in specific protein isoforms, highlighting the importance of considering alternative splicing when interpreting variant effects. A variant that disrupts function in one isoform may have no effect if that isoform is not expressed in relevant tissues, underscoring the need to integrate PLM predictions with expression context.

9.4.3. The ProteinGym Benchmark

ProteinGym provides a comprehensive benchmark for variant effect predictors, aggregating 217 deep mutational scanning assays covering diverse proteins (Notin et al. 2023). Deep mutational scanning experiments systematically measure the functional effects of thousands of variants in a protein, providing ground truth for computational method evaluation.

Method	Mean Spearman
ESM-1v	0.48
EVE (evolutionary model)	0.46
DeepSequence	0.44
PolyPhen-2	0.32
SIFT	0.30

PLMs achieve competitive or superior performance to methods that explicitly model evolutionary conservation from multiple sequence alignments, despite using only single sequences as input. This suggests that transformer attention over large sequence databases captures similar information to traditional alignment-based approaches, but in a form that generalizes more readily to novel sequence contexts.

9.5. ESMFold: Structure from Sequence

9.5.1. Eliminating the Alignment Bottleneck

The most dramatic demonstration of PLM capabilities came with ESMFold, which predicts protein 3D structure directly from ESM-2 embeddings (Lin et al. 2022). Traditional structure prediction, including AlphaFold2, relies heavily on multiple sequence alignments (MSAs). These computationally expensive searches against sequence databases can take hours per protein, and the quality of predictions depends critically on finding informative homologs.

ESMFold eliminates this requirement entirely. The architecture couples ESM-2 (15 billion parameters) with a structure module adapted from AlphaFold2. The language model embeddings replace MSA-derived features, providing the evolutionary context that the structure module needs to predict atomic coordinates.

The computational speedup is substantial: approximately 60-fold faster than AlphaFold2 for typical proteins, enabling metagenomic-scale structure prediction. This speed advantage makes it feasible to predict structures for the millions of protein sequences emerging from environmental sequencing projects, where computing MSAs would be prohibitively expensive.

ESMFold achieves atomic-level accuracy for many proteins, though slightly below AlphaFold2 for proteins that benefit from MSA information. The accuracy gap is largest for proteins with sparse evolutionary sampling, where MSAs provide information that single-sequence analysis cannot recover. For well-represented protein families, ESMFold approaches AlphaFold2 accuracy at a fraction of the computational cost.

9.5.2. What ESMFold Reveals About PLMs

ESMFold's success demonstrates that ESM-2's internal representations encode sufficient information to determine 3D structure. The language model has learned not just local sequence patterns but global folding principles, capturing what makes a sequence fold into a particular shape.

This has profound implications for understanding what PLMs learn. The attention that transformers pay to distant sequence positions during masked prediction is, in some sense, learning the physics of protein folding. Residues that need to be close in 3D space attend to each other in the transformer's attention matrices. The statistical patterns in protein sequences, shaped by billions of years of evolution and the physical constraints of protein folding, encode structural information that sufficiently powerful language models can decode.

9.6. Integration into Variant Interpretation Pipelines

9.6.1. CADD v1.7: PLM Features for Ensemble Methods

The Combined Annotation Dependent Depletion (CADD) framework integrates diverse annotations to score variant deleteriousness (Chapter 4). CADD v1.7 incorporated ESM-1v predictions as features within its existing integrative architecture (Schubach et al. 2024).

The integration approach treats PLM scores as additional annotations alongside conservation scores, functional annotations, and regulatory predictions. For each missense variant, ESM-1v scores are computed and included as features in CADD's gradient-boosted tree classifier. This allows the ensemble to learn how PLM predictions complement other evidence sources, potentially capturing cases where PLM and conservation signals provide independent information.

Performance gains from PLM integration are consistent across benchmarks. On ClinVar pathogenic versus common variant classification, CADD v1.7 improves from 0.94 to 0.95 AUROC. On deep mutational scanning datasets (31 assays), performance improves from 0.78 to 0.81 Spearman correlation. The PLM features particularly improve scoring for variants in regions with limited evolutionary conservation data, where traditional methods struggle but language models can still extract contextual information.

9.6.2. AlphaMissense: Combining PLM and Structure

AlphaMissense represents the current state-of-the-art in missense variant effect prediction, combining PLM representations with structural context (Cheng et al. 2023). Rather than treating PLMs as a feature source for an external classifier, AlphaMissense adapts AlphaFold's architecture directly for pathogenicity prediction.

The model learns to predict pathogenicity by combining three information sources. Sequence embeddings from ESM-style language modeling provide evolutionary context about amino acid preferences at each position. Structural context from predicted protein structures captures whether a position is buried or exposed, in a secondary structure element or loop, near active sites or binding interfaces. Evolutionary information from cross-species comparisons supplements the single-sequence PLM signal with explicit alignment-derived conservation.

The training data comes from population frequency databases, primarily gnomAD. Common variants, those observed frequently in healthy populations, provide weak labels for benign effects. Variants absent from large population databases, particularly those in constrained positions, provide weak labels for deleterious effects. Critically, AlphaMissense never trains on clinical pathogenicity labels from ClinVar, yet achieves state-of-the-art performance on clinical benchmarks. This demonstrates that the combination of PLM representations, structural context, and population genetics signals captures genuine functional information rather than memorizing clinical annotations.

AlphaMissense provides predictions for all approximately 71 million possible single amino acid substitutions across the human proteome. Of these, 89% are classified as either likely benign or likely pathogenic with sufficient confidence to be actionable, providing interpretable predictions for the vast majority of possible missense variants.

Method	ClinVar AUC	DMS Correlation	Information Sources
SIFT	0.78	0.30	Conservation
PolyPhen-2	0.82	0.32	Conservation + structure
CADD v1.7	0.95	0.81	Multi-feature integration
ESM-1v	0.89	0.48	Sequence only (zero-shot)
AlphaMissense	0.94	0.52	PLM + structure + population

AlphaMissense achieves top performance by integrating the strengths of multiple approaches: PLM-derived sequence understanding, AlphaFold-derived structural context, and population genetics-derived evolutionary constraint signals.

9.7. Lessons for Genomic Foundation Models

The success of protein language models established several principles that inform genomic foundation modeling. These lessons transfer, with appropriate modifications, to the DNA language models covered in subsequent chapters.

9.7.1. Self-Supervision Works

PLMs demonstrated that massive amounts of biological knowledge can be learned from unlabeled sequences. The same evolutionary pressures that shape proteins also shape DNA. Purifying selection removes deleterious variants, leaving statistical signatures in sequence databases that self-supervised models can learn to exploit. This principle underlies the entire foundation model paradigm: if sufficiently large models are trained on sufficiently large datasets with appropriate self-supervised objectives, they will learn representations that capture biological function.

9.7.2. Scale Matters

Performance improves predictably with model size, motivating the development of larger genomic models. The progression from 8 million to 15 billion parameters in ESM-2 showed consistent gains across structure prediction, contact prediction, and variant effect tasks. While the relationship

between scale and performance is not linear indefinitely, current models remain in a regime where additional capacity yields reliable improvements. This scaling relationship justifies the substantial computational investment required to train genomic foundation models.

9.7.3. Transfer Learning is Effective

Representations learned for one task (masked token prediction) transfer to other tasks (structure prediction, variant effects). This suggests that self-supervised pretraining captures fundamental biological knowledge rather than task-specific shortcuts. A model trained to predict masked amino acids is simultaneously learning about protein structure, function, evolutionary constraint, and disease relevance, even though none of these properties appear in the training objective. The same principle applies to genomic sequences: models trained to predict masked nucleotides may simultaneously learn about regulatory elements, evolutionary conservation, and variant effects.

9.7.4. Architecture Choices Matter

The BERT-style bidirectional encoder proved highly effective for proteins, where the entire sequence context is typically available. However, genomic sequences present different challenges: much longer lengths spanning kilobases to megabases, different information density with proteins being information-dense while intergenic regions are less so, and different symmetries including the reverse-complement structure absent in proteins. These differences motivate architectural adaptations in genomic language models, including hybrid architectures that combine convolutional and attention mechanisms, longer context windows, and specialized tokenization schemes.

9.7.5. Integration with Other Modalities

AlphaMissense showed that PLM embeddings combine effectively with structural information. Similarly, genomic models benefit from integration with epigenomic data, gene annotations, and other biological context. The most powerful variant effect predictors combine multiple information sources, using PLMs as one component of larger systems. This principle extends to genomic foundation models, where sequence-based representations complement rather than replace other genomic annotations.

9.8. Limitations and Ongoing Challenges

Despite their success, protein language models face several limitations that inform the development of genomic models.

9.8.1. Sequence Length Constraints

Most PLMs handle sequences up to 1,000 to 2,000 amino acids. While sufficient for most individual protein domains, this limits modeling of large protein complexes and does not directly transfer to the much longer sequences in genomics. Genomic language models must handle sequences spanning millions of bases, requiring architectural innovations beyond simple scaling of transformer attention.

9.8.2. Orphan Proteins

PLMs struggle with proteins that have few homologs in training databases. Orphan or dark proteins, those unique to specific lineages, lack the evolutionary signal that PLMs exploit. For these proteins, the statistical patterns learned from diverse sequence families provide less informative context. This limitation is less severe for genomic models trained on reference genomes, where even unique sequences exist in the context of conserved flanking regions.

9.8.3. Epistasis

Most variant effect predictions assume independence: the effect of mutation A does not depend on whether mutation B is present. Real proteins exhibit epistasis, where variant effects depend on sequence context. Two individually benign variants may be jointly deleterious if they disrupt compensatory interactions. Current PLM-based predictors do not explicitly model these interaction effects, though the contextual embeddings may capture some epistatic relationships implicitly.

9.8.4. Interpretability

While attention patterns correlate with biological features, understanding exactly what PLMs learn remains challenging. The field is developing interpretation methods (Chapter 17), but PLMs remain partially opaque. For clinical applications where explanations are valued, this interpretability gap limits adoption. Future work must balance the accuracy gains from complex models against the transparency required for clinical decision-making.

9.9. Significance

Protein language models established that transformer architectures can learn deep biological knowledge from sequence data alone. ESM's ability to predict structure, function, and variant effects without explicit labels demonstrated the power of self-supervised learning on evolutionary data. This success directly motivated the development of genomic language models. If proteins constitute a language that transformers can learn, perhaps DNA does too.

The genomic language models covered in Chapter 10 adapt PLM architectures and training strategies to the distinct challenges of DNA sequences: longer contexts, different alphabets, and the full complexity of gene regulation. The integration path continues as well: just as CADD v1.7 and AlphaMissense incorporate PLM predictions, future models will integrate genomic and proteomic

language models into unified frameworks for variant interpretation (Chapter 13) and multi-omic modeling (Chapter 14).

10. Genomic Foundation Models

> ⚠ Warning
>
> **TODO:**
>
> - Add figure: timeline of genomic language model development (DNABERT → Nucleotide Transformer → HyenaDNA → Caduceus → GROVER)
> - Add figure: architecture comparison diagram showing transformer vs Hyena vs Mamba approaches
> - Add figure: context length evolution visualization showing the dramatic expansion from 512 bp to 1 Mb
> - Add visualization: benchmark performance comparison across Nucleotide Transformer tasks
> - Add figure: conceptual diagram of in-context learning in genomics (HyenaDNA)
> - Add table: comprehensive model comparison with parameters, training data, context length, and key innovations
> - Citations: verify all citations are in bibliography

Genomic language models extend the ideas of protein language models (Chapter 9) to the DNA level. They treat genomes themselves as a corpus, learn statistical regularities through self-supervision, and reuse those representations for many downstream tasks. Where Chapters 5–7 focused on supervised sequence-to-function CNNs and specialized architectures, and Chapter 8 focused on representation and tokenization, this chapter turns to genomic foundation models: large, often transformer-based or hybrid architectures trained on unlabeled genomic sequence at scale.

These models aim to provide a single, reusable backbone for tasks ranging from regulatory annotation and variant effect prediction to cross-species transfer and clinical prioritization. They mark the transition from building one model per dataset to constructing general-purpose genomic backbones analogous to BERT, GPT, and ESM in natural and protein language modeling.

10.1. From Supervised CNNs to Self-Supervised Genomic Language Models

The CNN era represented by DeepSEA, ExPecto, and SpliceAI (Chapters 5–7) shared a common pattern. Models took one-hot encoded DNA sequence around a locus as input, predicted task-specific labels such as chromatin marks, expression levels, or splice junctions, and optimized supervised loss functions against those labels. This approach achieved remarkable performance but suffered from three fundamental constraints.

The first constraint was label dependence. Every new assay, cell type, or phenotype required new labeled data to train a model. A chromatin accessibility model trained on ENCODE data could not predict histone modifications without additional labeled examples for those marks. This created substantial overhead for each new application.

The second constraint was task coupling. Model design became tightly coupled to the specific task. SpliceAI's architecture was specialized for splice junction prediction, with convolutions designed to capture the relevant spatial patterns. ExPecto's spatial feature transformation was engineered specifically for the distance-dependent relationship between regulatory elements and transcription start sites. These architectural choices, while effective for their intended purposes, did not transfer naturally to other problems.

The third constraint was limited reuse. Features learned for one problem did not automatically transfer to others. A model trained to predict chromatin accessibility might learn representations of regulatory motifs, but those representations were not directly accessible for other tasks like variant effect prediction or gene expression modeling without substantial re-engineering.

Protein language models showed a different route: self-supervised learning on unlabeled sequences, with downstream tasks solved by probing or fine-tuning. Genomic language models import this recipe to DNA. The training data comprises large collections of genomic sequences across species, individuals, or functional regions. The training objectives include masked language modeling, where the model predicts masked bases or tokens from surrounding context, and next-token or sequence modeling, where the model predicts the next token in a sequence. Some models combine these self-supervised objectives with auxiliary tasks such as predicting known annotations.

These pretrained models can be used in multiple ways. The simplest approach freezes the model and trains lightweight probes for specific tasks. Fine-tuning updates the entire model or uses adapter modules for specialized downstream applications. Zero-shot or few-shot scoring compares log-likelihoods of alternative sequences or alleles without any task-specific training. The promise is that once a sufficiently powerful backbone is trained, it becomes the default starting point for nearly any DNA-level prediction problem.

10.2. DNABERT: BERT for K-merized DNA

DNABERT applied the BERT masked language modeling framework to genomic sequences, using overlapping k-mers (typically 6-mers) as tokens and training on human reference sequences (Ji et al. 2021). As discussed in Chapter 8, this design had several defining characteristics.

The tokenization scheme converted DNA sequences into overlapping k-mers, creating a discrete vocabulary of size 4^k. For 6-mers, this yields a vocabulary of 4,096 tokens. The model used the standard BERT architecture with masked token prediction as its training objective. Context windows were relatively modest, spanning a few hundred base pairs (typically 512 tokens). The model was then fine-tuned on downstream tasks including promoter classification, splice site prediction, and transcription factor binding site identification.

DNABERT provided proof of concept for several important ideas. Self-supervised pretraining on raw DNA can improve performance over training from scratch. Learned embeddings capture biologically meaningful regularities, even when trained only on the reference genome. BERT-style architectures can be re-used across multiple downstream tasks with modest fine-tuning.

However, the k-mer design introduced significant limitations detailed in Chapter 8. The overlapping k-mer tokenization provided no true sequence compression, as each nucleotide participated in multiple adjacent tokens. This created ambiguity in positional interpretation, since the precise position of a variant within the k-mer vocabulary was unclear. The quadratic attention complexity of transformers combined with redundant overlapping tokens severely limited the effective context length.

10.3. DNABERT-2: Improved Tokenization and Efficiency

DNABERT-2 revisited both tokenization and architecture, demonstrating how much representation choices matter for genomic language models (Z. Zhou et al. 2024). The key differences relative to the original DNABERT addressed its core limitations.

The tokenization scheme adopted improved approaches such as BPE-style merges that better compress redundancies and reduce effective sequence length. This allowed the model to represent longer genomic contexts within the same number of tokens. Architectural refinements improved efficiency, enabling scaling to larger contexts and training corpora without prohibitive memory costs.

On standardized benchmarks spanning sequence classification, regulatory element prediction, and variant effect scoring, DNABERT-2 achieved consistent gains over both the original DNABERT and non-pretrained baselines. These improvements validated the importance of thoughtful tokenization design for genomic applications.

The DNABERT family collectively established three important principles. Self-supervision on DNA works and is competitive with hand-engineered pipelines for many sequence annotation tasks. Tokenization choices have large practical consequences, as the seemingly minor decision of how to convert nucleotides into tokens substantially affects both computational efficiency and downstream performance. Masked language model training can produce reusable representations for diverse sequence tasks, suggesting that the foundation model paradigm transfers effectively from natural language to genomic sequence.

10.4. Nucleotide Transformer: Scaling Context and Diversity

DNABERT demonstrated feasibility, but its context windows and training data were modest relative to the scale of genomes. The Nucleotide Transformer pushed much further, emphasizing scale and diversity in both model size and training corpus (Dalla-Torre et al. 2023).

The training corpus spanned genomic data from multiple species and populations, exposing the model to diverse sequence patterns. The architecture comprised transformer encoders of various sizes, from moderate to very large parameter counts. Context length expanded to approximately 6 kb per input sequence, representing an order-of-magnitude increase over DNABERT while still using dense attention. The training objective remained masked language modeling on subsequences sampled from genomes.

The Nucleotide Transformer work contributed several important ideas to the field. Cross-species pretraining, where training spans many genomes rather than a single reference, exposes the model

to diverse sequence patterns, different regulatory architectures, and evolutionary constraints that recur across lineages. This mirrors the use of large multi-species multiple sequence alignments in protein language models but operates at the raw DNA level.

To quantify representation quality, the Nucleotide Transformer introduced a benchmark panel of genomic tasks that has become a standard yardstick for subsequent DNA language models. Typical tasks include promoter and enhancer classification, histone mark and chromatin accessibility prediction, variant and pathogenicity proxies, and regulatory element type classification. Models are evaluated via linear probes, shallow classifiers, or light fine-tuning.

As with protein and natural-language models, performance improved predictably with larger models, more pretraining data, and longer context windows. These scaling trends help forecast the returns from investing in even larger genomic language models.

Model	Architecture	Max Context	Complexity
DNABERT	Transformer	512 bp	$O(L^2)$
Nucleotide Transformer	Transformer	6 kb	$O(L^2)$
HyenaDNA	Hyena	1 Mb	$O(L \log L)$
Caduceus	Mamba	1 Mb	$O(L)$

10.5. HyenaDNA: Megabase Context at Single-Nucleotide Resolution

Quadratic attention limits transformer context length to tens of kilobases at best, even with aggressive engineering. This is a fundamental architectural constraint: processing a 100 kb sequence with dense attention requires on the order of 10^{10} attention computations per layer. HyenaDNA addressed this limitation by replacing attention with a Hyena long-convolution architecture that scales sub-quadratically, enabling processing of sequences up to 1 Mb in length (Nguyen et al. 2023).

The Hyena architecture uses implicit convolutions, parameterizing long convolutional filters through neural networks rather than storing explicit filter weights. This approach achieves $O(L \log L)$ complexity through efficient FFT-based convolution, compared to the $O(L^2)$ complexity of standard attention. The result is a 500-fold increase in context length over previous dense attention models while maintaining single-nucleotide resolution.

HyenaDNA introduced several qualitative advances that matter for biological applications. Processing megabase-scale windows allows the model to see entire gene bodies plus their flanking regulatory regions, long-range enhancer-promoter interactions spanning tens to hundreds of kilobases, and topologically associating domain (TAD) scale structure. This aligns better with biological reality, where regulatory interactions often span substantial genomic distances.

Despite its long context, HyenaDNA maintains base-level resolution by using single-nucleotide tokens. This means single-nucleotide variants can be evaluated in the context of megabases of surrounding sequence without the ambiguity introduced by k-mer tokenization.

On Nucleotide Transformer benchmarks and additional tasks, HyenaDNA demonstrated in-context learning behaviors that had not previously been observed in genomic models. Performance improved when examples were included in the input context without updating model weights, suggesting that

at sufficient scale, DNA models can adapt to new tasks or distributions via prompts rather than fine-tuning. This mirrors phenomena observed in large natural language models.

On GenomicBenchmarks and related evaluations, HyenaDNA achieved state-of-the-art results on the majority of tasks, often by substantial margins. These results illustrated that architectural innovations enabling longer context can simultaneously provide both extended range and improved predictive accuracy.

10.6. Caduceus: Bidirectional Modeling with Reverse-Complement Equivariance

A unique challenge in genomic sequence modeling is the double-stranded nature of DNA. Any sequence can be read from either strand, and the reverse complement of a sequence encodes the same information from the opposite strand's perspective. For many biological processes, predictions should be identical or related in a consistent way regardless of which strand is presented to the model.

Standard neural networks can produce divergent predictions for a sequence and its reverse complement, even when training data is augmented with both orientations. This inconsistency is problematic for applications like regulatory element prediction, where the functional element exists on one physical stretch of DNA regardless of how we choose to represent it computationally.

Caduceus addressed this challenge by building reverse-complement equivariance directly into the architecture (Schiff et al. 2024). The model extends the Mamba architecture, a state-space model with linear complexity in sequence length, to support both bidirectionality and reverse-complement equivariance. The BiMamba component enables information flow in both directions along the sequence, while the MambaDNA block ensures that predictions for a sequence and its reverse complement are mathematically related in the expected way.

The architectural innovations in Caduceus serve distinct purposes. Bidirectionality allows each position to incorporate information from both upstream and downstream context, which matters for tasks where the relevant context is not directionally asymmetric. Reverse-complement equivariance ensures consistent predictions across strand orientations, reducing spurious variability and improving calibration.

On downstream benchmarks, Caduceus outperformed previous long-range models. On challenging long-range variant effect prediction tasks, Caduceus exceeded the performance of models with ten times as many parameters that did not leverage bidirectionality or equivariance. This suggests that incorporating appropriate biological inductive biases can be as valuable as scaling model size.

10.7. GROVER: Generative Regulatory Foundation Models

Most genomic language models focus on modeling raw DNA sequence. GROVER takes a complementary approach, shifting attention from sequence to regulatory tracks (Sanabria et al. 2024). Rather than treating DNA as the primary input, GROVER is trained on multi-track functional genomics signals including ATAC-seq, histone modifications, and other epigenomic assays across many cell types and tissues.

The training objective predicts masked or held-out regulatory profiles conditioned on neighboring tracks, cell-type embeddings, or limited sequence context. The architecture uses a transformer-style backbone tailored to spatiotemporal grids of genomic positions crossed with assays and cell types.

GROVER occupies a role analogous to self-supervised vision models for images. It treats regulatory profiles as a high-dimensional signal over the genome and learns rich representations of regulatory states at each position. This supports tasks like imputation of missing assays, denoising of noisy experimental data, and cell-type-specific activity prediction.

While not a pure DNA language model, GROVER-style systems complement sequence-based models in important ways. DNA language models capture what the genome can do, encoding the potential regulatory activities specified by the sequence. Regulatory foundation models like GROVER capture what the genome is actually doing in specific contexts, representing the realized regulatory state in particular cell types and conditions. Later chapters explore how sequence-based and regulatory foundation models can be combined, using DNA language models to parameterize sequence priors and regulatory models for context-specific readouts.

10.8. Central-Dogma-Aware and Annotation-Enriched Models

The tokenization discussion in Chapter 8 described how biological structure can be encoded directly into the input representation. Recent models push this idea further by integrating central dogma knowledge and genomic annotations into the modeling framework itself.

10.8.1. Life-Code: The Central Dogma as Inductive Bias

Life-Code proposes codon-aware, central-dogma-informed tokenization to bridge DNA, RNA, and protein within a single language-modeling framework (Liu et al. 2025). The key insight is that different genomic regions should be tokenized differently based on their biological function.

Coding regions are tokenized as codons, the three-nucleotide units that specify amino acids during translation. This respects the genetic code's fundamental structure and enables the model to learn patterns at the level of the biological unit of selection for protein-coding sequences. Noncoding regions, which lack this inherent three-nucleotide structure, are tokenized via learned subword units optimized during training. The resulting unified representations span DNA, RNA, and protein, enabling knowledge sharing across modalities.

Life-Code uses knowledge distillation from protein language models to import protein-level structural knowledge into DNA and RNA sequence representations. This improves performance on tasks involving coding sequence, such as predicting the effects of missense mutations or expression changes, and achieves competitive or state-of-the-art results on tasks across all three omic modalities.

10.8.2. BioToken: Encoding Variants and Structure

BioToken extends tokenization beyond nucleotide content to include explicit genomic annotations (Medvedev et al. 2025). Rather than representing a genomic region purely as a string of nucleotides, BioToken creates tokens that encode additional biological context.

Variant-aware tokens explicitly represent SNPs, insertions, and deletions as distinct tokens rather than as implicit changes in the underlying sequence. Structural annotations encode information about exons, introns, UTRs, promoters, enhancers, and other regulatory elements. Functional context tokens include signals such as conservation scores, chromatin state, or known regulatory motifs.

This design moves toward fully structured genomic language models where the input is not only DNA bases but also position-specific metadata. The resulting representations can directly integrate sequence, structure, and functional annotations in a unified framework.

The associated model BioFM, built on BioToken, achieves competitive or superior results relative to specialized models like Enformer and SpliceAI across genomic benchmarks including noncoding pathogenicity prediction, expression modulation, sQTL prediction, and long-range genomic interactions. Notably, BioFM achieves state-of-the-art performance with significantly fewer parameters (265M), substantially reducing training costs and computational requirements compared to larger models.

Life-Code and BioToken foreshadow the multi-modal, multi-omic foundation models discussed in Part IV, where sequence is only one of many integrated information streams.

10.9. Using Genomic Language Models in Practice

Genomic language models support multiple usage patterns analogous to those established for protein language models. Understanding these patterns is essential for applying the models effectively.

10.9.1. Embeddings as Universal Features

The simplest approach extracts embeddings from a pretrained model and uses them as features for downstream tasks. The workflow involves several steps: extract embeddings for windows around loci of interest, pool or select positions relevant to the task (such as promoters, candidate enhancers, or variant sites), and train a lightweight downstream model such as a linear layer, small MLP, or logistic regression.

This approach supports diverse applications. Regulatory element classification can distinguish promoters, enhancers, silencers, and insulators based on their learned representations. Chromatin state prediction uses sequence embeddings to predict ATAC-seq or histone mark presence as an alternative to supervised models like DeepSEA. Variant effect scoring replaces or augments hand-crafted features in frameworks like CADD with language model derived features, analogous to CADD v1.7's incorporation of protein language model features. Splicing and transcript modeling combines language model embeddings with specialized architectures like SpliceAI.

Because the language model remains frozen in this approach, it is computationally efficient and avoids catastrophic forgetting when new tasks are added. The pretrained model serves as a general-purpose feature extractor whose representations support many downstream applications.

10.9.2. Fine-Tuning and Task-Specific Heads

When more labeled data is available, fine-tuning can significantly improve performance beyond what frozen embeddings provide. Full fine-tuning updates all language model parameters for a specific task, allowing the model to specialize its representations. Adapter-based tuning inserts small bottleneck modules into each layer and updates only those, keeping the backbone mostly frozen while still allowing task-specific adaptation.

Full fine-tuning tends to achieve the highest performance when sufficient labeled data is available, but it requires more compute and risks catastrophic forgetting of general knowledge. Adapter-based approaches provide a middle ground, achieving most of the performance gains while maintaining computational efficiency and preserving the backbone's general capabilities.

10.9.3. Zero-Shot and Few-Shot Scoring

For variant interpretation, genomic language models enable zero-shot scoring based on sequence likelihood. The approach computes the model's probability for a sequence containing the reference allele and compares it to the probability for the sequence containing the alternative allele. Variants that substantially reduce sequence probability are inferred to be more disruptive.

This approach requires no variant-specific training data and can score any single-nucleotide variant in any genomic context the model has learned to represent. The quality of zero-shot scoring depends on how well the model's learned probability distribution captures biological constraints, which tends to improve with model scale and training data diversity.

Few-shot approaches include task examples in the input context, allowing the model to adapt its behavior based on demonstrations without parameter updates. HyenaDNA demonstrated that genomic models at sufficient scale exhibit this in-context learning capability, opening new possibilities for rapid task adaptation.

10.10. Emerging Themes and Current Limitations

The development of genomic language models over the past several years has established several important themes while also revealing significant limitations.

Self-supervision provides a viable path to general genomic representations. Models trained purely on the statistical structure of DNA sequence, without any functional labels, learn representations that transfer to diverse downstream tasks. This validates the foundation model paradigm for genomics and suggests continued scaling will yield further improvements.

Scale and diversity matter substantially for model quality. Performance improves predictably with model size, training data volume, and training data diversity. Including multiple species, populations, and genomic contexts yields more robust representations than training on a single reference genome.

Long-range context is biologically necessary for many applications. Regulatory phenomena operate at tens to hundreds of kilobases, and the development of efficient architectures like HyenaDNA and Caduceus finally allows modeling these interactions at single-base resolution. The progression from 512 bp to 1 Mb context lengths represents a fundamental capability improvement.

Self-supervision and supervision are complementary rather than competing approaches. Self-supervised language models excel at learning broad, reusable features, but they do not automatically solve every downstream problem. Specialized architectures and supervised objectives, such as Enformer and related models discussed in Chapter 11, remain crucial for accurate quantitative prediction of complex genomic readouts.

Several important limitations remain. Current models struggle with complex variant patterns beyond single-nucleotide changes, including indels, structural variants, and epistatic interactions across distant loci. Training data and labels remain skewed toward certain ancestries, raising concerns about performance and calibration in underrepresented populations. Interpretability is limited, as it remains difficult to explain why a model assigns a particular score to a variant in terms that connect to biological mechanism. Integration with other data modalities (chromatin, expression, 3D genome structure, clinical phenotypes) is still in its early stages.

10.11. Summary

This chapter surveyed the landscape of genomic language models, from early proof-of-concept systems like DNABERT through scaled models like Nucleotide Transformer to architectural innovations enabling megabase context in HyenaDNA and Caduceus. We examined how models like GROVER complement sequence-based approaches by learning from regulatory tracks, and how annotation-enriched architectures like Life-Code and BioToken incorporate biological structure directly into the modeling framework.

The key lessons are that self-supervised pretraining transfers effectively to genomics, that architectural choices enabling long-range context provide both efficiency and accuracy improvements, and that biological inductive biases (reverse-complement equivariance, central dogma awareness, variant encoding) can substitute for raw scale in some applications.

In Chapter 11, we turn to Enformer and related long-range sequence-to-function models that explicitly predict molecular readouts from sequence. These models close the loop between self-supervised sequence understanding and supervised functional prediction, addressing a key limitation of pure language models: their indirect relationship to quantitative molecular phenotypes.

11. Long-range Hybrid Models

> ⚠ Warning
>
> **TODO:**
>
> - Add figure: Enformer architecture diagram showing CNN stem → transformer trunk → multi-task heads
> - Add figure: Comparison of effective receptive fields across models (DeepSEA 1kb → Basenji2 40kb → Enformer 200kb)
> - Add figure: Borzoi RNA-seq coverage prediction example showing transcription, splicing, and polyadenylation signals
> - Add visualization: Attention weight patterns showing promoter-enhancer interactions
> - Add table: Comprehensive comparison of Basenji2, Enformer, and Borzoi (context length, parameters, training data, output tracks)
> - Consider adding Evo2 and HyenaDNA as emerging alternatives to attention-based long-range modeling
> - Reference UK Biobank fine-tuned variants (UKEnformer, UKBorzoi) if published by time of print
> - Ensure cross-reference to AlphaMissense and AlphaGenome discussion in Chapter 13

11.1. Why Expression Needs Long-Range Models

ExPecto (Chapter 6) showed that gene expression can be predicted *ab initio* from sequence by combining a CNN-based chromatin model (Beluga) with a separate regression layer mapping chromatin features to expression across tissues (J. Zhou et al. 2018). This modular strategy worked surprisingly well, but it inherited two key limitations from its DeepSEA-style backbone (Chapter 5). First, the 40 kb input window captures proximal promoters and some nearby enhancers, but many regulatory interactions span 100 kb or more. Second, chromatin prediction and expression prediction are trained separately, leaving no opportunity for the expression objective to shape the representation of sequence.

As genomic datasets grew through ENCODE, Roadmap, FANTOM, GTEx, and other consortia (Chapter 2), it became clear that enhancers can regulate genes hundreds of kilobases away, that eQTLs often sit outside the promoter windows traditionally used for expression models, and that chromatin conformation introduces non-local dependencies between DNA segments through loops and topologically associating domains. Pure CNN architectures can expand their receptive field using dilated convolutions and pooling, but doing so at single-nucleotide resolution quickly becomes parameter- and memory-intensive. On the other hand, classic transformer architectures can model

long-range dependencies via attention, but their quadratic runtime and memory in sequence length makes naïve application to 200 kb sequences computationally infeasible (Chapter 10).

Hybrid architectures like Enformer and Borzoi emerged as a compromise between these constraints. These models use convolutions to extract local motif features and progressively downsample the sequence into a manageable number of latent positions, then apply self-attention over this compressed representation to capture long-range regulatory interactions across 100 to 200 kilobases. By predicting many signals at once, including chromatin profiles, transcription start site activity, and RNA-seq coverage, they enable multi-task learning and rich variant effect prediction. This chapter focuses on these hybrid designs, particularly Enformer (Ž. Avsec et al. 2021) and Borzoi (Linder et al. 2025), and how they changed what sequence-to-expression models can accomplish.

11.2. Problem Setting: Sequence-to-Expression at Scale

The models in this chapter tackle a demanding version of the classic sequence-to-function problem: given a long DNA sequence window around a genomic locus, predict a rich set of regulatory and transcriptional readouts across many cell types.

11.2.1. Inputs and Outputs

The input to these models is a one-hot encoded DNA sequence, typically around 200 kb centered on a candidate promoter or gene. Each position in the sequence is represented by one of four channels corresponding to the nucleotides A, C, G, and T, with N positions masked or handled by zeroing all channels. Beyond the raw sequence, the model must know where promoter-proximal bases and distal elements sit relative to each other. This positional information is encoded through convolutional receptive fields in the early layers and through explicit positional embeddings for the attention layers.

The outputs of Enformer and Borzoi are multi-task, multi-position predictions spanning multiple assays, multiple cell types, and multiple positions along the input window. The assays include chromatin accessibility measured by DNase-seq or ATAC-seq, histone modifications such as H3K4me3 and H3K27ac, and transcriptional activity measured by CAGE or RNA-seq. Each assay is predicted separately for hundreds of cell types and experimental conditions, and predictions are made at fixed strides across the input window, typically every 128 or 256 base pairs, yielding coverage tracks rather than single scalar values. The result is a dense tensor of predictions with dimensions corresponding to output positions, assay types, and cell types.

11.2.2. Training Objective

The typical training objective involves per-track, per-position regression using either a Poisson or negative binomial likelihood on read counts, or alternatively a mean-squared error loss on log-transformed counts. All tracks contribute to the loss simultaneously, though some models tune weights to prevent abundant assays like DNase from dominating scarce but potentially important ones like rare histone marks. The learning problem can be written as a function f_θ that maps a DNA sequence of approximately 200 kb to a tensor of continuous outputs indexed by tracks and positions, with parameters θ shared across assays, cell types, and genomic loci.

11.3. Enformer: CNN Plus Attention for 200 kb Context

Enformer (Ž. Avsec et al. 2021) is a landmark model that directly integrates long-range sequence context with cell-type-specific expression prediction using a hybrid CNN-transformer architecture. The name, a portmanteau of "enhancer" and "transformer," reflects its primary innovation: using attention mechanisms to capture the relationships between enhancers and promoters that may be separated by tens or hundreds of kilobases.

11.3.1. Architectural Overview

Conceptually, Enformer consists of three stages. The first stage is a convolutional stem that extracts local motifs and progressively downsamples the sequence. The second stage is a transformer trunk that applies self-attention to model long-range dependencies between the downsampled positions. The third stage comprises output heads that decode the attended representation into assay- and cell-type-specific coverage tracks.

The convolutional front-end takes approximately 200 kb of one-hot encoded sequence as input and applies stacked convolution-normalization-nonlinearity-pooling layers. This progressively compresses the input, with each pooling operation reducing the spatial dimension while the convolutional operations expand the channel dimension. By the end of this stage, the roughly 200,000 base pair input has been reduced to around 1,500 to 2,000 latent tokens, each summarizing a multi-kilobase region and encoding local motif configurations and short-range regulatory patterns. This compression step solves the computational problem of applying attention to raw nucleotides: rather than computing attention over 200,000 positions with quadratic cost, the model operates on a tractable sequence of around 1,500 positions.

The transformer trunk then applies several transformer blocks over this compressed sequence. Multi-head self-attention allows every downsampled position to attend to every other position, capturing relationships between distant enhancers and promoters or between multiple regulatory elements that may all contribute to a gene's expression. Feed-forward networks provide nonlinear mixing of information at each position, and residual connections with layer normalization stabilize training and enable deep stacks. Intuitively, the convolutional layers answer the question of what motifs and local patterns exist in each region, while the attention layers answer the question of how these regions interact across the 200 kb window to shape regulatory activity.

After the transformer blocks, Enformer applies task-specific output heads to each position in the latent sequence, producing coverage predictions for each combination of assay and cell type. For CAGE-based transcription start site activity, the model predicts coverage around TSS positions, and gene-level expression metrics can be obtained by aggregating predictions at positions near annotated transcription start sites through summing or averaging log counts across a small window.

Enformer differs from its predecessor Basenji2 in several key respects. It uses transformer blocks instead of dilated convolutions for long-range modeling, attention pooling instead of max pooling for downsampling, twice as many channels in the network, and 1.5 times longer input sequences (197 kb instead of 131 kb). These changes collectively enable the model to capture regulatory elements up to 100 kb from a gene, compared to only about 20 kb for Basenji2. This expanded receptive field is biologically important: estimates from high-confidence enhancer-gene pairs suggest that 47% of relevant enhancers lie within 20 kb of their target genes, but 84% lie within 100 kb.

11.3.2. Training Data and Cross-Species Learning

Enformer is trained on a large collection of human and mouse regulatory datasets. The human data includes DNase-seq, histone ChIP-seq, and CAGE across many cell types from ENCODE, Roadmap Epigenomics, and other consortia. Mouse data from analogous assays enables cross-species learning. Two key design choices shape the training regime.

First, joint human-mouse training encourages the model to learn regulatory principles conserved across mammals rather than overfitting to species-specific patterns. This approach also enables zero-shot transfer between species for some tasks, as representations learned from one species can generalize to the other. Second, entire chromosomes are held out for evaluation to avoid overly optimistic performance estimates that might arise from local sequence similarity between training and test examples. The loss aggregates over all targets, all positions in the output window, and all training loci.

11.3.3. Variant Effect Prediction

Like DeepSEA before it, Enformer can be used for in silico variant effect prediction. The procedure involves extracting a 200 kb window around a locus from the reference genome, running Enformer to obtain predicted coverage tracks, introducing an alternative allele into the window, re-predicting coverage, and computing the difference between alternate and reference predictions. This delta can be computed for each track at each position, and aggregating these differences around transcription start sites quantifies the predicted expression change for genes in each cell type.

This approach allows fine-grained assessment of how a variant might alter promoter-proximal signals and distal enhancer contributions. Because the model sees a 200 kb context, it can in principle detect cases where a variant disrupts a distal enhancer that regulates a gene tens of kilobases away. The variant-level scores can be integrated into downstream tools such as fine-mapping pipelines that require per-variant effect estimates.

11.3.4. Validation Against GTEx eQTLs

Enformer's variant effect predictions were systematically evaluated using GTEx eQTL data (Chapter 2). For each gene-tissue pair, known eQTLs (lead variants from association testing) were compared to non-eQTL variants in linkage disequilibrium. The evaluation used signed LD profile (SLDP) regression, which correlates predicted expression effects with observed eQTL effect sizes while accounting for LD structure. Enformer's predictions showed stronger alignment with observed eQTLs than prior models like Basenji2, with improvement especially notable at distal regulatory variants where long-range attention is crucial (Ž. Avsec et al. 2021).

In practice, this means Enformer can prioritize variants likely to be causal eQTLs rather than merely correlated through linkage disequilibrium. The model provides cell-type-specific effect predictions, which are critical for interpreting tissues with sparse experimental data. If a variant is predicted to have a large effect in a particular tissue, that prediction can inform downstream analyses of tissue-specific disease mechanisms.

11.3.5. Interpretation and Mechanistic Insight

While Enformer is a complex model, several interpretation strategies provide mechanistic insight. Gradient-based attribution computes gradients of gene-level expression predictions with respect to input sequence, highlighting bases or motifs that drive the predicted expression of a gene in a specific cell type. In silico mutagenesis systematically mutates bases to estimate their impact on target genes or tracks, identifying enhancers and key transcription factor binding sites controlling expression. Analysis of attention weights reveals which positions attend most strongly to a promoter, suggesting candidate long-range enhancers.

These tools have been used to map promoter-enhancer interactions directly from sequence and to suggest causal regulatory elements for disease-associated variants. Contribution scores computed for genes with CRISPRi-validated enhancers correlate with H3K27ac marks and highlight not only local promoter regions but also distal enhancers more than 20 kb away. By contrast, contribution scores from Basenji2 are zero for sequences beyond 20 kb due to its limited receptive field. This provides evidence that Enformer genuinely uses biologically relevant distal sequence when making predictions.

11.4. Borzoi: Transcriptome-Centric Hybrid Modeling

Enformer is primarily trained on chromatin and CAGE profiles, which capture regulatory states and transcription initiation but not the full complexity of RNA processing. Borzoi (Linder et al. 2025) extends the hybrid architecture paradigm to model the RNA transcriptome itself, with emphasis on finer-grained transcriptional features including splicing and polyadenylation.

11.4.1. Motivation

RNA-seq data carries richer information than a single expression scalar per gene. Coverage along exons and introns reflects transcription initiation, elongation, and termination. Splice junction usage reveals alternative splicing patterns, complementing specialized models like SpliceAI (Chapter 7). Coverage patterns around 3' UTRs and polyadenylation sites reflect mRNA stability, localization, and translation efficiency.

A general-purpose model that predicts base-level RNA-seq read coverage from DNA sequence could provide a unified framework for transcript-level variant effect prediction spanning transcription, splicing, and polyadenylation. It could also offer mechanistic insight into how regulatory sequence features shape the full life cycle of transcripts, from initiation through processing to eventual degradation.

11.4.2. Architecture

Borzoi builds on the Enformer-style backbone with modifications tailored to RNA-seq prediction. The convolutional front-end processes long DNA windows on the order of 100 to 200 kb, learning local motifs and regulatory patterns at single-nucleotide or modestly downsampled resolution. A hybrid long-range module uses attention and/or long-range convolutions to integrate information across the entire context, explicitly designed to capture relationships between promoters, internal

exons, and distal elements. Multi-layer output heads predict RNA-seq coverage tracks across the window, with separate tracks for sense versus antisense transcription, splice junction signals, and polyadenylation-related coverage around 3' ends.

Like Enformer, Borzoi is trained in a multi-task regime, but with stronger emphasis on RNA-related readouts. Where DeepSEA, Beluga, and Enformer mapped sequence to chromatin plus transcription start activity, Borzoi maps sequence to full transcriptome coverage.

11.4.3. From Chromatin Signals to RNA Readouts

This shift to RNA-level prediction supports several analyses not possible with chromatin-focused models. Promoter usage can be assessed by distinguishing alternative promoter transcription start sites based on coverage patterns. Alternative splicing can be predicted through differential exon inclusion or skipping, complementing the splice-site-focused approach of SpliceAI. Coverage drop-offs and polyadenylation-linked patterns enable modeling of 3' UTR and polyA site choice.

Variant effect prediction follows similar steps as with Enformer: predict transcriptome outputs for reference and alternate sequences, compute delta-coverage at exons, splice junctions, and 3' ends, and aggregate into variant-level scores for tasks like eQTL or sQTL prioritization. The richer output enables combined assessment of how a single variant might affect transcription, splicing, and polyadenylation simultaneously.

11.5. What Hybrid Models Changed

Hybrid CNN-transformer sequence models like Enformer and Borzoi introduced several conceptual advances over earlier architectures.

11.5.1. Explicit Long-Range Modeling

By combining convolutional downsampling with attention over latent tokens, these models achieve hundreds of kilobases of effective context with manageable compute. All positions in the compressed representation can interact, approximating many possible promoter-enhancer relationships. This is crucial for capturing distal enhancers that sit far from genes and for modeling complex regulatory architectures where multiple enhancers and silencers integrate to control expression.

Earlier CNN-only models like DeepSEA and Basenji2 could expand their receptive field through dilated convolutions, but the information flow between distant positions remained indirect, passing through many intermediate layers. Attention allows direct information exchange between any two positions in the compressed sequence, which the original Enformer paper showed outperforms dilated convolutions across model sizes and training data volumes.

11.5.2. Unified Multi-Task Learning Across Modalities

Hybrid models jointly predict chromatin accessibility, histone marks, and transcriptional activity in a single forward pass. This multi-task learning yields shared representations that capture general regulatory logic, regularization across assays and cell types that reduces overfitting to any single dataset, and a pathway to transfer learning where a single pretrained model can be adapted to downstream tasks.

The multi-task setup also enables consistency checking: a variant predicted to strongly increase H3K27ac (an enhancer mark) but not affect CAGE output would be suspicious, as these signals typically correlate at active regulatory elements. The model implicitly learns these relationships through joint training.

11.5.3. Improved Variant Effect Prediction for Expression

Compared to earlier CNN-only models like DeepSEA, Beluga, ExPecto, and Basenji2, Enformer demonstrated stronger eQTL concordance and better performance on expression-related benchmarks (Ž. Avsec et al. 2021). Hybrid designs can identify distal causal variants more reliably because their architecture naturally encodes long-range dependencies. Borzoi extends this further by providing detailed transcriptome-level readouts, enabling combined assessment of transcription, splicing, and polyadenylation for each variant and offering a richer mechanistic understanding of how sequence variation impacts the full RNA life cycle.

11.6. Limitations and Failure Modes

Despite their power, hybrid long-range models are not omniscient and introduce new challenges alongside their capabilities.

11.6.1. Data and Label Limitations

The training data for these models, drawn primarily from ENCODE, Roadmap Epigenomics, GTEx, and similar resources, have known biases. The assays focus on specific cell types, conditions, and genomic regions. GTEx eQTLs are enriched for individuals of European ancestry (Chapter 2). Many regulatory phenomena, such as RNA binding protein effects and 3D chromatin structure beyond simple contact frequency, are only partially captured by the available assays.

As a result, the models may underperform in cell types or ancestries not well represented in the training data. They may also misinterpret patterns that are confounded by technical artifacts such as batch effects or mapping biases. A predicted effect in a rare cell type or an underrepresented population should be treated with appropriate caution.

11.6.2. Sequence Context and Generalization

Enformer and Borzoi are trained on fixed window sizes around annotated loci, and their behavior outside those canonical windows may be less reliable. Training focuses on reference genome context, meaning large indels, structural variants, or rearrangements may be poorly modeled. The models assume linear genomic context: 3D chromatin architecture is only indirectly captured via sequence patterns correlated with looping, and explicit Hi-C or Micro-C integration remains limited.

These constraints mean that a variant prediction assumes the rest of the genome matches the reference, which is never true for any real individual. Epistatic effects between multiple variants in the same regulatory region, or between a variant and a structural rearrangement, are not captured.

11.6.3. Interpretability and Trust

Although attribution methods exist and have yielded biologically plausible results, attention weights and gradient-based scores are not direct causal evidence. Attributions can be noisy and sensitive to how targets are aggregated. For clinical use, predictions often require orthogonal validation through CRISPR perturbation, allele-specific expression assays, or other experimental approaches. These interpretability challenges are part of the broader issues discussed in the chapters on evaluation (Chapter 15) and confounders (Chapter 16).

11.7. Role in the Genomic Foundation Model Landscape

Hybrid architectures like Enformer and Borzoi occupy an interesting middle ground between task-specific CNNs and general-purpose genomic foundation models. Compared to earlier CNN systems, they model much longer context and support richer multi-modal outputs, offering significantly improved expression-related variant effect prediction. Compared to the self-supervised genomic language models discussed in Chapter 10, they are specialized and supervised on particular assays rather than trained with broad self-supervision on raw genomes. Their architecture is hand-crafted for specific tasks (chromatin plus expression) rather than serving as a universal pretraining backbone.

In practice, hybrid models serve multiple roles. They function as high-performance baselines for variant effect prediction tasks, especially when expression or RNA readouts are primary endpoints. Their representations can be adapted for downstream tasks or combined with pretrained language models over DNA. Their "convolutional stem plus long-range module plus multi-task heads" pattern has become a design template that newer architectures borrow, substituting attention for alternative long-range mechanisms such as state space models, Hyena, or Mamba (Chapter 12).

As the field moves toward large, multi-modal genomic foundation models that integrate sequence, chromatin, expression, and 3D structure, Enformer and Borzoi represent key waypoints. They demonstrate that long-range context is essential for accurate expression prediction, that hybrid architectures can make such context computationally tractable, and that multi-task supervision across regulatory layers is an effective path from raw DNA to clinically relevant variant effect predictions.

11.8. Summary

This chapter examined hybrid CNN-transformer architectures designed for long-range genomic prediction, focusing on Enformer and Borzoi as representative examples.

Enformer combines a convolutional stem with transformer blocks to achieve 200 kb context windows, enabling the model to capture distal enhancer-promoter interactions that purely convolutional models miss. Joint training on human and mouse data across thousands of chromatin and CAGE tracks produces representations that improve eQTL prioritization and provide cell-type-specific expression effect predictions. Borzoi extends this approach to predict RNA-seq coverage directly, enabling unified assessment of transcription, splicing, and polyadenylation effects.

The key lessons from this chapter are that long-range context substantially improves expression prediction, that hybrid architectures offer a practical solution to the computational constraints of attention over long sequences, and that multi-task learning across regulatory modalities yields representations useful for variant interpretation. At the same time, these models inherit biases from their training data, assume reference genome context, and require experimental validation for clinical applications.

In Chapter 12, we step back to consider what makes a model a "genomic foundation model" more broadly, examining the design dimensions, evaluation frameworks, and emerging architectures that define this rapidly evolving space.

Part IV.

Part IV: GFMs & Multi-omics

12. Genomic FMs: Principles & Practice

> ⚠ Warning
>
> **TODO:**
>
> - Add figure: taxonomy of genomic foundation models (DNA LM, seq→function, variant-centric, multi-omic) showing the four quadrants with representative models in each
> - Add figure: design dimensions diagram showing data, architecture, objectives, and tokenization as orthogonal axes
> - Add table: comparison of GFM families (context length, parameter count, pretraining objective, key applications)
> - Add figure: evaluation pyramid from molecular readouts to clinical decisions
> - Add decision tree flowchart for practitioners choosing appropriate GFM for their task
> - Add figure: adapter strategies (linear probe, LoRA, full fine-tune) with computational cost comparison

Genomic foundation models represent the culmination of several threads developed across the earlier parts of this book: high-fidelity variant calling, regulatory sequence-to-function prediction, protein language models, and long-context transformers for DNA. These models extend the ideas presented in previous chapters into systems that are general-purpose, pretrained at scale, and reusable across a wide range of genomic and genetic tasks.

This chapter steps back from individual architectures to address a more fundamental question: what does it mean for a model to be a genomic foundation model? We organize the emerging ecosystem into a practical taxonomy, distill design principles that guide model selection and development, and provide guidance for practitioners seeking to integrate these models into their workflows. The conceptual framework established here will guide the remaining chapters of Part IV as we examine specific application domains.

12.1. From Task-Specific Models to Genomic Foundation Models

The earlier chapters traced a fairly linear progression through the history of computational approaches to genomic prediction. Hand-crafted scores and shallow models such as CADD and early pathogenicity predictors established the value of integrating diverse annotations for variant interpretation (Rentzsch et al. 2019; Schubach et al. 2024). Task-specific deep models such as DeepSEA, ExPecto, Sei, Enformer, and SpliceAI demonstrated that neural networks could learn regulatory and splicing effects directly from sequence, often surpassing the performance of feature-engineered approaches (J. Zhou and Troyanskaya 2015; J. Zhou et al. 2018; Chen et al. 2022; Ž. Avsec et al. 2021; Jaganathan et al. 2019). Sequence language models over proteins and DNA, including ESM, DNABERT, Nucleotide Transformer, HyenaDNA, and GROVER, showed that general sequence

representations could be learned via self-supervision and then transferred to diverse downstream tasks (Rives et al. 2021; Lin et al. 2022; Brandes et al. 2023; Ji et al. 2021; Dalla-Torre et al. 2023; Nguyen et al. 2023; Sanabria et al. 2024).

Foundation models build on these ingredients but fundamentally change the contract between model and user. The primary product of a genomic foundation model is not a task-specific prediction head but rather a reusable representation, and sometimes a general interface, that can be adapted to many downstream tasks with modest additional supervision. HyenaDNA exemplifies this paradigm: a genomic foundation model pretrained on the human reference genome with context lengths up to one million tokens at single-nucleotide resolution using a Hyena-based long-range architecture. DNABERT-2, Nucleotide Transformer V2, Caduceus-Ph, GROVER, and related models form a parallel family of transformer-style DNA foundation models. A recent benchmark comparing these five models across diverse tasks including classification, gene expression prediction, variant effect quantification, and TAD recognition illustrates both the promise and the limitations of current DNA foundation models (Manzo, Borkowski, and Ovcharenko 2025).

At a high level, genomic foundation models extend the pretrain-then-finetune paradigm from natural language processing and protein modeling into genomics, but with domain-specific constraints that distinguish them from their counterparts in other fields. These constraints include extreme context lengths necessary to capture distal regulatory interactions, single-nucleotide sensitivity required for variant effect prediction, and strong mechanistic priors that arise from decades of molecular biology research.

12.2. What Makes a Model a Genomic Foundation Model?

The term "foundation model" is sometimes used loosely in the genomics literature, applied to any large neural network trained on genomic data. For practical purposes, it is useful to establish working criteria that separate true genomic foundation models from ordinary deep models that happen to operate on biological sequences.

12.2.1. Working Definition

A genomic foundation model is a pretrained model that satisfies several key properties. First, it learns from large-scale genomic data with minimal task-specific supervision, typically through pretraining on entire genomes or large portions thereof across species or populations. The objectives employed during pretraining include masked language modeling, next-token prediction, denoising, or multi-task sequence-to-function prediction.

Second, a genomic foundation model produces general-purpose representations. These take the form of embeddings of sequences, variants, loci, or genes that prove useful across many downstream tasks. Critically, these representations can be extracted and reused with light adapters or linear probes rather than requiring full model retraining.

Third, genomic foundation models are designed for broad transfer. They support many downstream tasks without retraining the full model, enabling transfer across assays (from chromatin marks to gene expression), across tissues, across species, and across variant types.

Fourth, these models scale along at least one dimension. Some scale context length, as in HyenaDNA's million-token window. Others scale parameter count, as in the ESM and Nucleotide Transformer families. Still others scale data diversity through pan-genomic pretraining or cross-species corpora.

Fifth, genomic foundation models typically expose a relatively standardized interface. This includes a common API for embeddings, sequence scoring, and mask-based perturbation, and models are often distributed via model hubs such as Hugging Face with documented recipes for downstream applications.

Many excellent deep models for genomics fail one or more of these criteria. Early versions of DeepSEA or SpliceAI, for instance, were trained for specific assays or tasks, used narrowly scoped inputs and outputs, and were not designed for broad reuse beyond their original application domains.

12.2.2. Foundation Models Versus Large Models

Scale alone does not make a model a foundation model. A very large Enformer-like model trained solely on human chromatin tracks is powerful but remains strongly bound to a specific prediction interface that maps sequence to a fixed set of chromatin tracks. By contrast, a DNA language model like HyenaDNA or DNABERT-2 is explicitly trained to model raw sequence using a general objective and is naturally repurposed as an embedding engine for diverse downstream applications.

This distinction matters because it affects how models should be evaluated. Foundation models must be assessed across families of tasks rather than single benchmarks, using resources like TraitGym for trait-level performance and ProteinGym for variant effect prediction (Benegas, Eraslan, and Song 2025; Notin et al. 2023). The distinction also affects how models should be integrated into existing pipelines, since foundation models serve as feature extractors while task-specific models typically serve as end-to-end predictors.

12.3. A Taxonomy of Genomic Foundation Models

The landscape of genomic foundation models can be organized into four broad families, each with distinct characteristics, strengths, and typical applications. Understanding this taxonomy helps practitioners select appropriate models for their specific needs and helps researchers position new contributions within the broader field.

12.3.1. DNA Language Models

The first family comprises DNA language models that learn sequence representations from raw nucleotide strings. Representative examples include DNABERT and DNABERT-2, which apply BERT-style masked language modeling to DNA sequences (Ji et al. 2021; Z. Zhou et al. 2024). The Nucleotide Transformer family scales this approach to larger models and cross-species training corpora (Dalla-Torre et al. 2023). HyenaDNA uses implicit convolutions rather than attention to achieve subquadratic complexity, enabling context lengths up to one million nucleotides (Nguyen et al. 2023). Caduceus incorporates bidirectional processing and reverse-complement equivariance as architectural inductive biases. GROVER combines BPE-style tokenization with training on regulatory tracks rather than raw sequence alone (Sanabria et al. 2024).

These models share several characteristics. They are typically trained on reference genomes with self-supervised objectives, they produce embeddings that can be probed or fine-tuned for diverse tasks, and they vary primarily in context length, architectural family (transformer versus state space model versus hybrid), and tokenization strategy.

12.3.2. Sequence-to-Function Genomic Foundation Models

The second family comprises sequence-to-function models that predict molecular readouts directly from sequence. These models blur into foundation model territory when their output space is sufficiently broad and their internal representations are reused for tasks beyond the original assay set. Examples include Enformer, which predicts thousands of chromatin and expression tracks from 200 kb sequence windows (Ž. Avsec et al. 2021), and Sei, which organizes predictions into interpretable sequence classes that capture regulatory grammar (Chen et al. 2022). These models typically operate over longer context windows of 100 kb or more and provide variant effect scores by computing delta-predictions between reference and alternative alleles.

Enformer serves as a prototypical example of a sequence-to-function model that has been widely reused as a feature extractor for downstream tasks including gene expression prediction and fine-mapping of regulatory variants. While these models were originally trained for specific assays, they approximate foundation models when the output space spans many cell types and assays and when their internal representations prove useful for tasks beyond the original prediction targets.

12.3.3. Variant-Centric Genomic Foundation Models

A third class of foundation models focuses not on raw sequence but on genetic variants as the fundamental unit. These models embed variants using contextual information from local sequence, gene structure, and external annotations, and they predict variant pathogenicity, molecular consequences, or trait-level effect sizes.

Examples in this space include CADD and its deep-learning-enhanced successor models, which integrate annotations and sequence features for broad variant pathogenicity scoring (Rentzsch et al. 2019; Schubach et al. 2024). AlphaMissense repurposes ESM-style protein language models to predict missense pathogenicity at scale (Cheng et al. 2023). Delphi, MIFM, and related models couple genomic foundation model embeddings with polygenic score estimation for complex traits (Georgantas, Kutalik, and Richiardi 2024; Rakowski and Lippert 2025; Wu et al. 2024). Emerging variant representation learning datasets and benchmarks such as GV-Rep explicitly probe how well foundation models represent genetic variants and clinical annotations.

Variant-centric foundation models blur the line between feature extractors and trait models. Their predictions can be plugged directly into polygenic score pipelines, risk stratification tools, or rare disease interpretation workflows, making them particularly relevant for clinical applications.

12.3.4. Multi-omic and Cross-Modal Foundation Models

Finally, a growing set of models aim to natively integrate multiple modalities. These include models that jointly process DNA sequence, chromatin state, and gene expression; models that incorporate

sequence and 3D genome structure from Hi-C or Micro-C experiments; and models that combine DNA with non-sequence modalities such as images or free text descriptions of function.

Recent work on architectures like Omni-DNA explores transformer-based auto-regressive models that jointly handle DNA and task-specific tokens, enabling multi-task learning over sequence, epigenetic marks, and even textual descriptions of function. These models move genomic foundation models closer to a unified interface for genome biology, though at the cost of more complex training objectives and data engineering requirements.

12.4. Design Dimensions of Genomic Foundation Models

When designing or selecting a genomic foundation model, it is helpful to think in terms of several orthogonal design dimensions. Each dimension involves trade-offs that affect model performance, computational requirements, and suitability for specific applications.

12.4.1. Data: What Does the Model See?

The choice of training data fundamentally shapes what a model can learn. Key decisions include species coverage, assay diversity, population diversity, and sampling strategies.

Regarding species coverage, models may be trained on human genomes only, focusing on clinical and human genetics applications, or they may incorporate cross-species pretraining on dozens or hundreds of species. Cross-species training, as employed by Nucleotide Transformer and many protein language models, encourages discovery of conserved regulatory code and can improve out-of-domain generalization (Dalla-Torre et al. 2023; Rives et al. 2021).

Assay diversity matters for sequence-to-function models. The choice of which epigenomic assays, cell types, and perturbation datasets to include during training determines what molecular readouts the model can predict and, more subtly, what regulatory patterns it learns to recognize. Collections like Cistrome provide rich training data spanning transcription factor binding, histone modifications, and chromatin accessibility across many cell types (R. Zheng et al. 2019).

Population diversity is crucial for avoiding biased models. Inclusion of genomes from diverse ancestries is necessary to prevent embedding population-specific biases into foundation models and downstream risk scores. Early deep learning approaches to polygenic score estimation, including Delphi and MIFM, explicitly tackle ancestry-aware evaluation to quantify and mitigate these biases (Georgantas, Kutalik, and Richiardi 2024; Rakowski and Lippert 2025; Wu et al. 2024).

Context length and sampling strategies also play important roles. Some models randomly slice long chromosomes into training windows, as in HyenaDNA. Others use targeted sampling around genes, enhancers, or known variants. Warm-up schedules that gradually increase context length can stabilize training for long-context models.

12.4.2. Architecture: How Does the Model Process Sequence?

Architectural choices determine the computational properties of a model, including maximum practical context length, memory and compute requirements, and ease of adaptation for different tasks.

Transformer architectures dominate current genomic foundation models and come in several flavors. Encoder-only models following the BERT design, such as DNABERT and Nucleotide Transformer, are well-suited for classification and embedding tasks. Decoder-only models following the GPT design, such as GROVER and some Omni-DNA variants, align naturally with generative tasks. Encoder-decoder hybrids support tasks requiring explicit outputs such as sequence-to-text explanations.

Attention-free long-range models address the quadratic complexity of standard attention. Hyena-based models like HyenaDNA use implicit convolutions to achieve subquadratic complexity, enabling million-token contexts. State space models and related architectures trade exact attention for scalable long-range interactions while maintaining competitive performance on many tasks.

Dense-attention long-range transformers demonstrate that with careful engineering and context extension schedules, dense-attention transformers can also reach approximately 200 kb contexts at single-nucleotide resolution. Models like Gene42 show that the attention-versus-efficiency trade-off is not absolute.

Hybrid architectures combine multiple approaches. CNN-plus-transformer stacks use local convolutions followed by global attention, as seen in Enformer-like models (Ž. Avsec et al. 2021). Cross-attention mechanisms can integrate DNA with auxiliary modalities such as chromatin state or 3D contact maps.

12.4.3. Objectives: What Does the Model Learn to Predict?

The training objective shapes the representations a model learns and its suitability for different downstream applications.

Masked token prediction randomly masks nucleotides or k-mers and trains the model to predict them given surrounding context. This approach, used by DNABERT, DNABERT-2, and many transformer-based models, encourages learning of local and medium-range dependencies (Ji et al. 2021; Z. Zhou et al. 2024).

Next-token prediction uses an autoregressive language model objective, as in GROVER and HyenaDNA. This approach naturally aligns with generative tasks and in-context learning, and it leverages techniques developed for large language models in natural language processing.

Denoising and span corruption objectives replace or permute spans of sequence and train the model to reconstruct them. These approaches encourage robustness to small perturbations and attention to long-range structure.

Multi-task sequence-to-function prediction directly predicts chromatin profiles, transcription factor binding, accessibility, expression, and other molecular readouts from sequence. Models like DeepSEA, Enformer, and Sei use this approach, which functions as a powerful regularizer and provides a direct bridge between sequence patterns and molecular phenotypes (J. Zhou and Troyanskaya 2015; Ž. Avsec et al. 2021; Chen et al. 2022).

Cross-modal objectives jointly predict sequence features and other modalities. Examples include contrastive alignment between DNA slices and 3D contacts or histone marks, and joint prediction of sequence, epigenetic tracks, and textual function labels in Omni-DNA-like architectures.

12.4.4. Tokenization and Representations

Tokenization presents a non-trivial design choice for DNA models, with different approaches offering distinct trade-offs.

Character-level tokenization treats each nucleotide as a separate token. This is the simplest approach and maintains single-nucleotide resolution, making it compatible with precise variant effect prediction. HyenaDNA and many sequence-to-function models use this approach (Nguyen et al. 2023).

K-mer tokenization groups nucleotides into overlapping or non-overlapping k-mers, creating vocabularies of size 4^k. For 6-mers, this yields 4,096 tokens. K-mer tokenization reduces sequence length and helps transformers reach longer effective contexts, but at the cost of positional ambiguity and reduced resolution (Ji et al. 2021).

Learned tokenization approaches, such as BPE-style methods used in BioToken, discover subsequence units optimized for downstream performance rather than using fixed vocabularies (Medvedev et al. 2025). These approaches can allocate vocabulary capacity efficiently, representing common patterns compactly while maintaining the ability to encode rare sequences.

Internally, genomic foundation models typically produce per-position embeddings $h_i \in \mathbb{R}^d$ for each nucleotide or token, pooled sequence embeddings that summarize an entire region through mean pooling, CLS tokens, or learned pooling operations, and variant embeddings constructed by contrasting reference versus alternative alleles, sometimes augmented with structural context. The choice of pooling strategy can significantly influence downstream performance, and benchmarking studies have found that simple mean pooling of per-token embeddings often outperforms more elaborate strategies across many tasks (Manzo, Borkowski, and Ovcharenko 2025).

12.5. Evaluating Genomic Foundation Models

Because genomic foundation models are intended to serve as foundations for many applications, their evaluation must be broader than single-task metrics. A model that excels at one benchmark may fail on others, and performance on standard benchmarks may not predict utility for real-world applications.

12.5.1. Downstream Task Suites and Benchmarks

Emerging benchmark suites provide structured evaluations across diverse tasks. ProteinGym evaluates variant effect prediction across many proteins for protein language models (Notin et al. 2023). TraitGym assesses trait-level performance of regulatory and genomic models across complex trait prediction tasks (Benegas, Eraslan, and Song 2025). Comparative evaluations of DNA language models and regulatory models, such as the work by Manzo and colleagues, compare models across regulatory genomics tasks (Manzo, Borkowski, and Ovcharenko 2025). DNA foundation model benchmarks systematically compare models like DNABERT-2, Nucleotide Transformer V2,

HyenaDNA, Caduceus-Ph, and GROVER across classification, variant effect, and TAD recognition tasks. Variant-centric benchmarks like GV-Rep probe models' ability to represent clinical variants and their genomic contexts.

A key lesson from these benchmarks is that no single model dominates all tasks. General-purpose DNA foundation models often perform well overall but may lag specialized architectures for gene expression and eQTL prediction, while excelling for variant prioritization and regulatory element annotation.

12.5.2. Evaluation Modes

Genomic foundation models can be evaluated in several regimes that test different aspects of their utility.

Zero-shot evaluation uses frozen embeddings with simple operations such as similarity computations or clustering, or with predefined scoring rules. This tests whether useful information is accessible without any task-specific training. An example would be using HyenaDNA embeddings directly for in-context learning on simple motif tasks.

Linear probes train shallow linear or logistic regression heads on top of frozen embeddings. This provides a quick measure of how easily information is linearly decodable from the model's representations and is often used as a diagnostic for representation quality.

Lightweight adaptation includes approaches like low-rank adaptation (LoRA), prompt tuning, or small MLP heads fine-tuned on specific tasks. These methods balance performance with computational cost and stability, enabling adaptation without the full expense of end-to-end fine-tuning.

Full fine-tuning updates all model parameters on a downstream task. This typically yields the best task-specific performance but requires more data and computation, and risks overfitting to the specific task distribution.

The choice among these evaluation modes depends on the amount of labeled data available, computational constraints, and whether the goal is to assess representation quality or to achieve maximum task performance.

12.6. Practical Integration of Genomic Foundation Models

For practitioners seeking to use genomic foundation models in their work, several questions guide the choice of model and integration strategy.

12.6.1. Selecting a Model for Your Task

The appropriate model depends on the specific application. For missense variant interpretation, protein language models like ESM-2 or AlphaMissense provide strong baselines with well-characterized performance (Cheng et al. 2023). For non-coding variant interpretation, sequence-to-function models like Enformer or DNA language models fine-tuned on regulatory tasks are more appropriate (Ž. Avsec et al. 2021). For tasks requiring very long genomic context, such as enhancer-promoter linking

or structural variant interpretation, models like HyenaDNA or long-context dense-attention models like Gene42 should be considered. For regulatory variant interpretation near genes, Enformer-like or DeepSEA-like models can be compared against DNA language models working via embeddings (J. Zhou and Troyanskaya 2015; Chen et al. 2022; Ji et al. 2021). For trait-level prediction with large cohorts, polygenic score pipelines incorporating GFM-based variant priors, such as Delphi or MIFM, offer promising approaches (Georgantas, Kutalik, and Richiardi 2024; Rakowski and Lippert 2025; Wu et al. 2024). For method development and benchmarking, standardized benchmark suites like TraitGym, ProteinGym, GV-Rep, and DNA foundation model comparison studies ensure that comparisons are meaningful (Benegas, Eraslan, and Song 2025; Notin et al. 2023; Manzo, Borkowski, and Ovcharenko 2025).

12.6.2. Integration Strategies

Once a model is selected, several integration strategies are available. The simplest approach uses the model as a feature extractor, computing embeddings or predictions for variants or sequences of interest and then feeding these features into downstream models or pipelines. This approach is computationally efficient and compatible with existing infrastructure.

Adapter-based fine-tuning keeps the foundation model frozen while training small adapter modules on task-specific data. This preserves the general knowledge in the foundation model while adapting its representations to the specific task.

End-to-end fine-tuning updates the entire model on task-specific data. This can achieve the best performance but requires more data and computation and may sacrifice generality.

Ensemble approaches combine predictions from multiple models, often achieving better performance and calibration than any single model. This is particularly valuable when different models have complementary strengths.

12.7. Safety, Robustness, and Responsible Use

As genomic foundation models become infrastructure for clinical and research pipelines, considerations of safety and robustness move from optional extras to essential requirements.

12.7.1. Robustness and Adversarial Sensitivity

Recent work on genomic foundation model robustness highlights that these models can be surprisingly sensitive to adversarial perturbations at both the input sequence level and through soft prompts in embedding space. Even when perturbations are hardly biologically plausible, they reveal fragility of decision boundaries in high-dimensional representation space and potential failure modes where small spurious changes strongly impact pathogenicity or variant effect predictions.

These findings suggest that adversarial testing should become part of genomic foundation model validation, especially for clinical use cases. Robust training approaches, including data augmentation, adversarial objectives, or distributionally robust optimization, may be needed for high-stakes applications.

12.7.2. Bias, Fairness, and Ancestry

Genomic foundation models trained predominantly on reference genomes or Euro-centric cohorts risk encoding biased priors. These biases can manifest as underestimation of risk in underrepresented ancestries and misclassification of benign variants that are common in certain populations but rare in training data.

Deep polygenic score and variant interpretation pipelines that incorporate genomic foundation models should perform ancestry-stratified evaluation and consider explicit debiasing through reweighting and careful calibration (Georgantas, Kutalik, and Richiardi 2024; Rakowski and Lippert 2025; Wu et al. 2024).

12.7.3. Data Governance and Privacy

Because genomic foundation models are often trained on large collections of genomic sequences, data use agreements and privacy protections must be respected. Some cohort-level datasets cannot be used for unrestricted pretraining due to consent restrictions. Even when training on reference genomes, leakage from labeled clinical datasets into training may complicate downstream evaluation.

To date, most published genomic foundation models emphasize training on public reference genomes or synthetic benchmarks, but clinical deployment will require stronger guarantees about data provenance and privacy protection.

12.8. Open Challenges and Future Directions

Genomic foundation models are still in their early days, and several open challenges stand out as important directions for future work.

12.8.1. Toward Unified Multi-omic Foundation Models

Current genomic foundation models remain fragmented across DNA-only language models, sequence-to-function models tied to specific assays, variant-centric pathogenicity models, and protein and RNA language models. A major frontier is the development of unified multi-omic foundation models that jointly model DNA, RNA, protein, chromatin, and 3D genome structure. Such models would support cross-modal queries, enabling questions like “given this variant, what is the likely impact on transcription factor binding, chromatin accessibility, and gene expression in a specific cell type?” They would also provide interpretable pathways connecting sequence variation to phenotypes. Models like Omni-DNA represent first steps in this direction, demonstrating that multi-task, cross-modal training is feasible at scale.

12.8.2. Integrating Causal and Mechanistic Structure

Most genomic foundation models are trained with purely predictive objectives. Incorporating more causal structure could improve robustness to distribution shift between cell types or interventions and enable counterfactual reasoning about hypothetical perturbations like enhancer knockouts.

Potential routes toward more causal models include causal representation learning on top of foundation model embeddings, mechanistic constraints derived from gene regulatory networks or biochemical kinetics, and joint modeling of perturbation data from CRISPR screens or gene knockouts with observational genomics.

12.8.3. Efficient and Accessible Deployment

Even if genomic foundation models train on large clusters, their deployment should be feasible in typical research labs and clinical environments. Approaches to improve accessibility include distillation into smaller student models, efficient inference via sparsity, quantization, and hardware-aware architectures, and task-specific adapters that keep the frozen backbone small enough for on-premise use.

The long-range efficiency of architectures like HyenaDNA and the emergence of dense-attention models like Gene42 suggest multiple viable paths to deployable genomic foundation models.

12.9. Summary

This chapter has provided a framework for understanding genomic foundation models as a distinct class of computational tools for genome biology. We defined what it means for a model to be a genomic foundation model, emphasizing the properties of scale, generality, and reusability that distinguish foundation models from task-specific deep models. We proposed a practical taxonomy organizing the field into DNA language models, sequence-to-function genomic foundation models, variant-centric genomic foundation models, and emerging multi-omic models.

We surveyed the core design dimensions along which models differ, including data composition, architecture, training objectives, and tokenization strategies. We discussed evaluation regimes and benchmark suites that assess genomic foundation models across diverse tasks and outlined how practitioners can integrate these models into variant interpretation, regulatory genomics, and trait prediction pipelines. Finally, we highlighted emerging concerns around robustness, bias, and responsible deployment that must be addressed as these models move toward clinical applications.

The remaining chapters of Part IV will dive deeper into specific application domains. Chapter 13 recasts variant effect prediction in the foundation model era, examining how protein and DNA-based approaches can be combined and calibrated. Chapter 14 broadens the view from isolated sequences to multi-omic and systems-level representations, exploring models that integrate genomic, transcriptomic, proteomic, and phenotype data. Throughout, the conceptual framework established here will help organize a rapidly evolving ecosystem of genomic foundation models.

13. Variant Effect Prediction

> ⚠ Warning
>
> **TODO:**
>
> - Add figure: AlphaMissense architecture diagram showing integration of MSA-based language modeling with AlphaFold2 structural context
> - Add figure: GPN-MSA input representation showing multi-species alignment stack and masked language modeling objective
> - Add figure: Evo 2 architecture overview showing StripedHyena 2 with context length scaling
> - Add figure: AlphaGenome unified architecture diagram showing convolutional encoder, transformer blocks, and multi-task prediction heads
> - Add visualization: comparative variant effect prediction across models (coding vs non-coding performance)
> - Consider adding schematic of practical VEP workflow integrating multiple tools
> - Add benchmark performance table across ClinVar, MAVE, and regulatory variant datasets

13.1. From Handcrafted Scores to Foundation Models

Variant effect prediction sits at the heart of modern genomics. Most variants discovered in clinical sequencing are rare and lack direct experimental evidence, yet clinicians still need to decide whether they are benign, pathogenic, or somewhere in between. Earlier in this book we encountered several approaches to this problem. Conservation and heuristic scores such as SIFT, PolyPhen, and CADD combine evolutionary constraint with manually engineered features to estimate deleteriousness (**?@sec-deleteriousness**) (Ng and Henikoff 2003; Adzhubei et al. 2010; Rentzsch et al. 2019). Sequence-to-function CNNs like DeepSEA and ExPecto predict chromatin and expression effects that can serve as proxies for regulatory impact (Chapter 5; Chapter 6). Specialized architectures like SpliceAI target specific molecular mechanisms such as splicing disruption (Chapter 7). Protein language models trained on massive sequence databases learn representations that correlate with fitness and can be adapted for missense variant effect prediction (Chapter 9).

The frontier today is shaped by foundation models that combine massive pretraining at proteome or genome scale, long-range context spanning kilobases to megabases, and multiple sources of information including sequence, structure, multi-species alignments, and multi-omic outputs. These systems represent a qualitative shift from the feature engineering paradigm of earlier methods. Rather than defining relevant features a priori and training classifiers on those features, foundation models learn rich representations from data and then apply those representations to variant interpretation tasks.

This chapter surveys four landmark systems that define the current state of the art: AlphaMissense for proteome-wide missense pathogenicity prediction, GPN-MSA for genome-wide variant effect prediction from multi-species alignments, Evo 2 as a generalist genomic language model spanning all domains of life, and AlphaGenome as a unified megabase-scale sequence-to-function model with state-of-the-art regulatory variant effect prediction. Together, they preview what genomic foundation models look like when specialized for variant interpretation.

13.2. AlphaMissense: Proteome-Wide Missense Pathogenicity

AlphaMissense, developed by DeepMind, provides precomputed pathogenicity scores for approximately 71 million possible human missense variants, covering almost every single amino acid change in the proteome (Cheng et al. 2023). This comprehensive scoring emerged from combining two powerful sources of information: the evolutionary constraints captured by protein language models and the structural context provided by AlphaFold2.

13.2.1. Combining Sequence and Structure

The model builds on two complementary pillars. The first is protein language modeling, where a transformer-based model is trained on massive multiple sequence alignments to learn which amino acids tend to appear at each position across evolution. From this training, the model infers how surprising a given amino acid substitution is in its evolutionary context. Positions that are highly conserved across species receive confident predictions that alternative amino acids are deleterious, while positions that vary freely across evolution are predicted to tolerate substitutions.

The second pillar is predicted three-dimensional structure from AlphaFold2. Structural context helps distinguish tolerated changes from disruptive ones in ways that sequence alone cannot capture. A substitution on a solvent-exposed loop may be well tolerated even if the position is somewhat conserved, while a substitution in a tightly packed hydrophobic core may be disruptive even at a position with some evolutionary variability. The structural environment, including local secondary structure, packing density, and proximity to functional sites, provides crucial information about the consequences of amino acid changes.

For each variant, AlphaMissense ingests the wild-type sequence, the substitution position and amino acid change, sequence context from the MSA, and structural environment derived from AlphaFold2. These features are fed into a neural network that outputs a pathogenicity probability between 0 and 1.

13.2.2. Training and Calibration

AlphaMissense employs a hybrid training strategy. Self-supervised pretraining learns general sequence and structural representations from evolutionary data, building on the foundational representations developed for AlphaFold2 and protein language modeling. Supervised calibration then uses ClinVar and similar databases for labeled pathogenic and benign variants, along with population frequency information from gnomAD under the assumption that common variants are more likely benign.

The model's raw scores are calibrated so that scores near 0 correspond to likely benign variants, scores near 1 correspond to likely pathogenic variants, and intermediate scores capture uncertainty and ambiguous cases. In practice, AlphaMissense adopts score cutoffs that approximately map to the "likely benign," "uncertain," and "likely pathogenic" categories used in clinical interpretation frameworks such as ACMG guidelines.

13.2.3. Performance and Clinical Utility

Across diverse benchmarks including ClinVar, curated expert panels, and multiplexed assays of variant effect (MAVEs), AlphaMissense achieves state-of-the-art AUROC and AUPRC for missense variant effect prediction. The model generalizes across many genes, including those with little prior annotation, and produces scores that correlate more consistently with experimental functional readouts than many earlier predictors.

These properties have led to rapid adoption. AlphaMissense scores have been integrated into clinical re-annotation of exomes, reclassification of variants of uncertain significance (VUS), and gene-specific studies where high-throughput functional assays are impractical. The precomputed nature of the predictions, covering essentially all possible missense variants, eliminates the need for users to run inference themselves.

13.2.4. Limitations and Caveats

Despite its impressive performance, AlphaMissense has important limitations that users must understand. The model handles missense variants only and does not natively score nonsense, frameshift, regulatory, or deep intronic variants. It operates on single variants at a time, ignoring combinations of variants such as compound heterozygosity or epistatic interactions. The training depends on labels from ClinVar and population databases, meaning that any biases in those resources, including ancestry representation biases, can propagate into scores. Finally, while attention maps and feature attributions can be examined, the reasoning underlying a particular score is often opaque.

For these reasons, clinical guidelines recommend treating AlphaMissense as supporting evidence to be combined with segregation data, functional assays, and population frequencies rather than as a standalone decision-maker.

13.3. GPN-MSA: Genome-Wide Variant Effect Prediction from Alignments

While AlphaMissense focuses on proteins, GPN-MSA tackles the harder problem of genome-wide variant effect prediction directly at the DNA level (Benegas, Albors, et al. 2024). This expansion is critical because most disease-associated variants discovered through GWAS and clinical sequencing fall in noncoding regions where protein-based methods provide no information.

13.3.1. An Alignment-Based DNA Language Model

GPN-MSA extends earlier Genomic Pre-trained Network (GPN) models by operating on multi-species genome alignments. The input is a stack of aligned sequences from multiple species, for example human plus dozens of mammals. The model sees both the reference sequence and auxiliary features encoding how each aligned species matches, mismatches, or gaps at each base.

Training uses a masked language modeling objective analogous to BERT in natural language processing. The model randomly masks nucleotides in the reference sequence and predicts the masked base given the surrounding context and the aligned sequences. This objective encourages the model to learn evolutionary constraints: positions where substitutions are strongly disfavored across species receive confident predictions for the reference base, while unconstrained positions allow more flexibility.

13.3.2. Variant Scoring Strategies

GPN-MSA supports several approaches to deriving variant effect scores. Likelihood-based scoring compares the model's log-likelihood or probability of the reference versus alternate allele at the variant position. Variants that substantially reduce the model's confidence in the sequence are inferred to be more disruptive. Embedding distance computes representations for reference and alternate sequences and uses their difference, for example Euclidean distance, as an effect magnitude. Influence scores quantify how much a variant perturbs the model's outputs across the surrounding genomic context.

Because the model operates on whole-genome alignments, it can score coding and noncoding variants, regulatory elements, introns, UTRs, and intergenic regions. It performs particularly well in regions with complex conservation patterns where simple phyloP-like scores struggle, capturing dependencies that go beyond position-by-position conservation.

13.3.3. Benchmarking and Applications

GPN-MSA demonstrates strong performance on genome-wide pathogenic versus benign classification datasets, variant sets from genome-wide association studies, and functional readouts from high-throughput reporter assays. Its primary utility lies in genome-wide prefiltering, where it can prioritize candidate causal variants in regulatory regions, and in complementing protein-focused tools by supplying information in regions where AlphaMissense is blind.

The key limitation is dependency on high-quality multi-species alignments. Coverage and quality drop in repetitive regions, structurally complex regions, or poorly aligned segments of the genome. For variants in such regions, GPN-MSA predictions should be interpreted with caution or supplemented with other methods.

13.4. Evo 2: A Generalist Genomic Language Model

Evo 2 pushes the foundation model paradigm to an extreme: it is a genome-scale language model trained across all domains of life, including bacteria, archaea, eukaryotes, and phages, on more than

9 trillion DNA tokens (Brixi et al. 2025). Rather than specializing for any particular organism or task, Evo 2 aims to be a general-purpose genomic foundation model analogous to large language models in natural language processing.

13.4.1. Scale and Architecture

Several features distinguish Evo 2 from earlier genomic models. The model uses autoregressive training on DNA, predicting the next base given the preceding context, analogous to next-token prediction in GPT-style text language models. The architecture is StripedHyena 2, which blends convolutional and attention mechanisms to support context windows up to 1 million base pairs while remaining computationally tractable. Multiple model sizes are available, including 7 billion and 40 billion parameter variants, with open-source weights, training code, and the OpenGenome2 dataset.

The cross-species training corpus is critical. By learning from genomes across the tree of life, Evo 2 captures evolutionary patterns that span far longer timescales than human-only or mammal-only models. This breadth comes at the cost of human-specific optimization, but the trade-off enables applications in non-model organisms and provides a distinctive view of sequence constraint.

13.4.2. Zero-Shot Variant Effect Scoring

Remarkably, Evo 2 can be used for zero-shot variant interpretation without any supervised training on variant labels. For a given locus, one computes the model's sequence likelihood for the reference allele, then computes the likelihood for the alternate allele or the sequence containing it. The difference in likelihood provides a variant effect score, with variants that strongly reduce probability inferred to be more disruptive.

In benchmarks reported in the preprint and follow-up analyses, Evo 2 achieves competitive or state-of-the-art accuracy for pathogenic versus benign classification across multiple variant types, including both coding and noncoding, even without variant-specific supervised training. When a simple supervised classifier is built on Evo 2 embeddings, it reaches state-of-the-art performance on tasks like BRCA1 VUS classification.

13.4.3. Cross-Species Variant Interpretation

Because Evo 2 is trained across diverse species, it naturally supports variant effect prediction in non-model organisms such as livestock and crops. It can help quantify mutation load, prioritize variants for breeding programs, and guide genome editing designs across species. However, its generality comes with trade-offs. Domain-specific models like AlphaMissense for human missense or AlphaGenome for regulatory variants may still outperform Evo 2 on certain human-centric tasks, and careful calibration and benchmarking are required before any clinical application.

13.5. AlphaGenome: Unified Megabase-Scale Regulatory Modeling

Where Evo 2 is generalist and sequence-only, AlphaGenome is explicitly designed as a multimodal regulatory model of the human genome, with a focus on variant effect prediction across many functional readouts (Z. Avsec, Latysheva, and Cheng 2025). It represents the current frontier in regulatory variant effect prediction.

13.5.1. Architecture: Convolutions and Transformers over 1 Megabase

AlphaGenome takes as input 1 megabase of DNA sequence and produces predictions at single-base resolution for a large set of genomic tracks. These tracks include chromatin accessibility and histone marks, transcription factor binding, gene expression measured by CAGE-like signals, three-dimensional genome contacts, and splicing features including junctions and splice site usage.

The architecture combines complementary components. Convolutional layers detect local sequence motifs, analogous to the early layers of DeepSEA and its successors. Transformer blocks propagate information across the full megabase context, capturing the long-range dependencies that CNNs struggle to model. Task-specific heads output different experimental modalities across many tissues and cell types. This design generalizes earlier models like Basenji, Enformer (for regulatory tracks), and SpliceAI (for splicing) into a single unified model (Chapter 11; Chapter 7).

13.5.2. Variant Effect Prediction Across Modalities

Given a reference sequence and a candidate variant, AlphaGenome scores variant effects by predicting genome-wide functional tracks for the reference sequence, predicting the same tracks for the sequence bearing the variant, and comparing predictions to obtain delta signals across regulatory elements, splicing patterns, gene expression levels, and three-dimensional contact maps affecting enhancer-promoter communication.

On extensive benchmarks, AlphaGenome achieves state-of-the-art accuracy in predicting unseen functional genomics tracks and shows strong performance on diverse variant effect tasks including noncoding disease variants, splicing disruptions, and regulatory MPRA data. Critically, it provides mechanistic hypotheses about which tracks and tissues are disrupted rather than only a single scalar risk score. An API makes AlphaGenome accessible to the research community, enabling large-scale variant scoring without local training infrastructure.

13.6. Comparing Design Choices Across Modern VEP Models

The models surveyed in this chapter span different points in a multidimensional design space. Understanding these differences helps practitioners choose appropriate tools for their applications.

Model	Input Modality	Context Length	Pretraining Data	Variant Types	Primary Outputs
AlphaMissense	Protein sequence + structure	Protein-length	MSAs + structural environment	Missense only	Pathogenicity probability
GPN-MSA	Multi-species DNA alignments	kb-scale windows	Whole-genome MSAs (multiple species)	Coding + noncoding	Likelihood / embedding-based scores
Evo 2	Raw DNA sequence	Up to ~1 Mb	OpenGenome2 (all domains of life)	All variant types	Zero-shot likelihood-based scores
AlphaGenome	Raw DNA sequence	1 Mb	Human genome + multi-omic tracks	All variant types	Multi-omic tracks + delta effects

Several key contrasts emerge from this comparison. In terms of scope, AlphaMissense is human-missense-specific with deep clinical calibration, GPN-MSA and AlphaGenome are human-genome-centric and span coding and regulatory variants, while Evo 2 is cross-species and general-purpose. For context and long-range effects, AlphaMissense operates at protein scale, GPN-MSA uses modest windows centered on the variant, and Evo 2 and AlphaGenome support megabase-scale context capable of capturing long-range regulatory interactions. Regarding outputs, AlphaMissense and GPN-MSA primarily produce scalar scores, Evo 2 outputs likelihoods and embeddings that require task-specific postprocessing, and AlphaGenome outputs rich functional profiles that enable mechanistic hypotheses about variant effects.

13.7. Practical Use: Choosing and Interpreting Modern VEP Tools

In realistic workflows, these models are complementary rather than competing. The choice of tool depends on the variant type, the available resources, and the downstream application.

13.7.1. Coding Missense Variants

For human missense variants, AlphaMissense provides a high-coverage, clinically calibrated score that serves as an excellent starting point. This should be complemented with protein language model embeddings from ESM or related systems for gene-specific or domain-specific modeling (Chapter 9), conservation and population data including GPN-MSA scores in coding regions and gnomAD frequencies, and gene-level context such as constraint metrics and disease association history.

13.7.2. Noncoding and Regulatory Variants

For regulatory variation in promoters, enhancers, introns, and intergenic regions, AlphaGenome is currently the most comprehensive option. It provides tissue-specific changes in chromatin and expression, splicing consequences for intronic and exonic variants, and potential disruption of

long-range enhancer-promoter interactions. GPN-MSA serves as a valuable complement when a conservation-grounded score is desired, when high-quality multi-species alignments are available, or when scanning broad regions genome-wide without requiring the full multi-omic output that AlphaGenome provides.

13.7.3. Cross-Species and Large-Scale Modeling

For non-human organisms or when building general-purpose genomic tools, Evo 2 is the natural choice. Its zero-shot variant scoring works in poorly annotated species, it can guide the design and screening of genome edits, and it can serve as a feature extractor feeding downstream supervised models trained on organism-specific labels.

13.7.4. Score Interpretation and Calibration

Regardless of the model, variant effect scores should be treated as probabilistic evidence rather than binary labels. Calibration is essential: does a score of 0.9 truly correspond to approximately 90% pathogenic variants, or is the score distribution shifted? The distribution of scores within a gene matters, since outliers relative to the gene's typical score distribution are more suspect. Consistency across tools strengthens confidence, and agreement between AlphaMissense, GPN-MSA, AlphaGenome, Evo 2, and simpler conservation metrics provides more reliable evidence than any single score.

Where possible, predictions should be tied back to mechanistic hypotheses such as splice site disruption or enhancer-promoter rewiring, and experimental follow-up through targeted assays, MPRA, or CRISPR screens should be considered for variants of high clinical interest.

13.8. Open Challenges and Future Directions

Even these state-of-the-art systems leave major gaps that define the frontier of variant effect prediction research.

13.8.1. Ancestry and Population Bias

Training data and labels remain skewed toward certain ancestries, raising concerns about performance and calibration in underrepresented populations. ClinVar submissions are predominantly from European-ancestry individuals, and population frequency databases like gnomAD have similar biases. Models trained on these data may systematically misclassify variants that are common in non-European populations but rare in training data. Addressing this bias requires both more diverse data collection and explicit modeling of population structure in variant effect prediction frameworks.

13.8.2. Complex Variant Patterns

Most models focus on single-base or single-amino-acid changes. Systematic handling of haplotypes, where multiple variants on the same chromosome may interact; indels and structural variants, which can have complex functional consequences; and epistatic interactions across distant loci remains in its infancy. The combinatorial explosion of possible multi-variant genotypes makes exhaustive scoring impractical, and the training data for multi-variant effects is sparse.

13.8.3. Integrating Multi-Omics and Longitudinal Data

AlphaGenome marks a step toward unified multi-omic prediction, but dynamic phenomena including developmental trajectories, environmental responses, and time-series measurements are only lightly modeled. The static nature of current variant effect predictions misses the reality that variant effects can be highly context-dependent, varying across cell types, developmental stages, and environmental conditions.

13.8.4. Interpretability and Clinical Communication

Translating high-dimensional predictions into explanations that clinicians and patients can understand, and that map onto emerging guidelines for AI-assisted variant interpretation, remains a human-factor challenge. A score alone, no matter how accurate, is often insufficient for clinical decision-making. Clinicians need to understand why a variant is predicted to be pathogenic, which aspects of the prediction are most certain, and how the prediction relates to the specific clinical question at hand.

13.8.5. Safe Deployment and Continual Learning

As more functional datasets and clinical labels accumulate, models will need continual updating without catastrophic forgetting, along with governance frameworks to track model versions and provenance. The rapid pace of model development creates challenges for clinical integration, where stability and reproducibility are paramount.

In subsequent chapters, we will connect these VEP systems to broader issues in evaluation (Chapter 15), confounding and bias (Chapter 16), and interpretability (Chapter 17), positioning them within the broader landscape of genomic foundation models. The models described here illustrate how the building blocks from earlier chapters, including NGS data, functional genomics, CNNs, transformers, and protein and DNA language models, coalesce into powerful end-to-end systems for variant interpretation.

14. Multi-omics & Systems Context

Warning

TODO:

- Add figure: Multi-omics integration strategies diagram showing early, intermediate, and late fusion approaches with representative models
- Add figure: CpGPT architecture schematic showing masked modeling objective, sample embeddings, and downstream task adaptation
- Add figure: GLUE framework diagram illustrating modality-specific VAEs, feature graph structure, and alignment objectives
- Add figure: GNN-based cancer subtyping comparison showing MoGCN patient-level graphs vs. CGMega gene-level modules
- Add figure: DeepRVAT set-based architecture showing variant aggregation into gene-level impairment scores
- Add figure: G2PT hierarchical structure from variants → genes → systems → phenotypes
- Add table: Comparison of multi-omics integration methods (GLUE, MoGCN, CGMega, etc.) with columns for input modalities, graph type, primary use case, and scalability
- Add table: Design patterns summary showing the five key patterns with representative methods and typical applications
- Consider adding conceptual diagram showing the trajectory from single-omic models toward whole-patient foundation models

Modern genomic foundation models excel at learning from sequences, structures, or single-omic profiles in isolation. Yet most complex traits arise from systems-level interactions: genetic variants perturb molecular networks, networks span multiple omics layers, and these layers interact with environment, development, and clinical context. A model that sees only one layer rarely captures the full story.

This chapter surveys how deep learning extends beyond single-omics to integrate methylation, chromatin, expression, protein, and clinical data into unified representations. As the final chapter of Part IV, it serves as a bridge from model-centric architecture design to systems-level, clinically grounded applications in Part VI. The methods introduced here illustrate emerging design patterns for systems-aware genomic foundation models that move from isolated sequences toward whole-patient representations.

We examine several archetypal systems that represent different facets of this integration challenge. CpGPT demonstrates how foundation modeling principles apply to DNA methylation, treating the methylome as a sequence-like object amenable to transformer-based pretraining. GLUE and its single-cell variant SCGLUE show how graph-linked embeddings can align cells across modalities when different omics are measured in different cells. Graph neural network approaches to cancer

subtyping, including MoGCN and CGMega, illustrate how patient similarity networks and gene-level graphs can integrate genomics, transcriptomics, and proteomics for classification and biomarker discovery. DeepRVAT, NeEDL, and the Genotype-to-Phenotype Transformer address rare variants and epistasis, effects that linear PGS models largely miss. Finally, deep learning frameworks for polygenic risk and fine-mapping, including Delphi and MIFM, extend the PGS paradigm from Chapter 3 with nonlinear architectures and foundation model features.

Together, these approaches point toward a future where genomic models reason across biological scales, from single nucleotides through molecular networks to whole-patient phenotypes.

14.1. Why Single-omics Models Are Not Enough

Earlier chapters emphasized how sequence-based models can predict variant effects from local DNA or protein context. These models already improve causal variant prioritization and polygenic risk scoring. However, they typically assume a narrow view of biology.

Most sequence models operate on a single molecular layer. A convolutional network or transformer may see only DNA sequence, or only expression values, without access to the other layers that mediate the flow of genetic information. Even when multiple outputs are predicted simultaneously, as in multi-task models like Enformer, the input remains a single modality.

Many downstream uses treat variant effects as additively summing across loci. The PGS framework from Chapter 3 exemplifies this assumption: effects of individual variants are estimated independently and combined through weighted sums. While linear models have well-understood statistical properties and interpretability, they cannot capture interactions between variants or between molecular layers.

Models rarely account for dynamic cellular state. The same sequence may have different regulatory consequences depending on cell type, developmental stage, or environmental exposure. Static sequence-to-function models provide context-averaged predictions that may not reflect biology in any particular condition.

Real diseases violate all three of these assumptions. Regulation is inherently multi-layered: genetic variants alter chromatin accessibility and DNA methylation, which modulate transcription, which affects splicing and translation, which determines protein levels and modifications. A variant's consequences propagate through this cascade in ways that single-layer models cannot fully capture.

Effects are context-dependent. A variant might be benign in one tissue and pathogenic in another, depending on which genes are expressed, which transcription factors are present, and how the local chromatin environment is configured. The same variant in different individuals may have different consequences depending on the genetic background and cellular states in which it operates.

Risk is combinatorial. Epistasis and pathway-level perturbations play significant roles in many complex traits. Two variants that individually have small effects might together strongly perturb a pathway, or might cancel each other out. Linear models assume effects are independent and additive, missing these interaction structures entirely.

Chapter 3 highlighted the "missing heritability" and limited cross-ancestry portability of traditional GWAS and linear PGS, motivating sequence-based deep learning. This chapter takes the next step: combining sequence-derived features with multi-omics and systems-level models that better reflect biological organization.

14.2. Foundations of Multi-omics Integration

Multi-omics data come in several flavors that present different integration challenges. Bulk-level profiles such as GWAS variants, bulk RNA-seq, and bulk proteomics aggregate signals across millions of cells, providing population-level or tissue-averaged views. Single-cell modalities including scRNA-seq, scATAC-seq, multiome assays, and spatial omics resolve cellular heterogeneity but introduce sparsity and technical noise. Epigenetic readouts such as DNA methylation arrays, histone modification ChIP-seq, and chromatin conformation capture provide orthogonal views of regulatory state. Clinical and environmental covariates from electronic health records, laboratory measurements, and lifestyle questionnaires add non-molecular dimensions that influence phenotypes.

Integration strategies for combining these data types typically fall into three broad categories that represent different trade-offs between architectural complexity and the ability to capture cross-modal structure.

Early fusion, also called feature-level integration, concatenates normalized features from multiple omics and feeds them into a single model. This approach is straightforward to implement and allows the model to learn arbitrary interactions between features. However, early fusion is sensitive to differences in scale and dimensionality between modalities, handles missing data poorly since any sample lacking one modality must be imputed or excluded, and can be dominated by whichever modality has the most features or highest signal-to-noise ratio.

Intermediate fusion, also called shared latent space integration, learns modality-specific encoders that map each omic into a common embedding space. Alignment between modalities is encouraged through reconstruction losses that require each encoder's latent representation to support decoding back to its original features, contrastive terms that pull together representations of the same biological entity across modalities, or graph constraints that enforce consistency with known biological relationships. Intermediate fusion is the dominant design in modern multi-omics deep learning because it handles missing modalities gracefully (only the available encoder needs to fire), allows modality-specific preprocessing and architectures, and can incorporate biological prior knowledge through the alignment objectives.

Late fusion, also called prediction-level integration, trains separate models for each modality and combines their outputs through ensemble methods or a meta-model. This approach is robust to missing modalities since each sub-model operates independently, and it allows each modality to use whatever architecture works best for its data type. However, late fusion may underutilize cross-omic structure that could inform predictions, since interactions between modalities can only be captured at the final combination stage.

Modern frameworks like GLUE and multi-omics graph neural networks predominantly adopt intermediate fusion, often augmented with graphs that encode known or inferred biological relationships. Gene-peak edges in single-cell multi-omics link chromatin accessibility peaks to the genes they regulate. Gene-transcription factor edges connect genes to the factors that bind their promoters and enhancers. Protein-protein interaction edges capture physical and functional relationships. Sample similarity edges connect patients or cells with similar molecular profiles. The rest of this chapter traces how these design choices implement systems-level reasoning in practice.

14.3. CpGPT: A Foundation Model for DNA Methylation

14.3.1. Methylation as a Systems Hub

DNA methylation occupies a privileged position in the regulatory hierarchy, sitting at a junction between genotype, environment, and phenotype. Methylation patterns integrate genetic influences, since sequence context affects which CpG sites can be methylated and polymorphisms can create or destroy CpG dinucleotides. They also integrate developmental programs, since methylation landscapes are extensively remodeled during differentiation and establish cell-type-specific regulatory states. Environmental exposures including diet, smoking, toxins, and stress leave lasting methylation signatures that persist long after the exposure ends.

Beyond serving as an integrative readout, methylation encodes rich information about cellular identity and state. Cell types can be distinguished by their methylation profiles, and within a cell type, methylation captures information about age, health status, and disease risk. Epigenetic clocks built from methylation data predict chronological age with remarkable accuracy, and deviations from predicted age correlate with mortality risk and disease burden (Camillo et al. 2024).

Traditional methylation models have been task-specific: one model for age prediction, another for mortality risk, another for tissue classification. Each model is trained from scratch on labeled data for its particular task, learning whatever methylation patterns happen to be predictive without necessarily capturing general structure. CpGPT reframes methylation as a foundation modeling problem, using large-scale pretraining to learn representations that transfer across tasks.

14.3.2. Architecture and Pretraining

CpGPT, the Cytosine-phosphate-Guanine Pretrained Transformer, treats methylomes as sequences or sets of CpG sites and uses transformer-style self-attention to model their structure (Camillo et al. 2024). The model was pretrained on over 1,500 DNA methylation datasets encompassing more than 100,000 samples from diverse tissues and conditions.

Several aspects of methylation structure make it amenable to transformer modeling. Local CpG correlations arise because nearby CpG sites tend to share methylation status, particularly within CpG islands. Long-range coordination reflects the fact that methylation patterns at distant genomic regions can be correlated through shared regulatory programs or chromatin compartmentalization. Global sample-level variation captures the systematic differences between samples that reflect tissue identity, age, disease status, and other biological variables.

CpGPT uses masked modeling objectives analogous to BERT-style language model pretraining. During training, a subset of CpG methylation values is masked, and the model learns to reconstruct them from the surrounding context. This forces the model to learn relationships between CpG sites and to capture the statistical structure of methylation profiles.

Multi-task pretraining provides additional signal. Auxiliary objectives such as array platform conversion, where the model learns to translate between different methylation measurement technologies, and reference mapping, where the model learns to align samples to reference profiles, encourage the model to learn robust representations that generalize across technical and biological variation.

The result is a sample-level embedding, analogous to a CLS token representation in language models, that provides a compact, task-agnostic summary of each sample's methylome. This embedding can be used directly for downstream prediction or as input to more complex models.

14.3.3. Zero-shot and Fine-tuned Tasks

Because CpGPT is trained on diverse cohorts spanning many tissues and conditions, it exhibits zero-shot or few-shot generalization to new tasks. For imputation and array conversion, the model can fill in missing CpG values or translate between different methylation array platforms, enabling harmonization of datasets collected with different technologies. For chronological age prediction, the pretrained model yields age estimates that match or exceed specialized epigenetic clocks, even without task-specific fine-tuning. For mortality risk prediction, CpGPT achieved state-of-the-art performance in the Biomarkers of Aging Challenge. Sample classification tasks such as distinguishing tissues, disease states, or exposure profiles also benefit from the learned representations.

In a multi-omics context, CpGPT-derived embeddings serve several roles. They can be inputs to downstream predictors, providing rich methylation features for risk scores, prognosis models, or treatment response prediction. They can function as one modality in a shared latent space that also includes expression, proteomics, and other data types. They can inject epigenetic state information into otherwise sequence-centric genomic foundation models, providing context about cellular identity and regulatory status.

Conceptually, CpGPT exemplifies a single-omic foundation model designed to plug into multi-omics architectures. The pretraining objective learns general methylation structure, and the resulting embeddings can be combined with other modalities for tasks that require systems-level reasoning.

14.4. GLUE: Graph-linked Unified Embedding for Single-cell Multi-omics

14.4.1. The Unpaired Single-cell Integration Problem

Single-cell experiments often profile different modalities in different cells. A typical study might include scRNA-seq data from one set of cells, scATAC-seq data from another set, and perhaps a small subset with both modalities measured simultaneously through multiome protocols. The central challenge is building a unified atlas that aligns these cells in a common space, recovers cell types and trajectories, and infers regulatory networks connecting chromatin to expression (Cao and Gao 2022).

This problem is harder than standard data integration because the feature spaces are entirely different. RNA-seq measures gene expression across roughly 20,000 genes. ATAC-seq measures chromatin accessibility across hundreds of thousands of peaks. There is no direct correspondence between features: a gene is not the same object as a peak. Aligning cells across modalities requires reasoning about how features in one modality relate to features in another.

Previous approaches addressed this through explicit feature conversion, for example by assigning ATAC-seq peaks to nearby genes and treating the resulting gene-level accessibility as comparable to expression. This conversion is straightforward but loses information, since the detailed structure of chromatin accessibility within a gene's regulatory region is collapsed into a single number. It also introduces arbitrary choices about how to define gene-peak assignments.

GLUE, Graph-Linked Unified Embedding, addresses this problem by combining modality-specific encoders with a graph of biological prior knowledge linking features across omics (Cao and Gao 2022).

14.4.2. Architecture

GLUE consists of three key components that work together to align cells across modalities while respecting biological relationships between features.

Modality-specific variational autoencoders provide the foundation. Each omic has its own encoder-decoder pair. Encoders map cells to a low-dimensional latent embedding, and decoders reconstruct modality-specific features from these embeddings. The variational structure encourages smooth, interpretable latent spaces.

A feature graph encodes biological relationships between features across modalities. Genes, peaks, and motifs form nodes in this graph. Edges capture relationships: a peak linked to a gene's promoter or a distal enhancer predicted to regulate the gene, or a transcription factor binding motif whose presence in a peak suggests regulation of nearby genes. A graph neural network learns feature embeddings that are consistent with this graph structure, ensuring that biologically related features have similar representations.

Alignment objectives tie the components together. Loss terms encourage the cell latent spaces from different modalities to align, so that RNA-only cells and ATAC-only cells with similar biological states end up near each other in the shared embedding space. The feature embeddings from the graph neural network are tied to the cell embeddings through the generative decoders, enforcing consistency between the learned representations and the prior biological knowledge.

The result is a unified embedding in which cells from multiple modalities can be jointly clustered, visualized, and used for downstream analysis. Cell type labels transfer across modalities: once cell types are annotated in one modality, the alignment allows those annotations to propagate to cells from other modalities.

14.4.3. Applications

The GLUE framework has demonstrated strong performance across several challenging applications. Multi-omics integration at single-cell resolution has been achieved for combinations of RNA, ATAC, methylation, and protein data. The graph structure provides a natural framework for regulatory network inference, linking chromatin features to gene expression through the learned feature relationships. Atlas construction over large cohorts has benefited from GLUE's ability to align datasets across laboratories, technologies, and biological conditions, in some cases correcting earlier annotation errors.

From the perspective of genomic foundation models, GLUE exemplifies graph-guided multi-modal pretraining. Modality-specific encoders learn representations of their respective data types, and the graph structure provides a principled way to align these representations across modalities. The framework is modular: new modalities can be added by training new encoders and connecting them to the feature graph, without retraining the entire system.

14.5. GNN-based Multi-omics Cancer Subtyping

Cancer is inherently a multi-omic disease. Driver mutations, copy number alterations, epigenetic reprogramming, and transcriptional rewiring jointly define tumor subtypes with distinct prognosis and treatment response. Models that integrate these layers can capture biological structure that single-omic approaches miss.

14.5.1. MoGCN: Patient Graphs from Multi-omics

MoGCN represents a graph-convolutional framework for cancer subtype classification that integrates genomics, transcriptomics, and proteomics (X. Li et al. 2022). The key insight is that patients can be represented as nodes in a graph, with edges encoding similarity relationships derived from their multi-omic profiles.

The architecture proceeds in stages. First, each omic is processed by an autoencoder that reduces dimensionality and learns compressed representations. Second, a patient similarity network is constructed by measuring similarity between patients based on their multi-omic features. Third, graph convolutional layers operate on this patient graph, learning node embeddings that incorporate both the patient's own features and information from similar patients. Finally, a classifier operating on these graph-enhanced embeddings predicts cancer subtypes.

This design captures several important properties. The patient similarity graph allows information to flow between related samples, improving predictions for patients whose individual profiles might be ambiguous but whose neighbors provide context. The multi-view structure, with separate processing for each omic followed by combination, allows each data type to contribute according to its informativeness. The graph convolutional layers can learn which neighbors are most relevant for classification.

MoGCN achieved strong performance on breast cancer subtype classification using TCGA data, outperforming methods that treated each sample independently or that combined omics through simple concatenation. Feature extraction from the trained model highlighted biologically meaningful genes and pathways, including epidermal development, cell migration, Wnt signaling, and ErbB signaling pathways for different subtypes. The patient similarity network provided clinically intuitive structure, with clear separation between subtypes and interpretable relationships between patients.

14.5.2. CGMega: Multi-omics Cancer Gene Modules

Where MoGCN focuses on patient-level graphs, CGMega operates on gene-level graphs that capture multi-omic relationships between genes (Hao Li et al. 2024). Nodes represent genes, and edges encode relationships from multiple data sources: co-expression from RNA-seq, correlation in copy number alterations, shared methylation patterns, and physical proximity in three-dimensional chromatin organization.

A graph attention network learns cancer gene modules, which are subsets of genes that co-vary across omics and associate with cancer phenotypes. This module-centric view aligns with systems biology intuitions: rather than seeking single-gene markers, CGMega identifies network-level signatures that reflect pathway dysregulation. The attention mechanism highlights which edges and neighbors are most important for each gene's module membership, providing interpretability.

Gene modules discovered by CGMega capture known cancer biology and reveal new associations. Modules enriched for cell cycle genes associate with proliferative subtypes. Modules involving immune signaling genes distinguish immunologically active from quiescent tumors. The multi-omic construction ensures that these modules reflect coordinated changes across molecular layers rather than artifacts of any single data type.

14.5.3. Design Patterns and Alternatives

A growing ecosystem of multi-omics subtyping methods uses related architectural patterns. Contrastive learning approaches learn sample embeddings by encouraging similar samples to have similar representations while pushing dissimilar samples apart. Generative models including variational autoencoders and generative adversarial networks jointly model multiple omics for unsupervised clustering, learning latent spaces that capture shared and modality-specific variation. Transformer-based hybrids blend multi-layer perceptrons and attention mechanisms for high-dimensional omics, using self-attention to capture feature interactions.

Common themes emerge across these methods. Modality-specific encoders with shared latent spaces appear repeatedly, allowing flexible handling of missing modalities while enabling cross-modal interactions. Graphs capturing patient-patient or gene-gene relationships structure the learning problem and provide interpretability. Emphasis on biological interpretability through clusters, modules, or attention patterns helps translate model outputs into biological hypotheses.

These cancer subtyping models illustrate how multi-omics integration naturally leads to graph-structured genomic foundation models. Sequences, epigenetics, and expression become nodes in learned biological networks, and the models learn to reason over these networks rather than treating each measurement in isolation.

14.6. Rare Variants and Epistasis in Systems Context

Chapter 3 discussed how standard PGS methods largely ignore rare variants and epistatic interactions, despite their importance for individual-level risk and disease mechanism. Rare variants, though individually uncommon, collectively explain substantial phenotypic variance and often have larger effect sizes than common variants. Epistasis, the non-additive interaction between variants, is theoretically expected from network biology and has been documented empirically for many traits. Multi-omics and systems models offer a framework to incorporate these effects more effectively than linear approaches.

14.6.1. DeepRVAT: Set-based Rare Variant Burden Modeling

DeepRVAT, Deep Rare Variant Association Testing, addresses a fundamental statistical challenge: rare variants have too few carriers to achieve individual statistical significance, yet collectively they carry important phenotypic information (Clarke et al. 2024). Traditional burden tests collapse all rare variants in a gene into a single count, losing information about variant severity. DeepRVAT instead learns gene-level impairment scores from variant annotations using set neural networks.

The architecture treats each gene's rare variants as an unordered set, reflecting the biological reality that the order of variants along a gene is not informative for their combined effect. Each variant is characterized by a vector of annotations including predicted functional impact, conservation, and structural features. A permutation-invariant neural network aggregates these annotations into a gene-level impairment score.

Crucially, DeepRVAT learns trait-agnostic representations. The gene impairment scores are trained to be predictive across multiple phenotypes simultaneously, which provides regularization and enables transfer to new traits. This multi-task learning encourages the model to learn biologically meaningful notions of gene damage rather than overfitting to any single phenotype.

The result improves both gene discovery and risk prediction. For gene discovery, DeepRVAT identifies more significant gene-trait associations than linear burden tests, particularly for genes where variant effects are heterogeneous. For risk prediction, the learned impairment scores identify individuals with high rare variant burden across multiple genes, enabling personalized risk assessment that linear PGS cannot capture.

DeepRVAT bridges the gap between variant-level annotations and gene-level burden, making it naturally compatible with sequence-based variant effect models from earlier chapters. Annotations from models like SpliceAI, AlphaMissense, or DNA foundation models can serve as input features, and the set neural network learns how to combine them into predictive gene-level scores.

14.6.2. NeEDL: Network-based Epistasis Detection

NeEDL, Network-based Epistasis Detection via Local search, addresses the complementary challenge of identifying epistatic interactions among variants (**kessler_needl_2023?**). The search space for epistasis is enormous: even considering only pairwise interactions among a million variants yields approximately 500 billion pairs to test. NeEDL uses network structure and optimization algorithms to make this search tractable.

The approach builds on network medicine principles. Genes and variants are embedded in a network based on biological prior knowledge, including protein-protein interactions, pathway membership, and co-expression relationships, as well as GWAS signals that suggest which variants influence the trait. Local search strategies explore combinations of variants that are close in this network and that jointly influence disease.

The optimization uses quantum-inspired algorithms that efficiently explore the combinatorial space of variant combinations. Rather than exhaustively testing all pairs or higher-order combinations, the search focuses on biologically plausible interaction sets defined by network proximity.

NeEDL does not operate as a full genomic foundation model, but it points toward systems-level combinatorial reasoning that future GFMs will need to support. The network structure provides biological constraints that make the epistasis search feasible, and the discovered interactions map onto interpretable pathways and cellular processes.

14.6.3. G2PT: Hierarchical Genotype-to-Phenotype Transformers

G2PT, Genotype-to-Phenotype Transformer, explicitly models the hierarchical structure connecting variants to phenotypes (Lee et al. 2025). Rather than treating variants as independent features to be weighted and summed, G2PT organizes variants into genes, genes into systems such as pathways and tissues, and systems into phenotype predictions.

The architecture uses transformer blocks at each level of this hierarchy. Variant-level attention captures interactions between variants within a gene. Gene-level attention captures interactions between genes within a system. System-level attention captures how different pathways and tissues contribute to phenotype risk.

Prior biological knowledge structures these attention patterns. Gene-pathway membership from databases like KEGG and Reactome defines which genes belong to which systems. Tissue expression patterns from GTEx indicate where each gene is active. These priors constrain the attention patterns, ensuring that the model learns biologically plausible interaction structures rather than arbitrary statistical correlations.

The hierarchical structure provides interpretability. After training, attention weights can be examined to understand which variants, genes, and systems most strongly contribute to risk for a given individual. This enables explanations like "high risk is driven by variants in genes A and B that together perturb pathway X in tissue Y."

G2PT can be viewed as an early example of a systems-aware genomic foundation model for genotype data. It unifies additive and interaction effects within a single deep architecture, using prior knowledge to guide learning toward biologically meaningful structure.

14.7. Deep Learning-enhanced Polygenic Risk and Fine-mapping

Chapter 3 framed polygenic scores as linear weighted sums of variant effects. This approach has attractive statistical properties including interpretability, efficiency, and well-characterized uncertainty. However, it misses nonlinear effects, cannot incorporate rich sequence-based features, and struggles with rare variants and cross-ancestry generalization. Deep learning extends the PGS paradigm along each of these dimensions.

14.7.1. Deep-learning PGS Frameworks

Deep-learning PGS frameworks like Delphi replace the linear combination of variant effects with flexible neural networks that learn complex functions of genotype and covariates (Georgantas, Kutalik, and Richiardi 2024).

The key technical contribution is enabling neural networks to handle genome-wide inputs. A typical GWAS includes hundreds of thousands to millions of variants, far more than can be naively input to a neural network. Delphi addresses this through efficient architectures that can process hundreds of thousands of variants while remaining computationally tractable.

The resulting models can capture dominance effects where heterozygotes differ from the midpoint of homozygotes, epistatic interactions where variant effects depend on genetic background, and gene-environment interactions where variant effects depend on non-genetic covariates. These effects

are learned from data rather than specified a priori, allowing the model to discover whatever structure best predicts the phenotype.

Empirical evaluations demonstrate improved discrimination compared to linear PGS across several traits, with the gains being largest for traits where nonlinear effects are most important. Importantly, Delphi also shows improved cross-ancestry generalization: the learned representations transfer more effectively than linear weights to populations not well represented in training data.

From a systems perspective, deep-learning PGS frameworks represent a move toward whole-patient risk modeling. While still primarily based on genotype plus covariates without explicit multi-omics integration, they demonstrate that the linear PGS paradigm can be extended to capture more biological complexity.

14.7.2. MIFM and Multi-ancestry Fine-mapping

Fine-mapping addresses a fundamental challenge in human genetics: GWAS identifies loci but cannot usually pinpoint causal variants. Within each associated locus, linkage disequilibrium means that many variants are correlated with the causal variant and show similar association signals. Fine-mapping methods attempt to distinguish causal variants from these correlated passengers.

Multiple-instance fine-mapping frameworks like MIFM address the key bottleneck that per-variant causal labels are rarely available (Rakowski and Lippert 2025). Instead, we typically know only that some variant or variants within a locus are causal. This is naturally framed as a multiple-instance learning problem: each locus is a "bag" of variants, loci with significant GWAS signals form positive bags, and a model learns to identify which variants within positive bags are responsible for the signal.

Deep sequence models provide per-variant features that inform fine-mapping. Predicted effects on chromatin accessibility, transcription factor binding, gene expression, and splicing from models described in earlier chapters create a rich characterization of each variant's functional potential. MIFM-type frameworks integrate these sequence-based priors with GWAS evidence to produce more accurate causal variant identification.

Multi-ancestry data provide additional resolution. Different populations have different LD patterns, so a variant that is correlated with many others in one population may be more isolated in another. Methods that jointly analyze multi-ancestry data can leverage these differences to refine fine-mapping, and deep learning provides flexible frameworks for combining signals across populations with different genetic backgrounds.

Connections to the rest of this book are direct. Variant effect predictors from Chapter 5, Chapter 6, Chapter 7, and Chapter 13 supply per-variant features. Multi-omics models from this chapter provide functional priors about regulatory activity, methylation, and chromatin accessibility. MIFM-type frameworks integrate these priors with GWAS evidence to produce more accurate, ancestry-aware fine-mapping that identifies the variants most likely to be causal.

14.8. Design Patterns for Multi-omics and Systems GFMs

Drawing these examples together reveals several design patterns that recur across systems-level genomic foundation models. These patterns provide a conceptual vocabulary for understanding existing methods and designing new ones.

Modality-specific encoders with shared latent spaces appear in CpGPT, GLUE, and many multi-omics subtyping models. Each omic has its own encoder architecture tailored to its data characteristics, whether that involves treating methylation as a sequence, using variational autoencoders for scRNA-seq, or applying graph convolutions to patient similarity networks. These modality-specific encoders map into a common embedding space where downstream tasks operate. This design supports flexible inference with missing modalities, since only the available encoders need to fire, and allows incremental addition of new data types by training new encoders without retraining existing components.

Graph-guided integration structures learning through biological prior knowledge. GLUE's feature graph links peaks to genes and transcription factors. CGMega's gene-level graphs encode multi-omic relationships. NeEDL's epistasis networks capture pathway structure and protein interactions. Graph neural networks, graph transformers, and attention mechanisms over graph edges provide natural tools for encoding these biological networks and learning representations that respect network structure.

Hierarchical modeling captures the organization of biological systems across scales. G2PT formalizes the hierarchy from variants to genes to systems to phenotypes. Similar hierarchies can be defined for omics layers: sequence gives rise to chromatin state, which influences methylation patterns, which affect transcription, which determines protein levels, which ultimately connect to clinical traits. Architectures that respect this hierarchy can learn more interpretable and generalizable representations than flat models that treat all features equivalently.

Set-based and bag-based learning handles collections of variants or features that lack natural ordering. DeepRVAT treats variants within a gene as an unordered set, using permutation-invariant architectures to aggregate them into gene-level scores. MIFM treats variants within a fine-mapping locus as a bag, learning to identify causal variants without explicit per-variant labels. This pattern is crucial when sample sizes are large, labels are sparse, and biological order is meaningless.

Foundation pretraining with task-specific adaptation follows the broader paradigm that defines foundation models. CpGPT is pretrained on massive methylation datasets covering diverse tissues and conditions, then adapted through fine-tuning or linear probing to specific tasks like age prediction or mortality risk. This pattern could extend to multi-omics pretraining, where models learn joint representations of sequence, chromatin, methylation, expression, and clinical data before specialization for particular applications.

These patterns collectively point toward general-purpose systems GFMs that can ingest heterogeneous biological data and output risk predictions, mechanistic hypotheses, or treatment recommendations. The field is not yet at this stage, but the methods surveyed in this chapter demonstrate the building blocks.

14.9. Practical Pitfalls and Considerations

Despite impressive progress, multi-omics and systems GFMs face particular challenges that practitioners must navigate. The issues examined in depth in Chapter 16 take on special importance when combining multiple data sources.

Batch effects and platform heterogeneity are endemic to multi-omics data. Different omics layers often come from different assays, laboratories, or time points. Sequencing depth varies between samples. Array platforms measure different subsets of features with different technical characteristics. Integration methods can inadvertently encode batch structure rather than biology if batch effects are not properly addressed. The problem is particularly acute when different modalities have different batch structures, since standard single-modality batch correction methods may not apply.

Sample size and missingness pose related challenges. Multi-omics datasets are typically smaller than single-omic datasets because the cost and complexity of generating multiple data types limit the number of samples that can be profiled. Many samples lack certain modalities entirely, requiring robust handling of missing data. Methods that require all modalities for every sample exclude large fractions of available data, while methods that impute missing modalities must avoid introducing artifacts.

Population diversity and fairness concerns that apply to PGS (Chapter 3) are amplified in multi-omics settings. Most large multi-omics datasets come from European-ancestry populations in high-resource healthcare systems. Models trained on these data may perform poorly or behave differently in other populations. Multi-omics GFMs risk amplifying disparities if trained primarily on non-representative cohorts, since the richer feature sets provide more opportunity for overfitting to population-specific patterns.

Evaluation complexity increases with the number of modalities and the breadth of potential applications. Multi-omics models can be evaluated at many levels: predictive performance on held-out data, biological consistency of learned representations with known biology, plausibility of inferred networks compared to experimental validation, and clinical utility when deployed in real-world settings. Overfitting to proxy metrics that are easy to compute may not translate to performance on the metrics that ultimately matter.

Interpretability and causal inference remain challenging. Attention scores and feature importance values provide some insight into model behavior, but they are not guarantees of causal mechanism. A model might attend to a feature because that feature is causal, or because it is correlated with something causal, or for spurious reasons related to batch effects or data collection. Integrating deep models with perturbation data from CRISPR screens and gene knockouts, and with robust causal inference frameworks, remains an open frontier.

Careful experimental design, thoughtful validation, and transparent reporting are therefore especially crucial for multi-omics GFMs. The additional complexity of multi-modal data creates additional opportunities for both insight and error.

14.10. Outlook: Toward Whole-patient Foundation Models

The methods in this chapter sketch an endgame for genomic deep learning that extends far beyond sequence-only models. The trajectory moves through several stages that the book has traced across

its chapters.

Genome-wide variant and sequence representation through hybrid CNN, transformer, and state-space model architectures established the foundation in Chapter 5 through Chapter 11. These models learn rich representations of sequence that capture regulatory grammar, variant effects, and long-range dependencies.

Multi-omics integration through graph-guided latent spaces adds new dimensions. CpGPT brings methylation into the foundation model paradigm. GLUE and related methods enable principled combination of modalities measured in different cells or samples. MoGCN and CGMega demonstrate how graph neural networks can integrate patient-level or gene-level multi-omic data for cancer subtyping and biomarker discovery.

Systems-level reasoning about rare variants and epistasis addresses effects that linear models miss. DeepRVAT learns gene-level impairment from rare variant sets. NeEDL searches for epistatic interactions guided by network structure. G2PT provides hierarchical models that explicitly represent the flow from variants through genes and pathways to phenotypes.

Clinically oriented risk modeling with deep PGS and fine-mapping connects genomic representations to patient outcomes. Delphi-like frameworks extend PGS to capture nonlinear effects and improve cross-ancestry generalization. MIFM-like methods integrate sequence-based variant features with GWAS evidence for more accurate fine-mapping.

A future whole-patient foundation model might unify all these threads. Such a model would jointly encode genotype, methylome, chromatin state, expression, proteomics, imaging, and electronic health record data. It would provide unified representations across tissues, cell types, and time points, capturing the dynamic nature of biological state. It would offer calibrated, equitable predictions of disease risk and treatment response across diverse populations. It would support mechanistic queries like "which pathways mediate this variant's effect in this tissue?" or "which interventions might counteract rare variant burden in this patient?"

Realizing this vision will require advances across multiple fronts. Data sharing and privacy-preserving learning must enable training on sensitive multi-omic and clinical data at scale. Scalable architecture design must handle the computational demands of truly multi-modal foundation models. Causal validation must distinguish correlative patterns from mechanistic understanding. Equity and fairness considerations must guide data collection and model development from the outset.

The methods surveyed here show that moving beyond single-omics is not merely incremental improvement but a qualitative change in what kinds of questions genomic models can address. The path from isolated sequence models to systems-level, clinically actionable genomics is becoming visible, even if substantial work remains to traverse it.

14.11. Summary

This chapter has surveyed how deep learning extends beyond single-omics to integrate methylation, chromatin, expression, protein, and clinical data into unified representations. We examined CpGPT as a foundation model for DNA methylation that treats methylomes as sequences amenable to transformer-based pretraining. GLUE demonstrated how graph-linked embeddings can align single-cell measurements across modalities when different omics are profiled in different cells. Graph neural network approaches including MoGCN and CGMega showed how patient-level and gene-level

graphs can integrate genomics, transcriptomics, and proteomics for cancer subtyping. DeepRVAT, NeEDL, and G2PT addressed rare variants and epistasis through set-based architectures and hierarchical modeling. Deep learning frameworks for polygenic risk and fine-mapping extended the PGS paradigm with nonlinear architectures and foundation model features.

Several design patterns emerged as common threads: modality-specific encoders with shared latent spaces, graph-guided integration, hierarchical modeling, set-based learning, and foundation pretraining with task-specific adaptation. These patterns provide conceptual vocabulary for understanding existing methods and designing new ones.

Practical challenges including batch effects, sample size limitations, population diversity, and evaluation complexity require careful attention. But the trajectory toward whole-patient foundation models that jointly encode multiple omics and clinical data is becoming clear.

The remaining chapters in Part V will address cross-cutting issues of evaluation, confounding, and interpretability that apply across all the models surveyed in this book. Part VI will then explore how genomic foundation models, including the multi-omics approaches from this chapter, translate into clinical practice.

Part V.

Part V: Reliability & Interpretation

15. Model Evaluation & Benchmarks

> ⚠ Warning
>
> **TODO:**
>
> - Add figure: evaluation pyramid visualization showing molecular → variant → trait → clinical levels with example tasks at each
> - Add figure: data splitting strategies diagram comparing random splits vs chromosome-based vs ancestry-based vs gene-family splits
> - Add figure: calibration plots showing well-calibrated vs poorly-calibrated pathogenicity predictions
> - Add table: metric families summary with columns for metric type, typical tasks, advantages, and limitations
> - Add figure: benchmark leakage examples showing common overlap patterns between training and evaluation sets
> - Consider adding schematic of foundation model evaluation regimes (zero-shot → probing → fine-tuning)

By now, we have seen genomic models operating at almost every scale. Variant calling from NGS reads (Chapter 1), polygenic scores and GWAS (Chapter 3), deleteriousness scores and variant effect predictors (Chapter 4; Chapter 13), CNN-based sequence-to-function models (Chapter 5 through Chapter 7), and genomic language models and foundation models (Chapter 10; Chapter 12) have each introduced their own metrics and benchmarks. Clinical risk prediction and pathogenic variant discovery (Chapter 18; Chapter 19) add still more evaluation considerations. What has been missing is a single place to answer a deceptively simple question: what does it mean for a genomic model to "work," and how should we systematically evaluate it?

This chapter provides that unifying view. We describe the major families of evaluation metrics and show how they map to typical genomic tasks. We organize evaluation across four levels, from molecular readouts through variant-level predictions to trait-level risk scores and finally to clinical decisions. We discuss data splitting, leakage, and robustness, the mechanics that make or break benchmarks regardless of how sophisticated the underlying architecture may be. We explain how to evaluate foundation models across different usage regimes, from zero-shot scoring through linear probing to full fine-tuning. Finally, we connect evaluation to the broader theme of reliability, linking forward to the detailed treatments of confounders in Chapter 16 and interpretability in Chapter 17.

Throughout, the theme is that architecture and scale matter, but evaluation choices often matter more. A state-of-the-art model evaluated on a leaky benchmark tells us less than a modest model evaluated on a clean one. A foundation model that achieves impressive perplexity but fails to improve downstream variant interpretation has not demonstrated clinical utility. Getting evaluation

right is prerequisite to knowing whether any of the sophisticated methods covered in this book actually work.

15.1. Evaluation as a Multi-Scale Problem

Genomic models are deployed at very different scales, and understanding this hierarchy is essential for designing appropriate evaluations. It helps to keep a simple mental pyramid in mind, with molecular readouts at the base and clinical decisions at the apex.

At the molecular and regulatory level, models take local sequence and epigenomic context as input and predict outputs such as chromatin accessibility, histone marks, transcription factor binding, splicing outcomes, or expression levels. Representative models at this level include DeepSEA-style chromatin predictors, SpliceAI for splice site prediction, and Enformer for long-range regulatory modeling. Evaluation here typically involves comparing predicted tracks or binary annotations against experimental measurements.

At the variant level, models take a specific variant (whether SNV, indel, or structural variant) and its surrounding context as input, producing outputs such as pathogenicity scores, predicted molecular impact, or fine-mapping posterior probabilities. Examples include CADD-style deleteriousness scores, AlphaMissense-like variant effect predictors, and Bayesian fine-mapping methods. Evaluation focuses on concordance with clinical annotations, allele frequency patterns, or experimental measurements of variant effects.

At the trait and individual level, models take a person's genotype or sequence along with other features as input and produce risk scores for complex traits, predicted phenotypes, or endophenotypes. Classical polygenic scores and GFM-augmented risk models (Chapter 3; Chapter 18) operate at this level. Evaluation compares predicted risk against observed outcomes in held-out cohorts, often with attention to calibration and discrimination across ancestry groups.

At the clinical and decision level, the inputs are model predictions combined with contextual factors such as guidelines, utility assumptions, and patient preferences. The outputs are actual decisions: whether to treat or not treat, screen or not screen, include a patient in a trial or exclude them. Examples include screening strategies, clinical decision support tools, and trial enrichment protocols. Evaluation at this level requires moving beyond accuracy metrics to consider decision curves, net benefit, and prospective validation.

Good evaluation starts from the intended level of action. If the goal is variant prioritization in a rare disease pipeline, improvement in AUROC on a chromatin benchmark is only indirectly relevant. If the goal is clinical risk stratification, better perplexity on a DNA language model test set is useful only insofar as it leads to more discriminative, better calibrated risk scores. The rest of the chapter climbs this pyramid while keeping a few core metric families in view.

15.2. Metric Families Across Genomic Tasks

Most evaluation in this book falls into four broad metric families, each suited to different types of predictions and scientific questions.

15.2.1. Classification Metrics

For binary or multi-class outputs such as pathogenic versus benign, open versus closed chromatin, or presence versus absence of a histone mark, the standard metrics derive from the confusion matrix. The area under the receiver operating characteristic curve (AUROC or simply AUC) measures the probability that a randomly chosen positive example is ranked above a randomly chosen negative example, providing a threshold-independent summary of discrimination. The area under the precision-recall curve (AUPRC) is more informative when positives are rare, as is typically the case when identifying pathogenic variants among many benign ones or causal variants among many correlated candidates. Simple metrics like accuracy, sensitivity, and specificity are intuitive but sensitive to class imbalance and require choosing specific decision thresholds.

In practice, variant effect predictors and clinical risk models typically report AUROC and AUPRC for prioritization tasks. Regulatory prediction models often report per-task AUROC averaged over hundreds of chromatin assays, sometimes with weighting schemes that emphasize difficult or clinically relevant targets.

15.2.2. Regression and Correlation Metrics

For continuous outputs such as expression levels, log-odds of accessibility, or quantitative traits, the standard metrics measure association between predicted and observed values. Pearson correlation measures linear association, while Spearman correlation measures rank-based association and is robust to monotone transformations of the data. The coefficient of determination (R^2) measures the fraction of variance explained, often computed against a simple baseline such as a mean-only model.

Sequence-to-expression models and multi-omics integrations frequently use correlation between predicted and observed tracks, as in Enformer-style evaluations that compare predicted and measured gene expression across cell types. Polygenic score performance is often reported as incremental R^2, the additional variance explained by genomic features over and above clinical covariates.

15.2.3. Ranking and Prioritization Metrics

Many genomics workflows are fundamentally about ranking rather than absolute prediction. The goal may be to prioritize variants in a locus for follow-up, rank genes or targets for experimental validation, or select individuals at highest risk for screening. While AUROC and AUPRC capture some aspects of ranking quality, additional metrics can be more directly relevant.

Top-k recall or enrichment measures the fraction of true positives captured in the top k predictions, directly addressing questions like “how many real causal variants would land in our top 20 candidates?” Enrichment over baseline measures how much more likely a high-scoring bucket is to contain true positives compared to random expectation. Normalized discounted cumulative gain

(NDCG) emphasizes getting highly relevant items near the top of the ranked list, with diminishing returns for items placed lower. These metrics often align better with practical questions about how predictions will actually be used.

15.2.4. Generative and Language Model Metrics

Self-supervised genomic language models (Chapter 10) introduce their own metrics related to the pretraining objective. Perplexity and cross-entropy on masked-token reconstruction tasks measure how well the model predicts held-out sequence content. Bits-per-base for next-token prediction or compression-style objectives provides a related measure of the model's ability to capture sequence statistics.

These metrics are important for assessing representation quality and for comparing pretraining runs, but they come with important caveats. They are distribution-specific, tied to the particular pretraining corpus and task, which limits comparability across models trained on different data. More importantly, improvements in perplexity do not automatically translate into better variant or trait predictions. A model might achieve excellent perplexity by capturing abundant patterns in the genome, such as repetitive elements and sequence composition, that are largely irrelevant for functional prediction. As a result, generative metrics should always be paired with downstream task metrics to assess real utility.

15.3. Levels of Evaluation: From Base Pairs to Bedside

We now walk through the pyramid from molecular readouts to clinical decisions, focusing on what good evaluation looks like at each level and the common pitfalls that can undermine it.

15.3.1. Molecular and Regulatory-Level Evaluation

At the molecular level, the core tasks include predicting chromatin accessibility, histone marks, and transcription factor binding profiles; predicting splicing outcomes such as percent spliced in (PSI) values or transcription start and termination sites; and predicting readouts from functional assays like massively parallel reporter assays (MPRAs) or CRISPR perturbation screens.

Common evaluation setups involve multi-task classification, where AUROC or AUPRC is computed for each assay and then averaged with or without weighting across assays. Track-wise regression computes Pearson or Spearman correlation between predicted and observed signal profiles across genomic positions. Out-of-cell-type prediction trains on some cell types and tests on others to assess generalization beyond the training distribution.

Several design choices shape the meaning of reported metrics. The granularity of labels matters: base-resolution predictions present a different challenge than predictions averaged over 128-base-pair bins. The size of context windows determines whether the evaluation tests local sequence features or long-range regulatory architecture. The definition of held-out biology, whether new transcription

factors, new cell types, or entirely new genomic loci, determines what kind of generalization is actually being tested.

Common pitfalls include overfitting to specific assays or idiosyncratic lab protocols and inadvertent leakage when nearby genomic regions or replicate experiments are split across train and test sets. A model might appear to generalize to "new" regions while actually leveraging sequence similarity or chromatin context shared with training examples.

15.3.2. Variant-Level Evaluation

At the variant level, tasks include classifying variants as pathogenic versus benign or damaging versus tolerated, predicting functional impact such as effects on splicing, expression, or protein stability, and fine-mapping to assign posterior probabilities of causality to variants in associated loci.

Common benchmarks derive from clinical labels in resources like ClinVar and HGMD, from curated variant sets assembled by diagnostic laboratories, from population-based labels using allele frequency strata in gnomAD-like resources, and from functional assays including saturation mutagenesis, MPRAs, and deep mutational scanning experiments. The choice of benchmark profoundly shapes what the evaluation measures.

Metrics typically include AUROC and AUPRC on binary labels, correlation or rank metrics against experimental effect sizes, and calibration-style metrics for probabilistic outputs. Reliability diagrams for pathogenicity probabilities or fine-mapping posteriors assess whether variants scored at 80% pathogenic are truly pathogenic about 80% of the time.

Several design questions deserve attention. The definition of the negative class matters enormously: common and presumably benign variants, frequency-matched controls, synonymous variants, or synthetic negatives as in CADD (Chapter 4) each create different evaluation contexts with different biases. The choice of what is held out determines the kind of generalization being tested; holding out entire genes, specific loci, or particular variant types tests different capabilities. For fine-mapping and similar tasks where multiple variants per locus compete for causal status, evaluating top-k recall of causal variants per risk locus is often more informative than global AUC across all variants.

This level is also where issues of circularity become especially acute. Scores trained on ClinVar and then evaluated on overlapping variants create feedback loops that inflate apparent performance. We return to this problem in Chapter 16.

15.3.3. Trait- and Individual-Level Evaluation

At the trait and individual level, tasks include predicting quantitative traits such as LDL cholesterol, height, or estimated glomerular filtration rate from genotypes and other features, case-control risk prediction for complex diseases like coronary artery disease or type 2 diabetes, and multi-trait and multi-task risk modeling that jointly predicts related phenotypes.

For quantitative traits, incremental R^2 measures the variance explained by genomic features over and above clinical covariates, directly quantifying what genetics adds to prediction. For binary or time-to-event outcomes, AUROC, AUPRC, and the concordance index (C-index) measure discrimination. Net reclassification improvement (NRI) asks how often individuals are moved across clinically

meaningful risk thresholds in the correct direction, a metric more directly tied to clinical utility than discrimination alone.

Important evaluation settings include within-ancestry versus cross-ancestry performance, building on the portability issues discussed in Chapter 3. Within-cohort versus external validation compares models trained and tested in the same biobank against models validated in entirely separate cohorts with different recruitment, sequencing, and clinical practices. Joint versus marginal contribution of genetics examines how much predictive information comes from genomic features when combined with electronic health records and other multi-omic data (Chapter 14).

Even for purely research models, reporting absolute performance alongside incremental gain over strong baselines is essential for understanding real impact. A polygenic score that achieves 0.65 AUROC for a disease sounds moderately impressive until one learns that clinical variables alone achieve 0.63.

15.3.4. Clinical and Decision-Level Evaluation

Clinical risk models, treatment response predictors, and trial enrichment models (Chapter 18) ultimately need to be evaluated in terms of decisions, not just scores. Beyond discrimination and calibration, several additional concepts become important.

Decision curves and net benefit compare different decision thresholds or policies by weighting true positives versus false positives according to clinical utilities. A model that achieves high AUROC but offers no net benefit at clinically relevant thresholds has not demonstrated clinical value. Cost-sensitive and utility-aware evaluation explicitly models different misclassification costs, recognizing that missing a high-risk patient has different consequences than unnecessary screening. Prospective and interventional evaluation through randomized trials, pragmatic trials, and observational implementations with careful monitoring provides the strongest evidence for clinical utility but is expensive and time-consuming.

This chapter provides only a high-level overview of clinical evaluation; Chapter 18 goes deeper into clinical metrics and deployment considerations, while Chapter 19 discusses evaluation of variant-centric discovery workflows.

15.4. Data Splits, Leakage, and Robustness

Metrics mean little without well-designed data splits. In genomics, the usual approach of randomly assigning 80% of examples to training, 10% to validation, and 10% to testing often fails to test the kind of generalization we actually care about. The structure of genomic data, with its hierarchical organization from bases to variants to individuals to populations, creates many opportunities for subtle information leakage.

15.4.1. Axes of Splitting

Several axes exist along which we can and often should split data. Splitting by individual ensures that genomes from the same person or family do not appear in both training and test sets, preventing models from memorizing individual-specific patterns. Splitting by locus or region holds out contiguous genomic segments such as specific chromosomes or megabase windows, testing whether models can generalize to entirely new genomic contexts. Splitting by gene or target holds out entire genes or protein families for variant effect and protein models, testing whether the model has learned general principles versus gene-specific idiosyncrasies. Splitting by assay, cell type, or tissue trains on some experimental contexts and tests on unseen ones, assessing whether learned regulatory logic transfers across biological conditions. Splitting by ancestry or cohort trains in one population or recruitment setting and evaluates in others, testing whether models generalize across human diversity.

Different scientific questions imply different splitting strategies. The question "Can this model generalize to new loci in the same cell type?" calls for locus or chromosome-based splits. The question "Can it generalize to new cell types?" requires cell-type splits. The question "Can it generalize to different populations or clinical settings?" demands ancestry and cohort splits. Matching the split to the intended use case is essential for meaningful evaluation.

15.4.2. Types of Leakage

Leakage arises when information about the test set sneaks into training, inflating apparent performance without improving real-world generalization. Several forms of leakage are common in genomics.

Duplicate or near-duplicate sequences across splits can occur when overlapping windows around the same variant appear in both training and test sets. Shared individuals or families across train and test can happen when different cohorts containing related individuals are combined without careful deduplication. Benchmark construction leakage occurs when evaluation labels are derived from resources that also guided model design or pretraining, creating circular dependencies. Hyperparameter tuning leakage results from repeatedly evaluating on the test set while choosing checkpoints or model configurations, gradually overfitting to the test distribution.

The practical takeaway is straightforward in principle but demanding in practice: always define the split to match the generalization you care about, then audit carefully for potential linkage and dataset overlap. Chapter 16 focuses on confounders and leakage as sources of biased performance estimates; here, the emphasis is on practical split design.

15.4.3. Robustness and Distribution Shift

Robustness is evaluated by deliberately shifting the data distribution beyond what the model encountered during training. Technical shifts involve new sequencing platforms, different coverage levels, or altered assay protocols. Biological shifts involve new species, tissues, disease subtypes, or ancestry groups not represented in training. Clinical shifts involve new hospitals, different care patterns, or later time periods with evolving patient populations and medical practices.

Robustness evaluations typically involve training on one platform or cohort and testing on another, comparing performance across subgroups such as ancestry-stratified AUROC, and stress-testing models under label noise or missing data. These experiments often reveal that performance on curated, independently and identically distributed benchmarks overestimates usefulness in messy real-world settings, especially for high-stakes clinical decisions.

A model that performs well on curated benchmarks may still struggle in real-world deployment for several reasons. Population diversity issues arise when training corpora underrepresent certain ancestries, leading to biased variant scoring (Chapter 2). Assay heterogeneity means that experimental conditions, laboratories, and technologies in deployment differ from the curated datasets used in training. Phenotypic complexity reflects the reality that many clinically relevant phenotypes involve long causal chains from variant to molecular consequence to tissue-level effect to disease, and models may capture only part of this cascade.

For these reasons, genomic model evaluation increasingly includes cross-population robustness testing, out-of-distribution evaluation on new tissues, cell types, or species, and end-to-end assessments on clinically relevant endpoints often combined with traditional statistical genetics tools.

15.5. Benchmarks, Leaderboards, and Their Limits

Benchmark suites such as those introduced for Nucleotide Transformer and related genomic language models serve important roles in the field. They provide standardized datasets, metrics, and splits that enable apples-to-apples comparisons between architectures. They encourage reproducibility by defining shared baselines against which progress can be measured.

However, benchmark-centric culture has well-documented pitfalls. Overfitting to the benchmark can occur when models are tuned aggressively on a small panel of tasks, achieving impressive headline numbers while degrading on tasks outside the benchmark. Narrow task coverage is common; many existing suites focus on chromatin and transcription factor binding while under-representing splicing, structural variation, or clinical endpoints. Misaligned incentives can emerge when the community prizes fractional improvements in AUROC over more important but harder-to-measure gains in robustness, calibration, or fairness.

Good practice treats benchmark scores as necessary but not sufficient evidence of model quality. They should be complemented with task-specific evaluations that mirror the intended downstream usage. Benchmarks should be periodically refreshed to include new assays, ancestries, and edge cases that stress-test models in new ways. The goal is to use benchmarks as a starting point for evaluation rather than as the final word on model quality.

15.6. Evaluating Foundation Models: Zero-Shot, Probing, and Fine-Tuning

Genomic foundation models (Chapter 12) complicate evaluation because there are multiple ways to use them, each testing different aspects of the learned representations.

15.6.1. Zero-Shot and Few-Shot Evaluation

In zero-shot settings, we apply the pretrained model without any task-specific training. Examples include using masked-token probabilities to rank variants by predicted deleteriousness and using embedding similarities to cluster sequences or annotate motifs. Evaluation in this regime focuses on how well these raw scores correlate with functional or clinical labels and whether few-shot adaptation with small linear heads trained on limited labeled data already yields strong performance.

Zero-shot performance serves as a stress test of representation quality and inductive biases. Strong zero-shot performance suggests that the pretraining objective has captured biologically relevant structure that transfers without explicit supervision. Weak zero-shot performance combined with strong fine-tuned performance suggests that pretraining provides useful initialization but the learned representations are not directly interpretable for the task.

15.6.2. Probing and Linear Evaluation

A common evaluation pattern freezes the foundation model, extracts embeddings for sequences, variants, or loci, and trains simple probes such as linear models or shallow MLPs on downstream labels. This approach isolates the usefulness of learned representations from the model's capacity to adapt during fine-tuning.

Key evaluation questions in the probing regime include how much label efficiency is gained compared to training from scratch, how stable probe results are across random seeds and small dataset variations, and whether probes perform well across diverse tasks or only on those similar to the pretraining objectives. Linear probing provides a clean measure of how much useful information is linearly decodable from model representations.

15.6.3. Full Fine-Tuning and Task-Specific Heads

For high-value tasks, practitioners often fine-tune the foundation model end-to-end, adding task-specific heads for classification, regression, or ranking and adapting to new modalities or clinical contexts. Evaluation then looks similar to classic deep model evaluation but with additional questions specific to the foundation model paradigm.

Transfer versus from-scratch baselines ask whether fine-tuning a foundation model meaningfully outperforms training a comparable architecture from scratch on the same downstream data. Catastrophic forgetting asks whether fine-tuning degrades performance on other tasks, and whether that degradation matters for the intended use. Robustness and fairness ask whether foundation model features inherit or amplify biases present in the pretraining data or introduced during fine-tuning.

Across all evaluation regimes, it is helpful to report absolute performance, the delta compared to strong baselines, and data efficiency curves showing how performance varies with the amount of labeled data. This comprehensive reporting reveals whether pretraining provides genuine benefit or merely matches well-tuned task-specific models.

15.7. Uncertainty, Calibration, and Reliability

Metrics like AUROC summarize ranking quality but say little about how trustworthy individual predictions are. For many applications, especially those involving clinical decisions, we care not only about whether the model is correct on average but also about whether its confidence estimates are meaningful.

Calibration refers to the property that predicted probabilities match observed frequencies. A variant scored at 0.8 probability of being pathogenic should truly be pathogenic about 80% of the time. Well-calibrated models support rational decision-making because the probability scores can be interpreted at face value. Poorly calibrated models, even if they rank examples correctly, provide misleading confidence estimates that can lead to inappropriate decisions.

The distinction between epistemic and aleatoric uncertainty is also important. Epistemic uncertainty arises from limited data and could in principle be reduced by gathering more training examples. Aleatoric uncertainty reflects inherent noise in the problem and cannot be reduced by additional data. Models that can distinguish these uncertainty types provide more actionable predictions, flagging cases where more data might help versus cases where uncertainty is irreducible.

Selective prediction or abstention allows models to say "I don't know" when confidence is low, focusing predictions on cases where the model is reliable. This capability is particularly valuable in clinical settings where the cost of errors is high.

Evaluation tools for uncertainty and calibration include reliability diagrams that plot predicted probabilities against observed frequencies, Brier scores that combine calibration and discrimination in a single metric, and calibration curves stratified by subgroup to identify differential calibration across ancestry, sex, or clinical site. Coverage versus accuracy curves for selective prediction show how accuracy changes as the model restricts predictions to increasingly confident cases: if the model predicts only on the 50% most confident samples, how accurate is it?

For clinical risk models, Chapter 18 covers calibration and uncertainty in more depth. For variant-centric tasks, similar tools apply to pathogenicity probabilities or fine-mapping posteriors, which must be interpreted cautiously in light of confounders discussed in Chapter 16.

15.8. Putting It All Together: An Evaluation Checklist

When designing or reviewing an evaluation for a genomic model, walking through a systematic checklist can help identify gaps and potential problems.

The first question concerns the level of decision. Is the model intended for molecular assay design, variant prioritization, patient risk stratification, or clinical action? The answer should determine which metrics are reported and how they are interpreted. Enrichment metrics make sense for variant ranking; net benefit matters for clinical decisions.

The second question concerns baselines. What are the comparison points? Strong non-deep baselines like logistic regression and classical polygenic scores establish floors that any sophisticated model should exceed. Prior deep models such as DeepSEA, SpliceAI, Enformer, and earlier foundation models establish the relevant state of the art. Reporting both absolute performance and gains over these baselines provides necessary context.

The third question concerns split design. Are individuals, loci, genes, assays, and ancestries appropriately separated between training and test sets? Is there any plausible path for leakage or circularity? These questions require careful auditing of data provenance and split construction.

The fourth question concerns robustness. How does performance vary across cohorts, ancestries, platforms, and time? How does the model behave under label noise or missing data? Robustness evaluations reveal whether benchmark performance translates to real-world utility.

The fifth question concerns uncertainty and calibration. For probabilistic outputs, are calibration and decision-level trade-offs reported? Are subgroup-specific metrics examined to identify differential performance across populations?

The sixth question concerns usage regimes for foundation models. How does the model perform in zero-shot, probing, and fine-tuning settings? Does pretraining help when labeled data are scarce, as measured by data efficiency curves?

The seventh question concerns the story beyond the benchmark. Does improved performance actually change downstream decisions or experimental design? For models intended for clinical deployment, are there plans for prospective or interventional evaluation?

15.9. Looking Forward

This chapter has provided a framework for thinking about evaluation across the full range of genomic models. The subsequent chapters flesh out specific aspects of reliability that evaluation alone cannot address.

Chapter 16 examines confounders, bias, and fairness in detail, showing how evaluation can mislead when data are structured in problematic ways. Population stratification, batch effects, label circularity, and benchmark leakage can all create illusions of performance that evaporate in deployment. Understanding these failure modes is essential for interpreting evaluation results critically.

Chapter 17 focuses on interpretability and mechanisms, turning models from black boxes into sources of testable biological hypotheses. When evaluation shows that a model works, interpretability helps us understand why it works and whether the reasons are biologically meaningful or artifacts of confounded data.

Together, these chapters aim to equip readers with the critical perspective needed to engage with the emerging literature on genomic foundation models. The question is never simply "what is the AUROC?" but rather "what has really been demonstrated, and how much should we trust it?" With careful attention to evaluation design, data splitting, robustness testing, and calibration assessment, we can distinguish models that represent genuine advances from those that merely perform well on convenient benchmarks.

16. Confounders in Model Training

Warning

TODO:

- Add figure: Schematic showing how ancestry structure creates spurious correlations between genotypes and phenotypes
- Add figure: Example PCA/UMAP visualization showing batch clustering vs. label clustering
- Add figure: Illustration of different data splitting strategies (individual-level, locus-level, chromosome-level, time-based)
- Add table: Summary of common confounders with detection methods and mitigation strategies
- Add case study box: Real-world example of ancestry confounding inflating model performance
- Add case study box: Example of benchmark leakage in variant effect prediction
- Consider adding discussion of case-control matching strategies
- Consider adding example of domain adaptation for batch correction

In previous chapters, we treated model performance curves and ROC–AUC numbers as if they transparently reflected how well a model learns biology. In practice, genomic data is riddled with structure that makes it dangerously easy for models, especially large, overparameterized ones, to exploit shortcuts.

Population structure, technical batch effects, benchmark leakage, and label noise can all inflate headline metrics while leaving real-world performance and clinical reliability largely unchanged. These issues are not unique to deep learning; they affect traditional statistics and GWAS as well. But the scale, flexibility, and opacity of modern genomic foundation models (GFMs) make them particularly susceptible.

This chapter surveys the main confounders that arise when training and evaluating genomic models, and outlines practical strategies to detect, mitigate, and transparently report them. Five recurring themes structure the discussion: ancestry stratification and population bias, benchmark leakage and train/test overlap, technical artifacts and batch effects, label noise and ground-truth uncertainty, and cross-ancestry transferability of polygenic scores and other models. Throughout, the key message is simple: architecture advances are only as meaningful as the datasets and evaluation protocols that support them.

16.1. Why Confounders Are Ubiquitous in Genomic ML

A confounder is a variable that influences both the features (such as genotypes or functional readouts) and the labels (such as case/control status or functional effect), creating spurious associations. In genomics, confounders abound for several interconnected reasons.

Data are observational rather than randomized. Disease labels, population sampling, and technical pipelines are all determined by real-world constraints and historical biases rather than experimental design. This stands in contrast to domains where randomized experiments can isolate causal effects.

Population structure is strong and multi-layered. Ancestry, relatedness, and local adaptation affect allele frequencies throughout the genome. These patterns create correlations between genetic variants and both phenotypic outcomes and geographical, environmental, and socioeconomic factors.

Technical pipelines are complex. Each step from sample collection through library preparation, sequencing, alignment, variant calling, and quality control can introduce systematic differences between cohorts. When these differences align with labels, they become exploitable shortcuts.

Labels are noisy. Clinical databases such as ClinVar and high-throughput functional assays contain uncertain and sometimes incorrect annotations. Models trained on noisy labels may learn to predict the noise rather than the underlying biology.

Deep models are powerful pattern detectors. If confounders produce consistent patterns that correlate with labels, models will happily learn those shortcuts instead of the causal biology we care about. The result is impressive performance on held-out data that share the same hidden structure, but brittle behavior as soon as we change ancestry, institution, assay, or time period.

16.2. Ancestry Stratification and Population Bias

16.2.1. How Ancestry Becomes a Shortcut

Human genetic variation is structured by ancestry: allele frequencies and haplotype patterns differ across populations due to demographic history, drift, and selection. Disease prevalence, environmental exposures, and health-care access are also ancestry- and region-dependent.

This creates a classic confounding scenario. Features, in the form of genotypes or sequence variants, reflect ancestry. Labels, whether case/control status, disease subtype, or even pathogenic versus benign annotations, can vary with ancestry. If a case cohort is primarily of one ancestry and controls are primarily of another, a model can achieve high predictive performance by acting as an ancestry classifier rather than a disease predictor. The same issue arises for variant effect prediction: variants common in one ancestry but rare in another can be spuriously tagged as pathogenic or benign based on how databases were curated rather than on their biological effects.

16.2.2. Manifestations in Genomic Models

Ancestry confounding manifests in several common patterns. Case/control imbalance across ancestries occurs when cases over-represent individuals of one ancestry while controls over-represent another. Reference database bias arises when variant annotations derive mostly from European-ancestry cohorts, making "benign" often synonymous with "common in Europeans." Implicit ancestry markers allow high-capacity models to recover ancestry even when explicit labels are removed, through cryptic relatedness, shared haplotypes, and local LD patterns that differentiate populations.

For transformer-based GFMs trained on large genomic corpora, even subtle ancestry differences are enough to support a shortcut. These models can pick up on patterns invisible to simpler methods, which is both their strength and, in the presence of confounders, their vulnerability.

16.2.3. Detecting Ancestry Confounding

Several practical diagnostics can reveal ancestry confounding. PCA or UMAP visualization of genotypes or embeddings allows inspection of whether cases and controls cluster by ancestry. If they do, that is a red flag requiring further investigation. Stratified performance evaluation examines metrics separately within each ancestry group; large performance drops or reversals across groups suggest the model relies on cross-ancestry differences rather than disease biology. Ancestry-only baselines fit a simple classifier on ancestry principal components or self-identified ancestry alone. If this baseline approaches the full model's performance, the model is likely exploiting similar information. Permutation tests within ancestry strata shuffle labels within ancestry groups. This should destroy performance for a truly disease-specific signal, but not for models relying on cross-ancestry differences.

16.2.4. Mitigating Ancestry Bias

Mitigation is imperfect, but several strategies help reduce the impact of ancestry confounding. Balanced study design recruits cases and controls with similar ancestry distributions wherever possible, or matches controls to cases on ancestry. Within-ancestry evaluation reports metrics for each ancestry separately and uses training–validation splits that preserve within-group structure. Covariate adjustment includes ancestry PCs, kinship matrices, or mixed-model random effects in simpler models; for deep models, conditioning on or adversarially removing ancestry signals from learned embeddings can help. Multi-ancestry training trains on diverse populations rather than restricting to a single ancestry, and explicitly models ancestry as a domain variable. Fairness-aware objectives introduce regularizers or constraints that penalize performance disparities across ancestry groups, which becomes especially important in clinical deployment contexts.

The broader implications of ancestry for model development and clinical deployment are discussed in Chapter 18 and Chapter 3. Here, the key point is that ancestry confounding can inflate apparent model performance while undermining the very generalization we care about.

16.3. Benchmark Leakage and Train/Test Overlap

Even with perfectly balanced ancestries, evaluation can be misleading if information leaks from training to test sets. Leakage is especially insidious in genomics because the genome is highly structured and redundant, public datasets and benchmarks are heavily reused, and many papers do not fully specify how splits were constructed.

16.3.1. Forms of Leakage

Common leakage patterns include individual overlap, where the same person or close relative appears in both train and test sets, directly or via related cohorts. Variant overlap occurs when exact variants, or near-identical ones at the same locus, appear in both splits, which happens when different datasets are merged without careful deduplication. Locus-level overlap splits variants in the same gene, regulatory element, or LD block between train and test. A model may learn locus-specific idiosyncrasies instead of general rules, achieving strong apparent performance without learning transferable patterns. Database reuse leakage occurs when benchmarks constructed from ClinVar, gnomAD, or other public databases overlap with external sets used for evaluation, whether through direct inclusion or through shared curation of the same underlying variants. Time-based leakage trains models on data that include later submissions of the same variants or patients that are used as "future" test examples.

For large models, even very small overlaps can inflate metrics, particularly when test sets are small. A model that memorizes a few hundred variants in a test set of a few thousand can achieve substantial apparent performance gains without learning anything general.

16.3.2. Safer Splitting Strategies

To reduce leakage, individual-level splits ensure that no individual, or closely related individuals if kinship is known, appears in both train and test sets. Locus- or gene-level splits hold out entire genes, enhancers, or genomic regions for variant effect prediction, so that test loci are truly unseen. Chromosome-based splits hold out entire chromosomes or chromosome arms for genome-wide tasks. This is not perfect, since genes on different chromosomes may share regulatory logic, but it greatly reduces local dependency leakage. Time-based splits train on data up to a cutoff date and test on later data, mimicking realistic deployment where models must predict on data that did not exist during training. Transparent data provenance tracks the origin of each sample and variant, including database version and submission ID, to avoid accidental reuse.

16.3.3. Evaluation Design and Reporting

Beyond the split itself, evaluation design matters. Reporting both in-distribution performance on the same cohort and out-of-distribution performance on new cohorts, ancestries, or technical pipelines reveals how much of apparent performance reflects genuine generalization. Cross-cohort benchmarks that train on one cohort and test on another with different recruitment or sequencing characteristics provide stronger evidence of robustness. Sharing code and detailed recipes for dataset construction allows others to reproduce and critique splitting choices.

16.4. Technical Artifacts: Batch Effects and Platform Differences

While ancestry and population structure reflect biological reality, batch effects are artifacts of the measurement process. Differences in sample collection protocols, library preparation kits, sequencing platforms and chemistry versions, read length, depth, and coverage, and alignment and variant calling pipelines can all introduce systematic shifts in feature distributions.

16.4.1. How Batch Effects Confound Models

Technical batches often correlate with labels. A case cohort may be sequenced at one institution on one platform, while controls are sequenced elsewhere with different protocols. A longitudinal study might switch from one capture kit or sequencer to another halfway through, coinciding with changes in enrollment criteria. Public datasets may aggregate studies with very different technical characteristics.

In such settings, a model can achieve high accuracy by recognizing batch signatures, such as patterns of missingness, depth, or noise spectra, rather than bona fide biological signals. The model learns to distinguish batches, not biology, and evaluation within the same batch structure produces misleadingly optimistic results.

16.4.2. Diagnosing Technical Confounders

Common diagnostics include embedding visualization by batch, which projects learned embeddings or expression/coverage profiles via PCA or UMAP, then colors points by batch, platform, or institution. Strong clustering by these variables suggests technical structure that the model might exploit. Batch-only baselines train a classifier using only batch labels or simple technical covariates such as read depth or platform indicators. High baseline performance is a warning sign that batch information predicts labels. Negative controls evaluate models on samples where labels should be uncorrelated with batch, such as technical replicates or randomized subsets. Replicate consistency examines whether predictions are consistent across technical replicates processed in different batches.

16.4.3. Mitigating Batch Effects

Mitigation is an active research area, with common approaches including careful study design that randomizes cases and controls across batches whenever possible, avoiding systematic alignment between batch and outcome. Preprocessing harmonization uses standardized pipelines for alignment and variant calling, reprocessing raw data when feasible to reduce inter-study differences. Statistical batch correction methods such as ComBat, Harmony, and related approaches can reduce batch effects in expression or chromatin data; similar ideas can be applied to embeddings from GFMs. Domain adaptation and adversarial training train representations that are predictive of labels while being invariant to batch or platform, using techniques like gradient reversal layers or distribution matching objectives. Explicit multi-domain modeling treats each batch or platform as a domain and learns domain-conditional parameters or mixture-of-experts models.

Even with aggressive correction, residual batch structure typically remains. Transparent reporting and robustness checks are essential for understanding how much residual confounding might remain.

16.5. Label Noise and Ground-Truth Uncertainty

Large-scale genomic models rely on labels from clinical variant interpretation databases, GWAS-derived case/control status, high-throughput functional screens such as MPRA, saturation mutagenesis, and CRISPR screens, and curated gold-standard sets for variant effect prediction, splicing predictions, or PGS. These labels are not error-free.

16.5.1. Sources of Label Noise

Conflicting annotations arise because ClinVar often contains variants with conflicting interpretations or uncertain significance, and criteria for pathogenicity change over time as knowledge advances. A variant classified as pathogenic five years ago may be reclassified as benign today, or vice versa. Ascertainment bias means that variants labeled as benign may simply be common in some populations, while variants labeled as pathogenic may be enriched in clinically ascertained cohorts that over-represent certain ancestries or disease presentations. Measurement noise in functional assays reflects the variable reproducibility of high-throughput experiments across labs, conditions, and replicates. Thresholding continuous scores into discrete classes compounds the issue by introducing arbitrary boundaries. Phenotyping noise arises because clinical case/control labels may be inaccurate due to misdiagnosis, incomplete records, or heterogeneous disease definitions across recruitment sites.

16.5.2. Consequences for Models

Label noise can limit achievable performance, especially for tasks with overlapping phenotype definitions. It can encourage models to learn spurious proxies that correlate with annotation errors rather than biology. It can bias calibration and decision thresholds, particularly in imbalanced settings where a small fraction of mislabeled examples in the minority class has disproportionate impact.

In some scenarios, training on noisy labels still improves performance if noise is roughly symmetric or if the dataset is very large. However, for rare disease variants and high-stakes predictions, even small fractions of mislabeled examples can be problematic.

16.5.3. Strategies for Robust Learning with Noisy Labels

Several approaches address label noise. Curated subsets restrict training and evaluation to high-confidence annotations, such as ClinVar pathogenic and benign classifications with multiple submitters and no conflicts, even at the cost of reduced size. Soft labels and uncertainty modeling use probabilistic labels derived from inter-rater disagreement, confidence scores, or continuous assay measurements rather than hard binary labels. Robust losses employ loss functions less sensitive to mislabeled points, such as label smoothing, margin-based losses, or methods that down-weight high-loss outliers. Noise-aware training explicitly models label noise, for example via a noise transition matrix or latent variable models, and jointly infers true labels alongside model parameters. Consensus across modalities combines evidence from protein structure, evolutionary conservation, regulatory context, and clinical data, treating disagreements as signals of uncertainty.

Mechanistic interpretability can also help flag model predictions that disagree with known biology, potentially identifying mislabeled examples in training data.

16.6. Cross-Ancestry PGS Transferability and Model Fairness

Polygenic scores and other genome-wide predictors have gained traction as potential tools for early disease risk stratification. However, many PGS have been developed primarily in individuals of European ancestry, raising concerns about reduced predictive accuracy in underrepresented ancestries, biased calibration where risk is systematically over- or under-estimated in certain groups, and downstream disparities if PGS-informed clinical decisions are applied uniformly.

16.6.1. Why Transferability Fails

Reasons for poor cross-ancestry transfer include allele frequency differences, where effect estimates calibrated in one population may not generalize when allele frequencies change. LD pattern differences mean that tagging SNPs used in PGS may capture causal variants in one ancestry but not another. Gene–environment interaction occurs because environmental exposures and lifestyle factors that interact with genetic risk differ across populations. Ascertainment and recruitment biases arise because early GWAS datasets often oversampled certain ancestries, clinical populations, or socioeconomic strata.

These issues carry over to deep learning-based PGS and GFMs fine-tuned for disease prediction. Even if the underlying model is trained on diverse genomes in a self-supervised fashion, the supervised fine-tuning and evaluation data can reintroduce bias.

16.6.2. Towards More Equitable Models

Approaches to improve cross-ancestry performance and fairness include multi-ancestry GWAS and training data that include diverse cohorts at the design stage rather than as an afterthought. Ancestry-aware modeling conditions effect sizes or model parameters on ancestry, or learns ancestry-invariant representations coupled with ancestry-specific calibration. Transfer learning and fine-tuning adapt models from ancestries with large datasets to those with smaller datasets using domain adaptation techniques. Fairness metrics report group-wise calibration, sensitivity, specificity, and decision-curve analyses rather than just overall AUC. Stakeholder engagement works with clinicians, ethicists, and affected communities to decide when and how PGS should be used, and what constitutes acceptable performance gaps.

16.7. From Cautionary Tales to Best Practices

Modern genomic foundation models promise impressive capabilities: genome-scale variant effect prediction, cross-species transfer, multi-omics integration, and clinically actionable risk scores. Yet without rigorous attention to confounders, these capabilities can be overstated or misapplied.

Emerging work on genomic evaluation frameworks emphasizes several principles. Data documentation provides detailed datasheets for datasets and benchmarks, including recruitment, ancestry

composition, technical pipelines, and label provenance. Robust evaluation protocols include cross-cohort, cross-ancestry, and time-split evaluations that stress-test models beyond their training distribution. Confounder-aware training explicitly models ancestry, batch, and label uncertainty, and uses adversarial or domain-adaptation techniques. Transparent reporting clearly communicates limitations, potential failure modes, and groups for whom the model has not been validated.

16.8. A Practical Checklist for Confounder-Resilient Genomic Modeling

To close, here is a concise checklist applicable when designing, training, and evaluating genomic models.

For population structure, quantify ancestry and relatedness via PCs or kinship. Ensure cases and controls are balanced within ancestry groups. Report performance stratified by ancestry.

For data splits and leakage, confine individuals, families, and closely related samples to a single split. Split at the locus, gene, or chromosome level where appropriate. Check for overlap with external databases used in evaluation.

For batch and platform effects, assess whether technical variables such as batch, platform, or institution are correlated with labels. Visualize embeddings colored by batch. Use harmonization, batch correction, or domain adaptation as needed.

For label quality, understand how labels are defined and quantify their uncertainty. Filter to high-confidence subsets for primary evaluation. Employ robust training strategies to handle label noise.

For cross-group performance and fairness, report metrics for each ancestry and relevant subgroup. Assess whether risk scores are calibrated across groups, or whether group-specific calibration is required. Consider the ethical and clinical implications of residual performance gaps.

For reproducibility and transparency, fully document dataset construction and splitting procedures and make them shareable. Ensure code and evaluation pipelines are available for independent verification.

By systematically addressing these points, we can ensure that the gains from modern architectures, whether transformers, SSMs, or GFMs, translate into trustworthy advances in genomic science and medicine, rather than brittle models that merely reflect quirks of our data and history.

17. Interpretability & Mechanisms

> ⚠ Warning
>
> **TODO:**
>
> - Add figure: Attribution method comparison showing ISM, DeepLIFT, and integrated gradients on the same regulatory sequence with a known CTCF motif
> - Add figure: TF-MoDISco pipeline schematic showing seqlet extraction → clustering → motif derivation → grammar inference
> - Add figure: Attention pattern visualization from a genomic language model showing operon-like structure or enhancer-promoter linkages
> - Add figure: Sei sequence class UMAP colored by regulatory program (promoter, enhancer, repressive, etc.) with example variants mapped
> - Add figure: Faithfulness vs plausibility illustration showing a motif that "looks biological" but fails counterfactual deletion tests
> - Add table: Comparison of attribution methods with columns for computational cost, reference dependency, noise characteristics, and typical use cases
> - Consider adding BPNet case study as concrete example of motif discovery workflow
> - Add code snippet or pseudocode for ISM calculation

17.1. Why Interpretability Matters for Genomic Models

Deep learning models in genomics increasingly operate as systems-level surrogates for biology. They predict chromatin features, gene expression, and variant effects directly from sequence, achieving accuracy that would have seemed implausible a decade ago. When such models drive mechanistic hypotheses or inform clinical decisions, understanding how they make predictions becomes as important as understanding how well they perform.

Interpretability in this context serves several distinct but interconnected roles. The most scientifically compelling is mechanistic insight: extracting sequence motifs, regulatory grammars, and long-range interaction patterns directly from trained models. A well-designed interpretability analysis can turn a black-box predictor into a source of candidate mechanisms that can be tested experimentally. When a model trained to predict chromatin accessibility learns filters that match known transcription factor binding motifs, this validates that the model has discovered biologically meaningful patterns. When the same analysis reveals novel motif variants or unexpected spacing constraints, it generates hypotheses that extend beyond what was known before training.

Interpretability also serves as a tool for model debugging and confounder detection. Deep networks can achieve high benchmark accuracy by learning spurious correlations rather than genuine regulatory signals. A model might learn that certain k-mers correlate with peak calls because of batch effects

in the training data, or that GC content predicts chromatin accessibility because GC-rich regions tend to be more mappable and thus better covered by sequencing. Interpretability methods can reveal such shortcuts by showing what features the model actually relies upon. This diagnostic function complements the data-level confounder analyses discussed in Chapter 16 by interrogating model internals directly.

In clinical and translational settings, interpretability supports variant interpretation workflows by explaining why specific rare or de novo variants are predicted to be damaging. A pathogenicity score alone may be insufficient for clinical decision-making; knowing that a variant disrupts a specific transcription factor binding motif in a disease-relevant enhancer provides interpretable evidence that can be combined with family history, functional assays, and literature review. Interpretability tools that produce such explanations bridge the gap between computational predictions and actionable clinical reasoning.

Finally, interpretability enables scientific communication by condensing high-dimensional latent representations into human-readable abstractions. Motifs, regulatory sequence classes, and interaction graphs can be shared across laboratories and applications in ways that raw model weights cannot. A published motif vocabulary derived from a foundation model becomes a reusable resource for the community, even if the original model is computationally expensive to run or subject to access restrictions.

This chapter surveys the main interpretability tools developed for genomic models, from convolutional filter analysis and saliency maps to global regulatory vocabularies and attention patterns in genomic language models. Throughout, the emphasis is on mechanistic interpretability: moving from correlational explanations ("what features correlate with the prediction?") to causal hypotheses ("what regulatory mechanism does the model imply?").

17.2. Interpreting Convolutional Filters as Motifs

Convolutional neural networks remain a workhorse for modeling cis-regulatory sequence, as described in Chapters 5 through 7. In many of these models, first-layer convolutional filters act as motif detectors. A filter slides along the one-hot encoded sequence, computing a dot product between its learned weights and the local sequence window at each position. High activation indicates that the subsequence closely matches the filter's preferred pattern.

17.2.1. From Filters to Motif Logos

Converting learned filters into interpretable motifs follows a standard workflow. The trained model is run on a large sequence set, typically the training data or genome-wide tiles, and for each filter the positions where its activation exceeds a threshold are recorded. The fixed-length windows around these high-activation positions are then extracted and aligned, and base frequencies at each position are computed to build a position weight matrix (PWM). This PWM can be visualized as a sequence logo, where letter heights reflect information content, and compared to known motif databases like JASPAR or HOCOMOCO using similarity scores. Filters that produce PWMs resembling characterized transcription factors can be annotated with candidate TF identities.

This procedure has been applied extensively to models like DeepSEA and its successors, demonstrating that early convolutional layers learn motifs for canonical transcription factors and chromatin-associated patterns. Such validation confirms that models are discovering biologically meaningful sequence features rather than arbitrary patterns that happen to correlate with training labels.

17.2.2. Beyond First-Layer Filters

Deeper convolutional layers aggregate lower-level motifs into more complex representations. These layers can encode combinatorial motifs that respond to pairs or clusters of transcription factor binding sites, grammar patterns involving distance or orientation constraints, and contextual preferences that depend on surrounding sequence composition like GC content or nucleosome positioning signals. However, directly interpreting deeper layers becomes increasingly difficult because receptive fields expand and nonlinearities accumulate. The activation of a deep-layer filter depends on intricate combinations of early-layer patterns, making it hard to summarize what the filter "means" in simple biological terms. This interpretive challenge motivates attribution-based approaches that trace predictions back to individual input bases rather than trying to interpret intermediate representations.

17.3. Attribution Methods: Connecting Bases to Predictions

Attribution methods assign an importance score to each input base, reflecting how much that position contributes to a prediction for a specific task and sequence. If a model $f(x)$ predicts some output from sequence x, attribution methods estimate the contribution of each base x_i to $f(x)$, typically for a specific output neuron such as chromatin accessibility in a particular cell type. The resulting attribution maps can reveal which sequence positions drive a prediction, highlighting candidate motifs and regulatory elements.

17.3.1. In Silico Mutagenesis

In silico mutagenesis (ISM) is conceptually the most straightforward attribution method and works with any model, regardless of architecture. For each position i and alternative base b, ISM creates a mutated sequence $x^{(i \to b)}$ and computes the change in prediction: $\Delta f_{i,b} = f(x^{(i \to b)}) - f(x)$. These changes can be aggregated across non-reference alleles to obtain a per-base importance score, typically by taking the maximum or mean absolute change.

ISM provides true counterfactual information about how the model responds to sequence perturbations. Unlike gradient-based methods that estimate local sensitivity, ISM directly measures what happens when a base is changed. This makes ISM the gold standard for faithfulness: if ISM shows that mutating a position changes the prediction, that is a direct observation rather than an approximation.

The primary limitation of ISM is computational cost. Scoring all possible single-nucleotide substitutions requires $L \times 3$ forward passes for a sequence of length L, which becomes expensive for long sequences or large models. Variants of ISM can reduce this cost by focusing on specific regions of interest or by using saturation mutagenesis only in targeted windows. For variant effect

prediction specifically, ISM reduces to computing the difference between reference and alternative allele predictions, which requires only two forward passes per variant.

17.3.2. Gradient-Based Methods

Gradient-based methods approximate how much the prediction would change if each input base were perturbed, using backpropagation rather than explicit perturbation. The simplest approach computes the gradient of the output with respect to the input: $s_i = \partial f(x)/\partial x_i$. With one-hot encoding, this gradient can be interpreted as the sensitivity to changing the nucleotide at position i. A common variant multiplies the gradient by the input to focus on positions where the current nucleotide (rather than hypothetical alternatives) is important.

Vanilla gradients require only a single backward pass per sequence, making them computationally efficient. However, they are susceptible to gradient saturation, where gradients vanish in regions where the model is already confident. Saturated regions may be functionally important but show near-zero gradients because small perturbations do not change the prediction.

DeepLIFT (Deep Learning Important FeaTures) addresses saturation by comparing neuron activations between an input and a reference sequence, distributing differences back to inputs using layer-wise rules rather than raw gradients. This approach avoids gradient saturation and enforces a consistency constraint: the sum of input contributions matches the difference in output between input and reference. DeepLIFT has been widely used for genomic models, particularly in conjunction with TF-MoDISco, where its base-level importance scores serve as inputs for motif discovery.

Integrated gradients (IG) compute the path integral of gradients along a linear interpolation from a reference sequence x' to the input x:

$$\mathrm{IG}_i(x) = (x_i - x_i') \int_{\alpha=0}^{1} \frac{\partial f(x' + \alpha(x - x'))}{\partial x_i} d\alpha.$$

This integral is approximated via a Riemann sum over discrete interpolation steps. Integrated gradients satisfy desirable theoretical properties including sensitivity (if changing an input changes the output, that input receives nonzero attribution) and implementation invariance (functionally equivalent networks produce identical attributions). In practice, IG tends to be less noisy than raw gradients.

All gradient-based methods require choosing a reference sequence, which significantly affects the resulting attributions. Common choices include random genomic sequence, dinucleotide-shuffled versions of the input that preserve local composition, or an average "non-functional" sequence. Different references emphasize different aspects of the signal. A shuffled reference highlights features that differ from random sequence with matched composition, while a zero reference (all bases equally weighted) treats any informative position as important.

17.4. From Attributions to Motifs: TF-MoDISco

Attribution maps highlight where the model focuses, but they do not automatically yield consistent motifs or regulatory grammars. A DeepLIFT attribution track might show high importance at scattered positions throughout a sequence without revealing that those positions collectively form

instances of the same transcription factor binding site. TF-MoDISco (Transcription Factor Motif Discovery from Importance Scores) was developed to bridge this gap by discovering motifs from attribution scores rather than from raw sequences.

The core insight of TF-MoDISco is that operating on importance-weighted sequences rather than raw sequences focuses motif discovery on positions the model actually uses. Traditional motif discovery algorithms applied to regulatory sequences must contend with the fact that most positions are not part of functional motifs. By extracting "seqlets" (short windows where total importance exceeds a threshold) and clustering them based on both sequence and importance profiles, TF-MoDISco identifies the specific patterns that drive model predictions.

The workflow begins by computing importance scores for many sequences using DeepLIFT, ISM, or integrated gradients. Local windows where total importance exceeds a threshold are extracted as seqlets, each representing a candidate motif instance. These seqlets are then compared using similarity metrics that consider both sequence content and the importance score profile, and clustered into groups corresponding to putative motifs. Within each cluster, seqlets are aligned and consolidated into position weight matrices and importance-weighted logos. The resulting motifs can be matched to known transcription factor binding sites or flagged as novel patterns.

Beyond individual motifs, TF-MoDISco enables grammar inference by analyzing how motifs co-occur within sequences. Mapping motif instances back onto the genome reveals patterns of co-occurrence, characteristic spacing between motif pairs, and orientation preferences. These grammatical rules can be validated through in silico experiments: inserting or removing motifs in synthetic sequences and observing whether predictions change as expected.

When applied to models like BPNet trained on ChIP-seq data, TF-MoDISco has recovered known transcription factor motifs, discovered novel sequence variants, and revealed grammars such as directional spacing constraints that have been validated with synthetic reporter assays. In the context of genomic foundation models, an analogous workflow applies: use the model to produce base-level attributions for a downstream task, run TF-MoDISco to extract a task-specific motif vocabulary, and analyze how motif usage varies across cell types, conditions, or species.

17.5. Interpreting Attention and Long-Range Context

Transformer-based models use self-attention to mix information across long genomic contexts, enabling them to capture distal regulatory interactions and genomic organization that are invisible to models with narrow receptive fields. Interpretability for these models often centers on attention patterns and long-range attribution, asking which distant positions influence predictions at a given location.

17.5.1. Attention in Genomic Language Models

Genomic language models (gLMs) trained on prokaryotic genomes treat genes or genomic tokens as elements of a sequence and learn to predict masked tokens, analogous to protein or text language models. Work on gLMs trained on millions of metagenomic scaffolds has shown that these models learn non-trivial genomic structure that can be read out from attention patterns.

Certain attention heads specialize in connecting genes that are part of the same operon or functional module. When attention weights are visualized as edges between gene positions, they reveal networks of co-regulated genes that often align with known operon boundaries. Other heads capture functional semantics, with attention patterns that cluster genes by enzymatic function or gene ontology category. Still others encode taxonomic signals, separating clades and capturing clade-specific gene neighborhood patterns.

These findings suggest that the model has inferred a "syntax" of gene neighborhoods: which genes tend to co-occur, in what order, and conditioned on phylogenetic context. While attention weights are not universally faithful explanations of model decisions (high attention need not correspond to large causal influence), attention analysis in genomic language models reveals emergent mechanistic structure that is consistent with known biological organization.

17.5.2. Distal Regulatory Elements in Enformer-Like Models

Enformer and related models predict chromatin features and gene expression from large genomic windows spanning 100 kb or more, combining convolutional layers for local feature extraction with transformer blocks for long-range integration. A central interpretability question for these models is which distal enhancers drive predicted expression at a given transcription start site, and how variants in distal elements propagate to gene-level outputs.

Gradient-based attributions can be computed over the entire input window, producing importance tracks that span tens to hundreds of kilobases. Visualizing these tracks alongside gene annotations reveals putative enhancers and silencers: positions where the model places high importance for predicting expression. Attention pattern analysis complements gradient methods by identifying attention heads that consistently link distal positions to transcription start site regions. These high-attention edges can be compared to chromatin conformation data from Hi-C experiments to assess whether the model has learned biologically plausible enhancer-promoter interactions.

In silico perturbation experiments provide additional validation. Candidate enhancers identified by attribution or attention can be deleted or scrambled in the input sequence, and the resulting change in predicted expression quantifies how much that element contributes to the model's output. Inserting synthetic motifs or strengthening existing motif scores can test dose-response relationships, asking whether enhancing a putative regulatory element produces the expected increase in predicted expression.

Together, these analyses can reveal candidate enhancer-promoter links and the transcription factor motifs that the model deems critical for gene regulation. They help translate raw attention weights and attribution scores into mechanistic hypotheses that can be tested experimentally.

17.6. Global Regulatory Vocabularies: Sei Sequence Classes

Most motif-based interpretation operates at the local level, asking which motifs appear in a particular sequence and how they contribute to a specific prediction. Sei takes a complementary global approach by learning a vocabulary of regulatory sequence classes that summarize the vast diversity of chromatin profiles across the genome.

17.6.1. The Sei Framework

Sei trains a deep sequence model to predict tens of thousands of chromatin profiles covering transcription factor binding, histone modifications, and chromatin accessibility across many cell types. The key interpretability step is to compress these thousands of outputs into a few dozen sequence classes, each representing a characteristic regulatory activity pattern.

Sequence classes are derived by clustering genome-wide predictions. For each of millions of genomic positions, Sei computes predicted chromatin profiles and projects them into a lower-dimensional space using principal component analysis. These projections are then clustered to identify recurrent patterns of regulatory activity. The resulting classes include promoter-like patterns enriched for H3K4me3 and TSS proximity, enhancer-like patterns with H3K27ac and H3K4me1, repressive patterns dominated by H3K27me3 or H3K9me3, and cell-type-specific modules corresponding to neuronal, immune, or other lineage-specific regulatory programs.

Each input sequence or variant can be scored against all sequence classes, effectively mapping it to a point in a low-dimensional regulatory activity space. This representation has several interpretability advantages. Instead of reasoning about thousands of raw chromatin predictions, one can describe a sequence in terms of human-interpretable categories: "this variant increases neuronal enhancer activity while decreasing polycomb repressive marks." Variants can be summarized by their shifts in sequence-class scores, yielding concise functional descriptions. GWAS loci can be enriched for specific sequence classes, revealing which tissues and regulatory programs are most relevant to a disease.

This notion of a regulatory vocabulary parallels word embeddings or topic models in natural language processing. It provides a bridge between highly multivariate model outputs and mechanistically interpretable axes of variation that can be communicated across studies and applications.

17.7. A Case Study: From Base-Pair Attributions to Regulatory Grammar

Putting the pieces together, a typical mechanistic interpretability pipeline for a CNN or transformer-based regulatory model proceeds through several connected stages.

The starting point is a trained predictive model, for example one that predicts chromatin accessibility or transcription factor ChIP-seq tracks from sequence. For sequences where the model makes confident predictions in a target cell type, base-level attributions are computed using DeepLIFT or integrated gradients. These attributions are fed into TF-MoDISco, which extracts seqlets from high-attribution regions, clusters them, and derives motifs. The resulting motifs are matched to known transcription factors where possible, and novel motifs are flagged for further investigation.

Grammar inference follows from analyzing motif instances across the full set of high-confidence predictions. Motif co-occurrence patterns reveal which factors tend to operate together. Spacing distributions between motif pairs identify characteristic distances that may reflect cooperative binding or nucleosome constraints. Orientation analysis determines whether certain motif pairs require specific relative orientations to function. In silico knock-in and knock-out experiments confirm these grammatical dependencies: if the model predicts that two motifs must co-occur for

high accessibility, deleting either motif from a sequence should reduce the prediction, while inserting both into a neutral background should increase it.

The local motif grammar can then be connected to global regulatory context. Motif-rich regions can be mapped to Sei sequence classes to understand what broader regulatory programs they participate in. For transformer-based models, attention patterns or long-range attributions can link local motif clusters to distal elements, revealing enhancer-promoter architectures or chromatin domain boundaries.

Validation closes the loop by connecting model-derived hypotheses to external evidence. Do motif disruptions align with reporter assay effects or allelic imbalance measured in functional genomics experiments? Do inferred enhancer-promoter links correspond to contacts observed in Hi-C or to effects measured in CRISPR perturbation screens? This integrated approach moves beyond descriptive saliency maps toward testable hypotheses about regulatory logic.

17.8. Evaluating Interpretations: Faithfulness versus Plausibility

Not all explanations are equally trustworthy. Effective interpretability work must grapple with the distinction between plausibility (does the explanation "look" biological?) and faithfulness (does the explanation accurately reflect the internal computation of the model?).

An explanation is plausible if it matches prior biological knowledge. Discovering a motif that resembles CTCF is plausible because CTCF is a well-characterized chromatin organizer. Plausibility provides reassuring sanity checks but does not guarantee that the model actually uses the plausible feature. An explanation is faithful if perturbing the identified features changes the model's output as predicted. If removing a putative CTCF site from a sequence causes the model's chromatin accessibility prediction to drop, the explanation has some degree of faithfulness.

Several pitfalls complicate the relationship between plausibility and faithfulness. Attention weights in transformer models need not correspond to large changes in output; high attention may reflect information routing rather than causal influence on predictions. A model might attend strongly to certain positions for bookkeeping purposes without those positions driving the final output. Combining attention analysis with attribution or perturbation experiments yields more reliable insights by checking whether high-attention positions are also high-importance positions under counterfactual tests.

Gradient-based attribution methods can produce noisy maps or miss important features in saturated regions where gradients are near zero. Comparing multiple methods (ISM, DeepLIFT, integrated gradients) and checking for consistency helps identify robust signals. If different methods agree that a position is important, confidence increases; if they disagree, the discrepancy warrants investigation.

Perhaps most problematically, models may learn shortcut features that produce clean, plausible-looking motifs but are not mechanistically meaningful. A model might learn that certain k-mers correlate with peak calls because of barcode sequences in the training data, or that GC content predicts accessibility because of mappability biases. These shortcuts can produce interpretable patterns that are biologically vacuous.

Recommended practices for validating interpretations include sanity checks where model weights are randomized (attributions should degrade to noise) or training labels are scrambled (derived

motifs should disappear or lose predictive power). Counterfactual tests delete or scramble high-attribution regions to confirm that predictions drop accordingly, or insert discovered motifs into neutral backgrounds to test gain-of-function effects. Benchmarking on synthetic datasets with known ground-truth grammar provides controlled settings where the ability of interpretability methods to recover planted motifs and interactions can be quantified.

17.9. A Practical Interpretability Toolbox

For practitioners working with genomic foundation models and their fine-tuned derivatives, several interpretability strategies form a practical toolbox.

Local effect estimation focuses on individual variants or short sequence windows. For variant effect prediction, comparing reference and alternative allele scores provides direct effect estimates, while small-window ISM around variants reveals which nearby positions modulate the effect. Per-base attributions can be aggregated into per-variant or per-motif scores for summary statistics.

Motif and grammar discovery begins with computing base-level attributions for sequences where the model makes high-confidence predictions. Running TF-MoDISco or similar algorithms builds a motif vocabulary that can be compared across tasks, cell types, or training conditions. Grammar analysis examines motif co-occurrence, spacing, and orientation to infer combinatorial rules.

Global context visualization applies to transformer-based models, where attention patterns can reveal which distant positions the model considers when making predictions at a given location. For hybrid architectures like Enformer, combining long-range attributions with contact maps helps hypothesize regulatory architectures that span tens to hundreds of kilobases.

Regulatory vocabularies and embeddings use frameworks like Sei to project sequences into interpretable regulatory activity spaces. Clustering variants, enhancers, or genomic regions by their sequence-class profiles reveals shared regulatory programs and enables compact summaries of complex predictions.

Model and dataset auditing uses interpretability tools to identify reliance on confounded or undesirable features. Cross-referencing with the confounder taxonomy from Chapter 16 helps design deconfounded training and evaluation schemes. If interpretability reveals that a model relies heavily on GC content or batch-specific signals, this diagnoses a problem that evaluation metrics alone might miss.

Human-in-the-loop analysis integrates motif and sequence-class outputs into visualization tools such as genome browsers with attribution tracks, motif annotations, and class scores. Domain experts can then iteratively refine hypotheses, identifying patterns that merit experimental follow-up and flagging predictions that seem biologically implausible.

17.10. Outlook: From Explanations to Mechanistic Models

Interpretability in genomic deep learning is evolving from post hoc explanation toward model-assisted mechanistic discovery. Foundation models provide rich latent spaces and long-range context that capture regulatory information at unprecedented scale. Attribution and motif discovery tools translate those representations into candidate regulatory grammars that can be tested experimentally.

Global vocabularies like Sei's sequence classes offer interpretable axes spanning thousands of assays, enabling systematic characterization of regulatory programs across the genome.

Attention analysis in genomic language models reveals emergent gene-level organization, suggesting that models trained on raw sequence implicitly learn operon structure, co-regulation patterns, and phylogenetic context. These findings hint at scalable ways to capture systems-level biology from sequence alone, complementing the multi-omic integration approaches discussed in Chapter 14.

The next frontier is to close the loop between interpretability and model development. Insights from interpretability (motifs, grammars, sequence classes) can inform better architectures and training objectives. Experimentally validated grammars can be fed back into models as inductive biases, constraining the hypothesis space to biologically plausible solutions. Evaluation frameworks can measure not only predictive accuracy but also mechanistic fidelity: how well do model-derived hypotheses align with the causal structure of regulatory biology revealed by perturbation experiments?

In this sense, interpretability is not merely a diagnostic for black-box models. It is a central tool for turning genomic foundation models into engines of biological discovery, capable of bridging the gap between sequence-level predictions and the mechanistic understanding that underpins robust clinical translation. When a model's explanations match experimental observations and generate validated predictions, it becomes more than a predictor: it becomes a hypothesis-generating system that accelerates the scientific enterprise.

Part VI.

Part VI: Applications

18. Clinical Risk Prediction

Warning

TODO:

- Add figure: Clinical risk prediction pipeline overview showing data flow from genotype/EHR through GFM feature extraction to risk stratification and clinical decision support
- Add figure: Calibration plot examples contrasting well-calibrated versus miscalibrated models, with stratification by ancestry
- Add figure: Model drift monitoring dashboard concept showing input distribution shifts, output score histograms over time, and performance degradation alerts
- Add figure: Multi-modal fusion architecture comparison (early, intermediate, late fusion) with representative genomic and clinical inputs
- Add table: Evaluation metrics summary (discrimination, calibration, clinical utility) with appropriate use cases and limitations
- Add table: Regulatory considerations for GFM-based clinical decision support systems across FDA, CE marking, and other frameworks
- Consider adding decision curve example for cardiometabolic risk stratification showing net benefit across threshold probabilities

Modern genomic foundation models provide increasingly rich representations of DNA, RNA, proteins, and multi-omic context. The preceding parts of this book have traced how these models learn from sequence and structure, predict molecular functions, and integrate information across biological scales. The natural next question is practical: how do we turn these representations into actionable predictions for individual patients?

This chapter focuses on clinical risk prediction and decision support, the task of estimating the probability, timing, or trajectory of outcomes such as incident disease, progression, recurrence, or adverse drug reactions. The discussion emphasizes how genomic foundation models and related deep learning approaches extend traditional polygenic scores with richer sequence-based features and epistatic structure through methods like Delphi, G2PT, and MIFM (Georgantas, Kutalik, and Richiardi 2024; Lee et al. 2025; Rakowski and Lippert 2025). These models combine genomic features with electronic health records and multi-omics to produce holistic patient-level risk representations, building on the systems-level integration strategies introduced in Chapter 14. Throughout, the emphasis is on evaluation, calibration, uncertainty quantification, and fairness considerations that are essential for high-stakes clinical decisions.

The chapter concludes with case studies in cardiometabolic risk, oncology risk and recurrence, and pharmacogenomics, illustrating how foundation models move from computational representations to clinical utility.

18.1. Problem Framing: What Is Clinical Risk Prediction?

Clinical risk prediction is the task of mapping patient data to probabilistic statements about future outcomes. The inputs include genotypes, family history, clinical measurements, imaging, and environmental factors. The outputs are probabilities or hazard estimates that answer questions such as: What is this patient's 10-year risk of coronary artery disease if treated with standard of care? Given current tumor characteristics and therapy, what is the hazard of recurrence within two years? If we start this medication, what is the probability of a severe adverse drug reaction in the next six months?

These questions fall into several archetypes that differ in their temporal structure and clinical context. Individual-level incident risk concerns whether a currently disease-free individual will develop disease within a specified time window, such as 10-year type 2 diabetes risk. Progression and complication risk asks which patients with an existing condition will develop complications, for example nephropathy in diabetes or heart failure after myocardial infarction. Prognosis and survival involve time-from-baseline to events such as death, recurrence, or transplant, often with censoring and competing risks that complicate standard regression approaches. Treatment response and toxicity prediction concerns whether a patient will benefit from one therapy versus another and their risk of severe toxicity or adverse drug reactions.

Genomic foundation models enter these problems as feature generators. They transform raw genomic and multi-omic data into structured embeddings, variant effect scores, or region-level functional annotations that can then be combined with clinical covariates in downstream prediction models. Real-world deployment typically requires fusing genomic features with electronic health records, imaging, and other omics, mirroring the multi-omics integration strategies discussed in Chapter 14.

18.2. Feature Sources for Clinical Prediction

The features that enter clinical risk models can be organized into three broad categories that draw on different parts of the foundation model landscape.

The first category comprises genomics and regulatory features derived from DNA-level models. Zero-shot variant scores from DNA foundation models such as Nucleotide Transformer, HyenaDNA, and GPN provide sequence-based predictions of variant deleteriousness without requiring trait-specific training (Dalla-Torre et al. 2023; Nguyen et al. 2023; Benegas, Batra, and Song 2023). Coding variant scores from protein language models, including systems similar to AlphaMissense (discussed in earlier chapters), capture the impact of missense mutations on protein structure and function. Fine-mapped causal variant probabilities from methods like MIFM provide posterior estimates of which variants within a GWAS locus are likely causal, allowing risk models to weight variants by their evidence for causality rather than treating all associated variants equally (Rakowski and Lippert 2025).

The second category encompasses multi-omics and systems context features. Cell-type-resolved epigenomic and transcriptomic embeddings from frameworks like GLUE, SCGLUE, and CpGPT capture regulatory state across chromatin accessibility, methylation, and expression (Cao and Gao 2022; Camillo et al. 2024). Rare-variant burden and pathway-level representations from DeepRVAT aggregate the predicted effects of multiple rare variants into gene-level or pathway-level impairment scores (Clarke et al. 2024). Tumor-level representations from models such as SetQuence and

SetOmic, or from graph neural network-based cancer subtypers, encode the complex mutational landscapes of individual tumors (Jurenaite et al. 2024; X. Li et al. 2022; Hao Li et al. 2024).

The third category includes clinical covariates and electronic health record data. Demographics, vitals, laboratory results, and medication history provide non-genomic risk factors that often have substantial predictive power. Problem lists, procedures, and imaging-derived features add diagnostic context. Time-varying trajectories of biomarkers such as estimated glomerular filtration rate, hemoglobin A1c, or tumor markers capture disease dynamics that static snapshots miss.

18.3. Fusion Architectures

Architecturally, risk models that combine these feature sources typically adopt one of the fusion strategies echoed from Chapter 14, each with distinct tradeoffs.

Early fusion concatenates foundation model-derived genomic embeddings with static clinical covariates and feeds them into a single model such as a multilayer perceptron or survival regression. This approach is simple to implement and allows the model to learn arbitrary interactions between genomic and clinical features. However, early fusion is sensitive to differences in scale between modalities, handles missing data poorly since samples lacking one modality must be imputed or excluded, and can be dominated by whichever input has the most features or highest signal-to-noise ratio.

Intermediate fusion trains separate encoders for genomics, electronic health records, and multi-omics that produce modality-specific embeddings. A fusion layer, which might use attention mechanisms, cross-modal transformers, or graph-based integration, then combines these embeddings into a patient-level representation that downstream prediction heads use for risk estimation. Intermediate fusion is often most attractive from a practical standpoint because it allows modularity (foundation model encoders can be swapped as new versions become available) while still enabling cross-modal interactions that can capture how genomic risk manifests differently depending on clinical context.

Late fusion trains independent models for each modality, such as a polygenic score-only model and an electronic health record-only model, then combines their predictions through ensemble methods or a meta-model. This approach is robust to missing modalities since each sub-model operates independently, and it allows each modality to use whatever architecture works best for its data type. However, late fusion may underutilize cross-modal structure since interactions between genomic and clinical features can only be captured at the final combination stage rather than learned jointly.

18.4. Evaluation: Discrimination, Calibration, and Clinical Utility

High performance on held-out test sets is necessary but not sufficient for clinical deployment. Risk models must be discriminative, well-calibrated, robust to distribution shift, and clinically useful in ways that justify the costs of implementation.

18.4.1. Discrimination

Discrimination measures how well a model ranks individuals by risk, distinguishing those who will experience an outcome from those who will not. For binary endpoints such as disease occurrence within a fixed time window, the area under the receiver operating characteristic curve (AUROC) summarizes discrimination across all possible classification thresholds. When outcomes are rare, as is often the case for severe adverse drug reactions or specific disease subtypes, the area under the precision-recall curve (AUPRC) is more informative because it is sensitive to how well the model identifies true positives among many negatives. For survival tasks with time-to-event outcomes and censoring, the concordance index (C-index) and time-dependent AUC generalize discrimination metrics to the survival setting.

Strong discrimination is necessary but not sufficient. A model that ranks patients correctly but systematically overestimates or underestimates absolute risks will lead to inappropriate clinical decisions. For a broader discussion of how discrimination metrics are used across molecular, variant-level, and trait-level tasks, see Chapter 15.

18.4.2. Calibration and Risk Stratification

Calibration asks whether predicted probabilities match observed frequencies. If a group of patients is assigned 20% risk of an event, approximately 20% of that group should actually experience it. Well-calibrated predictions can be taken at face value and used directly for clinical decision-making, whereas miscalibrated predictions mislead clinicians and patients regardless of how good the discrimination is.

Calibration is assessed through calibration plots that compare predicted risk deciles to observed event rates, statistical tests like the Hosmer-Lemeshow test, and proper scoring rules like the Brier score that combine calibration and discrimination into a single metric. These assessments should be stratified by clinically relevant subgroups such as ancestry, sex, and age, since a model that is well-calibrated overall may be systematically miscalibrated for specific populations.

For polygenic score-informed models, calibration is especially important because raw polygenic scores are often centered and scaled rather than calibrated to absolute risk. Mapping a polygenic score to an absolute probability of disease typically requires post-hoc models that incorporate baseline incidence and clinical covariates. Foundation models can shift score distributions as architectures evolve, meaning that recalibration may be required when swapping or updating encoders.

18.4.3. Uncertainty Estimation

In high-stakes clinical settings, models should know when they do not know. Uncertainty quantification allows models to flag predictions where confidence is low, either because the input is unusual or because the model has limited evidence for its predictions.

Common approaches to uncertainty estimation include ensemble variance, where multiple models trained with different random seeds provide prediction intervals based on their disagreement, and Monte Carlo dropout, which approximates Bayesian uncertainty by averaging predictions across multiple stochastic forward passes. Conformal prediction provides a more principled framework

for outputting risk intervals or prediction sets with guaranteed coverage under exchangeability assumptions.

For foundation model-based systems, uncertainty can be decomposed into genomic uncertainty (confidence in variant effect predictions, fine-mapping probabilities, or embedding reliability) and clinical uncertainty (extrapolation to new care settings, practice patterns, or patient populations). Selective prediction or abstention allows models to decline to make predictions on cases where uncertainty is high or inputs are out-of-distribution, such as patients from rare ancestries missing from training data or novel tumor subtypes that the model has not encountered. Communicating uncertainty transparently is a core component of responsible decision support.

18.4.4. Fairness, Bias, and Health Equity

Many genomic and electronic health record datasets reflect historical and structural inequities in who is genotyped, which populations are recruited into biobanks, and how healthcare is documented and delivered. Risk models can amplify these biases if not carefully evaluated and designed.

Ancestry and polygenic score portability remain central concerns. As discussed in Chapter 3, classical polygenic scores substantially underperform in under-represented ancestries due to the European bias in GWAS design. Foundation model-based methods such as Delphi and G2PT have the opportunity, but not the guarantee, to improve portability by leveraging functional priors and cross-ancestry information (Georgantas, Kutalik, and Richiardi 2024; Lee et al. 2025). Whether they succeed depends on training data composition, evaluation practices, and explicit attention to cross-ancestry performance.

Measurement and access bias affect electronic health record features. Which patients get genotyped, which laboratory tests are ordered, how diagnoses are coded, and how thoroughly clinical notes are documented all differ systematically across patient populations, care settings, and health systems. A model trained on one system's data may encode these institutional patterns rather than underlying biology.

Group-wise evaluation is essential. Calibration and discrimination should be assessed separately by ancestry, sex, socioeconomic proxies, and care site. A model that appears well-calibrated overall but is miscalibrated for specific groups will exacerbate rather than reduce health disparities. When necessary, fairness constraints such as equalized odds or affirmative designs targeting historically disadvantaged groups can be incorporated into model training, though such constraints involve tradeoffs with overall performance that must be navigated thoughtfully.

Equity is not an afterthought. For foundation models, it should inform what data to pretrain on, which benchmarks to report, and how to deploy models in practice.

18.5. Prospective Validation, Trials, and Regulation

Retrospective performance metrics, even when computed on held-out test sets with appropriate splitting strategies, are not sufficient to justify clinical use. Clinical risk models typically require several additional layers of validation before deployment.

Prospective validation evaluates model performance in a temporally held-out cohort, ideally in multiple health systems with different population structures and practice patterns. A model trained

on data from 2015-2020 should be tested on patients from 2021-2023 to assess whether it generalizes across time. Multi-site validation tests whether a model trained at one institution transfers to others with different patient populations, sequencing platforms, and clinical workflows.

Impact studies measure whether using the model actually changes clinician behavior and improves patient outcomes. A risk model might achieve excellent discrimination and calibration, but if clinicians do not trust it, do not integrate it into their workflow, or override its recommendations based on unmeasured factors, the model will have no clinical impact. Demonstrating that a model leads to better statin targeting, fewer adverse drug reactions, or reduced unnecessary imaging requires prospective studies that compare outcomes between patients whose clinicians used the model and those whose clinicians did not.

Randomized or pragmatic trials provide the strongest evidence when models materially influence treatment decisions. Observational evaluations, even prospective ones, cannot fully account for confounding between which patients receive model-guided care and which do not. For high-stakes decisions like treatment selection, randomization may be necessary to demonstrate causal benefit.

Regulatory landscapes increasingly recognize learning systems and continuous updates. Foundation models complicate this further because a "fixed" risk model may rely on a backbone that improves over time. Updates to the foundation model can change risk rankings and calibration even if the downstream prediction head remains unchanged. Regulatory strategies include locked models with explicit versions that require reapproval for each update, change control plans that prespecify acceptable ranges of performance drift, and adaptive approvals that allow constrained forms of continual learning under monitoring requirements.

Regardless of the regulatory framework, clear documentation of data provenance, foundation model versions, training procedures, and validation results is essential for both regulatory compliance and scientific reproducibility.

18.6. Monitoring, Drift, and Continual Learning

Once deployed, foundation models and downstream risk models operate in non-stationary environments. Clinical practice patterns change as new treatments and guidelines emerge. Patient populations drift as screening programs expand or contract. Laboratory assays and sequencing pipelines evolve, introducing subtle distributional shifts in input features.

Monitoring systems should track input distributions such as genotype frequencies and electronic health record feature patterns to detect when the current patient population differs from the training population. Output distributions including risk score histograms and the fraction of patients above decision thresholds reveal whether model behavior is changing over time. Performance metrics over time, often computed via rolling windows or periodic audits, detect calibration or discrimination degradation before it becomes clinically consequential.

When drift is detected, several responses are possible depending on severity and type. Recalibration may suffice if the model's ranking behavior remains sound but the mapping from scores to probabilities has shifted. Refitting a calibration layer to current data can restore well-calibrated predictions without retraining the entire model. Partial retraining of prediction heads or fusion layers can adapt to new environments while keeping foundation model weights fixed, preserving regulatory status of the backbone while adjusting to local conditions. Full continual learning, including updating

foundation model backbones, requires careful safeguards to avoid catastrophic forgetting (where the model loses performance on previously well-handled cases) and maintain regulatory compliance.

The modular design patterns from Chapter 14, with clear interfaces between foundation encoders and clinical prediction layers, are crucial for maintainable and updatable decision support systems.

18.7. Case Studies

To make these ideas concrete, we examine three stylized case studies that build on models and concepts from earlier chapters. Each illustrates different aspects of foundation model integration into clinical risk prediction.

18.7.1. Cardiometabolic Risk Stratification

The goal of cardiometabolic risk stratification is to identify individuals at high risk of major adverse cardiovascular events, including myocardial infarction, stroke, and cardiovascular death, over a time horizon such as 10 years. This is among the most mature applications of genomic risk prediction, with established clinical frameworks like the Framingham Risk Score and ASCVD Risk Estimator providing baselines against which genomic augmentation can be evaluated.

The inputs for such a model combine genotype data from biobank-scale genotyping or whole-genome sequencing with foundation model features and clinical data. Variant effect scores from DNA foundation models like Nucleotide Transformer, HyenaDNA, and GPN provide sequence-based annotations for variants in cardiometabolic risk loci (Dalla-Torre et al. 2023; Nguyen et al. 2023; Benegas, Batra, and Song 2023). Polygenic models like Delphi or G2PT produce patient-level genomics embeddings tuned for cardiometabolic outcomes (Georgantas, Kutalik, and Richiardi 2024; Lee et al. 2025). Clinical data including age, sex, body mass index, blood pressure, lipids, smoking status, diabetes status, and current medications provide the non-genomic risk factors that drive most of the predictive signal in traditional risk scores.

A model design for this application might proceed in several stages. First, a DNA foundation model computes variant-level annotations such as predicted enhancer disruption in cardiomyocyte or hepatocyte contexts. Second, these annotations and genotypes feed into Delphi or G2PT to obtain a patient-level genomics embedding tuned for cardiometabolic outcomes. Third, an intermediate fusion network combines the genomics embedding with electronic health record covariates. Finally, the fused representation trains to predict 10-year major adverse cardiovascular event risk using survival or discrete-time hazard losses.

In clinical use, such a model would stratify patients into risk categories that inform statin initiation, consideration of PCSK9 inhibitors, or intensive lifestyle intervention. Individual-level explanations, drawing on G2PT attention weights or Delphi variant contributions, would highlight which variants and pathways most contributed to risk, connecting the prediction to interpretable biology. Equity evaluation would ensure that performance and calibration hold across ancestries and care sites, avoiding the portability failures that plague traditional polygenic scores.

18.7.2. Oncology: Risk and Recurrence Prediction

In oncology, the goal is often to predict recurrence risk and treatment benefit for patients with solid tumors after surgery or first-line therapy. Unlike cardiometabolic risk where germline variants dominate, oncology applications must integrate somatic mutation landscapes with germline background and multi-omic tumor characterization.

The inputs combine somatic landscapes from whole-exome or whole-genome tumor sequencing with tumor representations from deep set or transformer architectures such as SetQuence and SetOmic (Jurenaite et al. 2024). Multi-omics profiles of tumor expression, methylation, and chromatin can be integrated through frameworks like GLUE and CpGPT (Cao and Gao 2022; Camillo et al. 2024). Graph neural network-based subtyping from models like MoGCN and CGMega provides embeddings or cluster assignments that capture tumor subtype structure (X. Li et al. 2022; Hao Li et al. 2024). Clinical features including stage, grade, performance status, and treatment regimen provide essential prognostic context.

The model design encodes somatic mutation sets with SetQuence or SetOmic to obtain tumor-variant embeddings. Transcriptomic and epigenomic profiles integrate via GLUE-like latent spaces and CpGPT methylation embeddings. These combine with graph neural network-based subtype embeddings to capture tumor-microenvironment and histopathological context. The fused tumor-level representations join with clinical features in a time-to-recurrence model using flexible deep survival networks.

Clinical use would provide risk estimates that guide adjuvant therapy decisions, such as intensifying chemotherapy or adding targeted agents for high-risk patients. Candidate biomarkers or pathways identified through foundation model importance scores and attention maps could inform trial stratification. Continuous monitoring would track drift as treatment standards evolve, updating models to reflect new targeted therapies and immune checkpoint inhibitors that change the baseline hazard.

18.7.3. Pharmacogenomics and Adverse Drug Reaction Risk

The goal of pharmacogenomic risk prediction is to identify patients at high risk of severe adverse drug reactions before initiating therapy. Examples include myopathy on statins, severe cutaneous adverse reactions to certain antibiotics and anticonvulsants, and cardiotoxicity from oncology agents. Some pharmacogenomic associations, such as the HLA-B*5701 association with abacavir hypersensitivity, are well-established and already implemented clinically (Mallal et al. 2008). Foundation models offer the potential to extend such predictions to variants and drugs without established single-gene associations.

The inputs include germline variation in pharmacogenes such as the CYP family and HLA alleles, along with variants across the broader genome that might modulate drug metabolism or immune responses. Variant effect scores from both DNA and protein language models provide predictions of how coding and regulatory variants affect drug metabolism and immune genes. Clinical context including co-medications, comorbidities, organ function (particularly liver and kidney), and prior adverse reactions provides essential non-genomic risk factors.

The model design uses foundation models to derive mechanistically meaningful features for variants in pharmacogenes, such as predicted impact on protein stability, binding affinity, or gene regulation.

These features aggregate across loci into a pharmacogenomic risk embedding, possibly using a G2PT-style transformer restricted to relevant genes (Lee et al. 2025). The genomic embedding combines with electronic health record data in a multi-task classification model that predicts adverse reaction risk for multiple drugs or drug classes simultaneously, sharing representation learning across related prediction tasks.

Clinical use would flag patients at high risk before initiating therapy, prompting genotype-guided drug choice or dose adjustment. Reports would tie risk predictions back to specific variants and pharmacogenes, aligned with existing clinical pharmacogenomics guidelines from organizations like CPIC and PharmGKB. Cross-ancestry evaluation would ensure that the model does not exacerbate existing disparities in access to safe and effective therapy, a particular concern given the European bias in pharmacogenomics research.

18.8. Practical Design Patterns and Outlook

Across these case studies, several design patterns for foundation model-enabled clinical prediction recur. Treating foundation models as modular feature extractors, with clear separation between encoders and clinical prediction heads, eases updates and regulatory management. Embracing multi-modal fusion that combines genotype, multi-omics, and electronic health records takes advantage of the integration architectures developed throughout this book. Prioritizing calibration, uncertainty, and fairness as first-class design constraints rather than post-hoc additions ensures that models are suitable for high-stakes decisions. Bridging interpretability and mechanism, using the tools from Chapter 17 to connect individual risk predictions to variants, regions, and pathways, enables mechanistic hypotheses and clinician trust. Designing for continual learning and monitoring assumes that clinical practice and data distributions will change and builds pipelines that can adapt responsibly.

In the broader arc of this book, clinical risk prediction and decision support represent a key translation layer that connects the representational gains of genomic foundation models to the realities of patient care. The next chapters extend these ideas to other application domains: pathogenic variant discovery in rare disease and cancer workflows (Chapter 19), and drug discovery and biotech applications (Chapter 20), further exploring how foundation models reshape translational genomics.

19. Pathogenic Variant Discovery

> ⚠ Warning
>
> **TODO:**
>
> - Add figure: variant prioritization pipeline flowchart showing GFM integration points (from raw variants through filtering, VEP scoring, aggregation, and final ranking)
> - Add figure: schematic of DeepRVAT-style deep set architecture for rare variant aggregation
> - Add figure: knowledge graph visualization showing PrimeKG structure with gene nodes, disease associations, and multi-omic edges
> - Add figure: closed-loop discovery workflow diagram illustrating the hypothesis factory concept (model prediction → experimental validation → model refinement cycle)
> - Add table: comparison of GFM-based VEP tools (AlphaMissense, GPN-MSA, Evo 2, AlphaGenome) with their variant classes, training data, and key strengths
> - Add table: summary of graph-based gene prioritization methods (MoGCN, CGMega, GLUE) with architectures and applications
> - Consider case study box: worked example of a rare disease diagnosis using GFM-enhanced pipeline
> - Consider case study box: noncoding driver discovery in cancer using regulatory GFMs

Clinical genetics ultimately cares about specific variants and genes: which changes in a patient's genome plausibly explain their phenotype, and which loci are compelling targets for follow-up in the lab. The previous chapters focused on foundation models for variant effect prediction (Chapter 13), multi-omics integration (Chapter 14), and clinical risk prediction (Chapter 18). This chapter shifts the emphasis from prediction to discovery workflows.

The central question is: given a huge space of possible variants and genes, how can genomic foundation models help us efficiently home in on those most likely to be causal? We will treat "pathogenic" broadly, covering both Mendelian variants with large effects and complex trait variants that modulate risk more subtly. Genomic foundation models appear at multiple stages of these pipelines. They serve as variant-level effect predictors, exemplified by AlphaMissense, GPN-MSA, Evo 2, and AlphaGenome, that score coding and noncoding changes (Cheng et al. 2023; Benegas, Albors, et al. 2024; Brixi et al. 2025; Z. Avsec, Latysheva, and Cheng 2025). They function as inputs or priors for fine-mapping and rare variant association tests (Wu et al. 2024; Rakowski and Lippert 2025; Clarke et al. 2024). They provide node features in gene and network models, including graph neural networks over multi-omics and knowledge graphs (Cao and Gao 2022; X. Li et al. 2022; Hao Li et al. 2024; Chandak, Huang, and Zitnik 2023). And they guide the design of CRISPR, MPRA, and other functional assays, closing the loop between in silico prediction and experimental validation (Ž. Avsec et al. 2021; Linder et al. 2025).

This chapter walks through these roles from locus-level variant ranking, to Mendelian disease diagnostics, to graph-based gene prioritization, and finally to closed-loop workflows that blend foundation models with systematic perturbation experiments.

19.1. From Variant Effect Prediction to Prioritization

Chapter 13 surveyed state-of-the-art variant effect prediction systems. Models such as AlphaMissense, GPN-MSA, Evo 2, and AlphaGenome assign each variant a score reflecting predicted impact on protein function, regulatory activity, or multi-omic phenotypes (Cheng et al. 2023; Benegas, Albors, et al. 2024; Brixi et al. 2025; Z. Avsec, Latysheva, and Cheng 2025). In isolation, these scores are powerful but not yet a full prioritization pipeline. Discovery workflows require several additional steps that transform raw predictions into actionable rankings.

19.1.1. Contextualizing Variant Scores

A raw variant effect score has different implications depending on the variant class, gene context, and clinical question at hand. For variant class, a moderately damaging missense variant carries different weight than a predicted splice-site disruption or an enhancer variant that subtly alters transcription factor binding. For gene context, a variant in a highly constrained gene with tissue-specific expression in the relevant organ is more compelling than an equally scored variant in a gene with no biological connection to the phenotype. For clinical context, the threshold of evidence differs between dominant Mendelian disease (where a single heterozygous variant may suffice), recessive disease (requiring biallelic variants), and complex trait modifiers (where many variants of small effect accumulate).

Consider a concrete example: a moderately damaging missense variant in a highly constrained gene expressed in the relevant tissue may be more compelling than a strongly damaging variant in a gene with no supporting biology. The variant effect score alone cannot capture this distinction. Effective prioritization requires integrating the score with gene-level constraint metrics, tissue expression profiles, pathway annotations, and phenotype matching.

19.1.2. Aggregating Variants to Loci and Genes

Discovery problems often operate at the locus or gene level, requiring some aggregation of variant scores. Several strategies have emerged for this aggregation. Max or top-k pooling focuses on the worst predicted variant per gene or locus, on the theory that a single highly damaging variant may be sufficient to disrupt gene function. This approach works well for loss-of-function mechanisms but may miss genes where multiple moderate variants accumulate to cause dysfunction.

Burden-style aggregation sums or averages the predicted impact of all rare variants in a gene, possibly weighted by allele frequency and predicted effect size. This approach captures scenarios where multiple variants contribute to gene dysfunction but requires careful handling of variant independence assumptions. Mechanism-aware aggregation separates coding versus regulatory contributions, or promoter versus distal enhancer effects, using tissue-specific scores from models like Enformer or AlphaGenome (Ž. Avsec et al. 2021; Z. Avsec, Latysheva, and Cheng 2025). This approach

recognizes that different variant classes may act through distinct biological mechanisms and deserve separate treatment in prioritization.

19.1.3. Combining VEP with Orthogonal Evidence

Variant effect prediction is rarely used alone in modern discovery pipelines. Effective prioritization combines VEP scores with multiple orthogonal evidence streams. Population data from resources like gnomAD provide allele frequency and constraint information, including metrics like pLI, LOEUF, and missense and loss-of-function intolerance scores that indicate which genes are sensitive to damaging variation. Clinical databases like ClinVar and HGMD provide expert-curated variant classifications and disease-gene associations that anchor new predictions to established knowledge. Functional annotations from conservation scores (PhyloP, PhastCons), chromatin state maps, and known regulatory element catalogs provide biological context for variant interpretation (Siepel et al. 2005). Pathway and network context, including membership in pathways enriched for the trait or centrality in relevant biological networks, helps prioritize genes with plausible mechanistic connections to the phenotype.

Genomic foundation models enter this stack as feature providers, often replacing or augmenting hand-crafted features with learned representations that capture more nuanced sequence-function relationships.

19.1.4. Calibration and Interpretability

For prioritization tasks, ranking performance may matter more than perfectly calibrated probabilities, but interpretable risk categories remain crucial in clinical and experimental settings. This pushes toward several practices. Score thresholds should be associated with empirical positive predictive value estimates, allowing users to understand the trade-off between sensitivity and specificity at different cutoffs. Qualitative explanations, such as "strong disruption of a conserved splice donor in a haploinsufficient gene," help clinicians and researchers understand why a variant was flagged. Visualizations of attention maps, saliency, or motif-level contributions (Chapter 17) can reveal what sequence features drove the prediction, supporting mechanistic interpretation.

In other words, genomic foundation models provide high-resolution local perturbation scores, but the art of discovery lies in wiring those scores into larger decision frameworks that account for biological context, clinical relevance, and the practical constraints of follow-up experiments.

19.2. Rare Variant Association and Complex Trait Discovery

In the GWAS paradigm discussed in Chapter 3, common variants are tested individually for association with phenotypes. For rare variants, which are individually too infrequent to achieve statistical significance, this approach fails. Instead, gene- or region-based burden tests aggregate rare variants across individuals to detect association signals. Here, variant effect prediction plays two key roles.

19.2.1. Variant Weighting and Filtering

Classical burden tests often restrict analysis to "damaging" variants using simple filters: predicted loss-of-function variants, or variants exceeding a CADD score threshold. These binary filters discard information about the continuous spectrum of predicted effects. Genomic foundation models provide richer filters and weights that enable more nuanced analysis. Fine-grained distinctions among missense variants become possible using AlphaMissense scores, which provide continuous pathogenicity estimates across the proteome (Cheng et al. 2023). Regulatory variants predicted to modulate gene expression can be included in burden calculations, expanding the analysis beyond coding sequence. Continuous weights reflecting predicted effect size, rather than binary include/exclude decisions, allow the statistical framework to incorporate uncertainty about variant pathogenicity.

19.2.2. End-to-End Deep Set Models

DeepRVAT exemplifies a newer paradigm for rare variant association. Rather than hand-engineering burden summaries from predefined features, a deep set network ingests per-variant features (including foundation-model-derived VEP scores) and learns to aggregate them into a gene-level risk signal (Clarke et al. 2024). This approach offers several advantages over traditional methods.

The architecture supports heterogeneous variant classes within a gene, allowing the model to learn how coding, regulatory, and splice variants contribute differently to gene dysfunction. The aggregation function is learned rather than specified, enabling the model to discover non-additive interactions while preserving permutation invariance across variants. The framework naturally accommodates multiple phenotypes and covariates within a single model, enabling joint analysis across related traits.

As more cohorts with whole-exome or whole-genome sequencing become available, these foundation-model-enhanced burden frameworks blur the line between GWAS and rare variant analysis, providing a continuum of variant discovery tools that span the allele frequency spectrum.

19.3. Mendelian Disease Gene and Variant Discovery

In Mendelian disease genetics, the questions tend to be more concrete: which variant explains this patient's phenotype, and which gene is implicated? Whole-exome or whole-genome sequencing of trios and families produces thousands of variants per individual, and the diagnostic challenge is to identify the one or few variants that are pathogenic from this large background.

19.3.1. The Standard Diagnostic Pipeline

The traditional approach to Mendelian variant prioritization follows a structured workflow. Quality control and filtering removes low-quality calls and technical artifacts, then applies allele frequency filters (typically less than 0.1% in population databases), inheritance mode filters (de novo, recessive, X-linked), and variant type filters (loss-of-function, missense, splice, structural). Gene-centric ranking

aggregates candidate variants per gene, incorporating constraint metrics from gnomAD and known disease-gene catalogs from OMIM and other resources. Phenotype similarity, often computed using HPO-based matching between patient phenotypes and known gene syndromes, further prioritizes candidates. Manual curation by clinical geneticists reviews gene function, expression patterns, animal models, and literature, assessing segregation in the family, de novo status, and evidence of pathogenic mechanism.

19.3.2. Genomic Foundation Models in Mendelian Diagnostics

Genomic foundation models reshape several stages of this process. For coding impact assessment, AlphaMissense provides proteome-wide missense pathogenicity estimates with continuous scores that often outperform traditional tools (Cheng et al. 2023). Coding-aware foundation models like cdsFM further capture codon-level context and co-evolutionary patterns, providing richer representations of protein-level effects (Naghipourfar et al. 2024).

For regulatory and splice prediction, genome-wide models like GPN-MSA, Evo 2, and AlphaGenome estimate the effect of noncoding and splice-proximal variants, filling a critical gap for Mendelian variants outside exons (Benegas, Albors, et al. 2024; Brixi et al. 2025; Z. Avsec, Latysheva, and Cheng 2025). These models can flag deep intronic variants that create cryptic splice sites or regulatory variants that disrupt enhancer function, classes of variants that traditional pipelines often miss.

Combined variant-gene scoring integrates these multiple evidence streams. For each gene, one can aggregate the maximum or weighted VEP score across all candidate variants, maintain separate tallies for loss-of-function, missense, regulatory, and splice variants, and incorporate gene-level features (constraint, expression, pathways) and phenotype similarity. A simple model might compute a composite gene score as a learned function of these features, trained on cohorts with labeled diagnoses.

19.3.3. Rare Disease Association at Scale

Beyond single-family diagnostics, large consortia collect rare disease cohorts where the goal is to discover new gene-disease associations rather than diagnose individual patients. DeepRVAT-style models provide one blueprint for this analysis. The approach represents each individual as a set of rare variants with multi-dimensional VEP features drawn from foundation models and traditional tools. Deep set networks map from per-variant features to individual-level phenotype predictions or gene-level association signals (Clarke et al. 2024). Multi-omics context from GLUE-like models, including tissue-specific expression and chromatin accessibility, provides additional features that help distinguish signal from noise (Cao and Gao 2022).

This approach pushes Mendelian discovery closer to the foundation model paradigm: instead of hand-designed burden statistics, we train flexible architectures that learn how to combine variant-level representations into gene- and phenotype-level insights.

19.4. Graph-Based Prioritization of Disease Genes

Many discovery problems are inherently network-structured. Genes interact through pathways, protein-protein interaction networks, co-expression modules, regulatory networks, and knowledge graphs. Graph neural networks offer a natural way to fuse node features from foundation models (such as aggregated VEP scores and expression profiles) with graph structure capturing biological relationships, learning to predict labels such as disease associations, essentiality, or cancer driver status.

19.4.1. Multi-Omics Integration and Cancer Gene Modules

GLUE and SCGLUE frame multi-omics integration as a graph-linked embedding problem, connecting cells and features across modalities (Cao and Gao 2022). In the context of cancer driver discovery, methods like MoGCN apply graph convolutional networks to multi-omics data, learning gene-level representations that capture both sequence-level features and network context (X. Li et al. 2022). CGMega extends this approach to identify driver gene modules that are recurrently perturbed across patients, moving from individual gene ranking to pathway-level hypotheses (Hao Li et al. 2024).

19.4.2. Knowledge Graphs for Target Prioritization

Knowledge graphs like PrimeKG integrate diverse relationship types, including gene-disease associations, drug-target interactions, pathway membership, and protein-protein interactions, into a unified graph structure (Chandak, Huang, and Zitnik 2023). Graph neural networks can then propagate information across this structure, allowing known disease associations to inform prioritization of novel candidate genes.

In practice, a graph-based prioritization workflow might proceed as follows. First, construct a multi-relational graph with genes as nodes and edges representing protein interactions, pathway co-membership, regulatory relationships, and phenotype associations. Second, initialize node features using foundation model outputs: aggregated VEP scores for variants in each gene, expression embeddings from single-cell or bulk RNA-seq, and constraint metrics. Third, train a graph neural network to predict known disease-gene associations or cancer driver status. Fourth, apply the trained model to score candidate genes, prioritizing those with high predicted scores and biologically plausible network context.

This approach naturally handles the fact that disease genes tend to cluster in biological networks: perturbation of one gene in a pathway often implicates functionally related genes, and graph-based methods can capture these dependencies.

19.5. Closed-Loop Discovery: Foundation Models, Perturbation, and Iteration

The most powerful use of genomic foundation models in variant discovery may be in closed-loop workflows that integrate computational prediction with experimental validation. Rather than treating models as static predictors that output final rankings, this paradigm treats them as hypothesis generators that guide experimental design and improve through feedback.

19.5.1. The Hypothesis Factory Concept

In the limit, we approach a semi-automated "hypothesis factory" workflow. The process begins with GWAS, rare variant, or tumor sequencing data that identifies candidate loci. Foundation models plus graph-based methods prioritize candidate variants and genes from these data. Perturbation experiments, including CRISPR screens, MPRA assays, or functional genomics studies, are designed to test the top-ranked hypotheses. Experimental results provide new functional data that update the models, refining predictions for the next round. The cycle iterates, progressively sharpening our understanding of the underlying mechanisms.

19.5.2. Guiding Experimental Design

Foundation models can guide experimental design in several ways. For CRISPR tiling screens, models like Enformer can predict which regulatory elements are most likely to affect expression of a target gene, allowing focused tiling around high-priority regions rather than exhaustive screening (Ž. Avsec et al. 2021). For MPRA design, variant effect predictions can identify which candidate variants are most likely to show allelic effects in reporter assays, improving the hit rate and reducing the number of elements that need to be tested. For functional follow-up of GWAS hits, foundation model attributions can suggest which transcription factor binding sites or enhancer motifs are disrupted by risk variants, guiding mechanistic experiments.

19.5.3. Updating Models with Experimental Feedback

As experimental data accumulate, they can feed back into model training and evaluation. New MPRA results provide ground truth for regulatory variant effect prediction, allowing models to be fine-tuned or recalibrated on relevant cell types and contexts. CRISPR screen hits identify validated enhancer-gene pairs that can improve training data for long-range regulatory models. Functional validation of candidate disease genes updates the labels available for graph-based prioritization methods.

This closed-loop paradigm represents a shift from models as one-time predictors to models as components of iterative discovery systems that improve through use.

19.6. Case Studies and Practical Considerations

To ground these ideas, consider two representative application areas that illustrate how foundation model-enhanced pipelines operate in practice.

19.6.1. Rare Disease Diagnosis Pipelines

Modern rare disease centers increasingly adopt foundation model-enhanced diagnostic workflows. The process begins with variant filtering and annotation: standard quality control and frequency filters followed by annotation with foundation model-based VEP scores for coding, regulatory, and splice variants, constraint metrics, and ClinVar evidence. A gene-ranking model then performs per-gene aggregation of variant scores and features, using a trained model that predicts the likelihood of each gene being causal based on retrospective cohorts with known diagnoses. Phenotype integration adds HPO-based similarity to known gene syndromes and network-based propagation of phenotype associations using knowledge graphs like PrimeKG (Chandak, Huang, and Zitnik 2023). Finally, expert review by geneticists and clinicians inspects the top-ranked genes and variants, cross-checking against patient phenotypes, family segregation, and literature.

Compared to traditional pipelines, the foundation model-enhanced version tends to surface non-obvious candidates, such as noncoding or splice variants with strong predicted functional effects that would be filtered out by traditional approaches. It provides more nuanced prioritization among multiple missense variants in the same gene, distinguishing likely pathogenic changes from tolerated polymorphisms. And it offers richer mechanistic hypotheses to guide follow-up experiments, connecting variant-level predictions to specific molecular mechanisms.

19.6.2. Cancer Driver Mutation Discovery

In cancer genomics, the goal is to distinguish driver mutations from a large background of passenger mutations. Foundation models and graph-based methods contribute at multiple levels. Variant-level scoring uses coding VEP models like AlphaMissense and cdsFM-like architectures for missense drivers (Cheng et al. 2023; Naghipourfar et al. 2024), and regulatory sequence models like Enformer, AlphaGenome, and TREDNet to evaluate noncoding mutations in promoters and enhancers (Ž. Avsec et al. 2021; Z. Avsec, Latysheva, and Cheng 2025; Hudaiberdiev et al. 2023).

Gene- and module-level aggregation sums somatic variants per gene, weighted by predicted functional impact, then applies graph neural networks such as MoGCN and CGMega to identify driver gene modules that are recurrently perturbed across patients (X. Li et al. 2022; Hao Li et al. 2024). Set-based models akin to DeepRVAT can relate patient-specific variant sets to tumor subtypes or clinical outcomes (Clarke et al. 2024).

Functional follow-up designs focused CRISPR tiling screens around candidate regulatory elements, prioritized by foundation model predictions. Predicted driver genes are validated in cell line or organoid models, integrating transcriptional responses with multi-omic readouts (Chapter 14). These pipelines exemplify multi-scale integration: foundation models for variant-level effects, graph neural networks for network-level reasoning, and high-throughput perturbations for experimental validation.

19.7. Outlook: Towards End-to-End Discovery Systems

Biomedical discovery of pathogenic variants is moving from manual, hypothesis-driven workflows toward data- and model-driven pipelines where foundation models act as a central substrate. They transform raw sequence variation into rich, context-aware variant embeddings that capture more information than hand-crafted features. They provide priors and features for fine-mapping, rare variant association, and gene prioritization that improve statistical power and biological relevance. They guide the design of targeted perturbation experiments, which in turn provide new data to refine the models.

At the same time, several challenges remain. Robustness and generalization across ancestries, tissues, and disease cohorts is incompletely characterized, and performance often degrades on populations underrepresented in training data. Calibration and interpretability suitable for clinical and experimental decision-making requires ongoing attention, as overconfident predictions can mislead follow-up efforts. Evaluation frameworks like TraitGym that fairly compare models and reveal domain gaps are essential for continued progress (Benegas, Eraslan, and Song 2025). Ethical and regulatory considerations around automated variant classification and gene discovery in sensitive contexts demand careful attention as these tools move toward clinical deployment.

In the next chapter, we zoom out to the broader drug discovery and biotech landscape (Chapter 20), where many of these discovery building blocks are embedded in industrial-scale pipelines that span from genetic association to target validation, biomarker discovery, and eventually clinical translation.

20. Drug Discovery & Biotech

Warning

TODO:

- Add figure: Drug discovery pipeline diagram showing where genomics/GFMs enter (target ID, biomarker discovery, MoA/resistance)
- Add figure: Target discovery workflow schematic from GWAS → fine-mapping → VEP scoring → gene aggregation → ranked targets
- Add figure: Functional genomics screen design cycle showing GFM-guided library design → perturbation → readout → model refinement
- Add figure: Lab-in-the-loop GFM architecture showing hypothesis generation → experiment design → evidence integration → portfolio decisions
- Add table: Build vs. buy vs. fine-tune decision matrix with pros/cons for each strategy
- Add table: Model catalog overview showing DNA LMs, seq-to-function models, and VEP models with key characteristics
- Consider adding a case study box illustrating a complete target-to-biomarker workflow

Genomic foundation models are built to turn raw sequence and multi-omic data into reusable biological representations and fine-grained predictions (Chapter 12). Previous chapters demonstrated how these models improve variant effect prediction (**?@sec-vep**), long-range regulatory modeling (Chapter 11, Chapter 5), and disease genetics workflows (Chapter 18, Chapter 19). This chapter zooms out to ask a more translational question: how do genomic foundation models actually plug into drug discovery and biotech workflows?

Rather than walking step-by-step through a single therapeutic program, this chapter offers a compact, high-level map of where GFMs are already useful or plausibly soon will be. The focus is on three broad roles. First, target discovery and genetic validation use human genetics, variant-level scores, and gene-level evidence to prioritize safer, more effective targets. Second, functional genomics and perturbation screens leverage GFMs to design, interpret, and iteratively improve large-scale CRISPR, perturb-seq, and MPRA experiments. Third, biomarkers, patient stratification, and biotech infrastructure turn model outputs into actionable signals for trial design while integrating GFMs into industrial MLOps stacks.

Throughout, the aim is not to promise end-to-end AI drug discovery, but to show pragmatic ways that genomic foundation models can reduce risk, prioritize hypotheses, and make experiments more informative, especially when coupled to high-quality human data.

20.1. Where Genomics Touches the Drug Discovery Pipeline

The canonical small-molecule or biologics pipeline is often summarized as target identification and validation, followed by hit finding and lead optimization, preclinical characterization (covering safety, pharmacokinetics, and toxicology), and finally clinical trials through post-marketing surveillance. Genomics most directly enters at three points along this trajectory.

At the earliest stages, human genetic associations from GWAS, rare-variant burden analyses, and somatic mutation landscapes point to potential targets. Variant-level effect predictions and gene-level constraint metrics help de-prioritize potentially unsafe or non-causal signals, while fine-mapping approaches identify the specific variants most likely to drive observed associations.

Later in development, genetic risk scores, regulatory embeddings, and multi-omic signatures define patient subgroups and endpoints for trials. Embeddings from GFMs make it easier to find molecularly coherent patient strata beyond traditional clinical labels, enabling more precise cohort enrichment and response prediction.

Throughout the pipeline, functional genomics screens and perturbation assays help dissect how a compound perturbs cellular networks. GFMs can predict which perturbations matter most and suggest follow-up experiments that maximize information gain about mechanism and resistance pathways.

Other AI-for-drug-discovery efforts focus on molecular design, docking, or protein structure prediction; those applications are largely beyond the scope of this book. Here we stay close to the DNA- and RNA-centric capabilities developed in earlier chapters: variant effect prediction, regulatory modeling, and multi-omics integration.

20.2. Target Discovery and Genetic Validation

Human genetics provides some of the strongest evidence that modulating a particular target can safely change disease risk. GFMs do not replace classical statistical genetics, but they provide richer priors and more mechanistic features for identifying and validating targets.

20.2.1. From Variant-Level Scores to Gene-Level Targets

Variant effect prediction models provide a natural starting point for target discovery. Earlier chapters introduced genome-wide deleteriousness scores such as CADD, which integrate diverse annotations and, more recently, deep and foundation-model features (Rentzsch et al. 2019; Schubach et al. 2024). Protein-centric VEP GFMs including AlphaMissense, GPN-MSA, and AlphaGenome combine protein language models, structure, and long-range context to score coding variants (Cheng et al. 2023; Benegas, Albors, et al. 2024; Z. Avsec, Latysheva, and Cheng 2025; Brandes et al. 2023). Sequence-to-function models such as Enformer and long-context DNA language models (including Nucleic Transformer and HyenaDNA) predict regulatory outputs from large genomic windows (Ž. Avsec et al. 2021; He et al. 2023; Nguyen et al. 2023; Trop et al. 2024).

Drug target teams rarely care about individual variants per se; they care about genes and pathways. The key move is therefore to aggregate variant-level information into gene-level evidence. For coding variants, this means summarizing missense and predicted loss-of-function variants in each gene using

VEP scores, partitioning variants by predicted functional category (likely loss-of-function versus benign missense, for example) and by allele frequency, then deriving gene-level metrics such as burden of predicted damaging variants in cases versus controls.

For noncoding and regulatory variants, the aggregation problem is more complex. Teams can aggregate variant effect predictions on enhancers, promoters, and splice sites that link to candidate genes via chromatin interaction maps or models like Enformer (Ž. Avsec et al. 2021; He et al. 2023). Long-range GFMs connect distal regulatory elements to target loci across distances of 100 kilobases to 1 megabase, enabling attribution of noncoding signals to specific genes.

Constraint and intolerance metrics provide another dimension. Combining VEP-informed burden with gene constraint measures (as used implicitly in CADD and downstream tools) helps identify genes that are highly intolerant to damaging variation (Rentzsch et al. 2019; Schubach et al. 2024). Extremely constrained genes may be risky targets due to essentiality or toxicity concerns, while dose-sensitive but not lethal genes may present more attractive therapeutic opportunities.

From a GFM perspective, the core idea is to treat gene-level evidence as an aggregation problem over high-dimensional variant embeddings. Instead of manually defining a handful of summary statistics, teams can feed variant embeddings or predicted functional profiles into downstream models that learn which patterns matter most for disease.

20.2.2. Linking Genetic Evidence to Target Safety and Efficacy

Classical human genetics has established several now-standard heuristics for target selection. Human knockout individuals carrying biallelic loss-of-function variants provide natural experiments on what happens when a gene is effectively inactivated. Protective variants that reduce disease risk suggest directionality of effect, indicating that partial inhibition of a protein is beneficial rather than harmful. Pleiotropy, meaning associations with many unrelated traits, may signal safety liabilities.

GFMs reinforce and extend these ideas in several ways. Fine-mapping methods and multiple-instance models like MIFM can distinguish truly causal regulatory variants from correlated passengers (Wu et al. 2024; Rakowski and Lippert 2025). Combining these approaches with regulatory GFMs tightens the map from GWAS locus to variant to target gene. VEP scores from protein and regulatory GFMs can approximate effect sizes, estimating how severe a missense change is or how strongly a regulatory variant alters expression (Cheng et al. 2023; Benegas, Albors, et al. 2024; Z. Avsec, Latysheva, and Cheng 2025). This helps differentiate subtle modulators from catastrophic loss-of-function mutations. Finally, GFMs provide multi-task predictions across chromatin marks, transcription factor binding, expression, and splicing that make it easier to interpret how a risk locus affects biology (Ž. Avsec et al. 2021; Benegas, Ye, et al. 2024).

In practice, a target discovery workflow might proceed as follows. Starting from GWAS summary statistics or rare variant analyses, teams apply fine-mapping (such as MIFM) to identify candidate causal variants (Wu et al. 2024; Rakowski and Lippert 2025). They then score candidate variants with VEP GFMs for both protein and regulatory effects, map variants to genes using long-range regulatory models like Enformer, Nucleic Transformer, and HyenaDNA (Ž. Avsec et al. 2021; He et al. 2023; Nguyen et al. 2023), and aggregate signals into gene-level genetic support scores incorporating constraint and pleiotropy information. The result is a ranked list of candidate targets with structured evidence that can be compared across diseases and programs.

20.2.3. Evolving from Hand-Curated to Model-Centric Target Triage

Historically, target triage relied heavily on manual curation. Experts would review GWAS hits, literature, and pathway diagrams, but limited quantitative information was available for most genes, especially in non-classical pathways. GFMs shift this toward a model-centric, continuously updated view.

New data from biobank sequencing or single-cell atlases can be fed through trained GFMs to update variant and gene evidence. The same underlying model suite can support many disease programs, enabling consistent cross-portfolio comparisons. Benchmark frameworks like TraitGym emphasize standardized evaluation of genotype-phenotype modeling, helping teams choose appropriate model stacks for a given trait (Benegas, Eraslan, and Song 2025).

The limiting factor becomes less about whether an annotation exists and more about whether teams can interpret the model's representation and connect it to biological plausibility and druggability. This theme echoes discussions in **?@sec-vep** and Chapter 17 about the importance of interpretable predictions.

20.3. Functional Genomics Screens in Drug Discovery

While human genetics offers observational evidence, drug discovery also relies heavily on perturbation experiments: CRISPR knockout, knockdown, and activation screens; base-editing or saturation mutagenesis around key domains; MPRA and massively parallel promoter/enhancer assays; and perturb-seq and other high-throughput transcriptomic readouts. Genomic foundation models are well positioned to both design and interpret such screens.

20.3.1. Designing Smarter Perturbation Libraries

Traditional pooled screens often rely on simple design rules, such as one sgRNA per exon or tiling a region at fixed spacing. GFMs offer richer priors for library design. Variant effect scores from models like AlphaMissense or GPN-MSA can prioritize which amino acid positions are most likely to reveal functional differences when mutated (Cheng et al. 2023; Benegas, Albors, et al. 2024). Regulatory GFMs (Enformer, DeepSEA, Borzoi) can highlight which enhancer or promoter regions are predicted to have the largest expression effects in the cell type of interest (Ž. Avsec et al. 2021; J. Zhou and Troyanskaya 2015; Linder et al. 2025). Combinatorial designs can use model uncertainty to select perturbations that maximize expected information gain, focusing experimental budget on variants or regions where predictions are least confident.

This approach yields more informative libraries: instead of uniformly tiling a locus, teams can oversample positions that models flag as functionally important and undersample positions predicted to have negligible effects.

20.3.2. Interpreting Screen Results with GFM Features

After running a screen, GFMs help interpret which hits are most biologically meaningful. Embedding-based clustering can group perturbations with similar predicted functional profiles, even if their phenotypic readouts differ due to noise. Learned embeddings help propagate signal to weakly observed genes or variants, providing a form of regularization that improves detection of subtle effects.

20.3.3. Closing the Loop with Model Retraining

Perhaps the most powerful application is using screen outcomes as labeled examples to fine-tune sequence-to-function models in the relevant cell type or context. This lab-in-the-loop refinement turns generic GFMs into highly tuned models for the cell system of interest.

For example, an MPRA that assays thousands of enhancer variants yields sequence-activity pairs that can dramatically improve expression-prediction GFMs in that locus or tissue. Conversely, model predictions can suggest follow-up experiments (additional variants, cell types, or perturbation strengths) that would be maximally informative given previous data. This iterative cycle between computation and experiment accelerates discovery while improving model accuracy in disease-relevant regions of sequence space.

20.4. Biomarker Discovery, Patient Stratification, and Trial Design

Even when a target is well validated, many programs fail in late-stage trials because the right patients, endpoints, or biomarkers were not selected. GFMs, combined with large cohorts, offer new tools for defining and validating biomarkers.

20.4.1. From Polygenic Scores to GFM-Informed Biomarkers

Classical polygenic scores (PGS) summarize the additive effect of many common variants on disease risk. Deep learning methods such as Delphi extend this idea by learning non-linear genotype-phenotype mappings directly from genome-wide data (Georgantas, Kutalik, and Richiardi 2024).

GFMs can enhance these approaches in several ways. Instead of using raw genotypes as input, models can use VEP-derived scores, variant embeddings, or gene-level features produced by GFMs. This captures non-additive effects, regulatory architecture, and variant-level biology in a more compact representation. Foundation models trained across diverse genomes (such as Nucleotide Transformer, GENA-LM, and HyenaDNA) provide features that may generalize more robustly across populations than trait-specific models (Dalla-Torre et al. 2023; Fishman et al. 2025; Nguyen et al. 2023). Fine-mapping-aware approaches like MIFM further reduce dependence on linkage disequilibrium patterns that vary across ancestries (Wu et al. 2024; Rakowski and Lippert 2025).

By integrating regulatory and expression predictions, risk models can also distinguish genetic influences on disease onset versus progression, enabling more targeted enrichment strategies for different trial designs.

In trial design, such models can enrich for high-risk individuals in prevention trials, define genetic subtypes that may respond differently to the same mechanism, or construct composite biomarkers that mix genetics with conventional clinical features.

20.4.2. Multi-Omic and Single-Cell Biomarker Discovery

Beyond DNA variation, drug development increasingly leverages multi-omic and single-cell readouts. Whole-genome or exome tumor sequencing can be combined with expression, methylation, and copy-number profiling. Single-cell multiome datasets (RNA + ATAC) characterize cell-state landscapes in disease (Jurenaite et al. 2024; Yuan and Duren 2025). Microbiome sequencing provides insight into host-microbe interplay and response to therapy (Yan et al. 2025).

GFMs and related architectures help here in several ways. Set-based and graph-based encoders, such as SetQuence/SetOmic, treat heterogeneous genomic features for each tumor as a set, using deep set transformers to extract predictive representations (Jurenaite et al. 2024). Gene regulatory network inference models such as LINGER leverage atlas-scale multiome data to infer regulatory networks that can serve as biomarkers of pathway activity (Yuan and Duren 2025).

Multi-scale integration combines DNA and RNA GFMs with graph neural networks over gene and protein networks to build end-to-end predictors that map from genotype plus cell state to clinical endpoints (Gao et al. 2023; Benegas, Ye, et al. 2024). Embeddings from protein language models (such as ESM-2-based variant models) provide additional structure for coding variants (Brandes et al. 2023; Marquet et al. 2024).

A typical biomarker discovery workflow uses GFMs to generate rich embeddings for patients from tumor genomes, germline variation, or multi-omic profiles. Teams then cluster or perform supervised learning to identify molecular subgroups with differential prognosis or treatment response, validating candidate biomarkers on held-out cohorts or external datasets before deploying them in a trial.

The key shift is that biomarkers are no longer limited to a handful of hand-picked variants or expression markers: they become functions over high-dimensional genomic and multi-omic embeddings, learned in a data-driven way yet grounded in biological priors from GFMs.

20.5. Biotech Workflows and Infrastructure for GFMs

For pharma and biotech organizations, the primary challenge is not whether they can train a big model but how to integrate GFMs into existing data platforms, governance, and decision-making processes.

20.5.1. GFMs as Shared Infrastructure

In a mature organization, GFMs should be treated as shared infrastructure rather than ad hoc scripts developed by individual teams. A well-organized model catalog contains DNA language models (such as Nucleic Transformer, HyenaDNA, and GENA-LM), sequence-to-function models (such as Enformer and Genomic Interpreter), and variant effect predictors (AlphaMissense, GPN-MSA, AlphaGenome, CADD v1.7) (He et al. 2023; Nguyen et al. 2023; Fishman et al. 2025; Ž. Avsec et

al. 2021; Z. Li et al. 2023; Rentzsch et al. 2019; Schubach et al. 2024; Cheng et al. 2023; Benegas, Albors, et al. 2024; Z. Avsec, Latysheva, and Cheng 2025).

Feature services provide centralized APIs that take variants, genomic intervals, or genes as input and return embeddings, predicted functional profiles, or risk features. Logging and versioning ensure that analyses can be reproduced even as models and data evolve.

Data governance maintains clear separation between models trained on public data versus sensitive internal cohorts. Guardrails define where internal data can be used for fine-tuning and how resulting models can be shared.

Embedding GFMs in this way allows multiple teams across target identification, biomarker discovery, and clinical genetics to reuse the same core representations rather than each building bespoke models.

20.5.2. Build Versus Buy Versus Fine-Tune

Organizations face three strategic options when adopting GFMs. Using external GFMs as-is offers low up-front cost and benefits from community benchmarking (such as TraitGym for genotype-phenotype modeling), but may not capture organization-specific populations, assays, or traits (Benegas, Eraslan, and Song 2025).

Fine-tuning open-source GFMs on internal data retains powerful general representations while adapting to local data distributions. This approach requires careful privacy controls and computational investment, but often provides the best balance of generality and specificity.

Training bespoke internal GFMs offers maximum control and allows alignment of pretraining with available data and target use cases. However, this approach is expensive and requires complex MLOps, with risk of overfitting to narrow datasets if not complemented by broader pretraining.

In practice, many groups adopt a hybrid strategy. They start with public GFMs for early exploration and non-sensitive tasks, gradually fine-tune on internal biobank or trial data when added value is clear, and maintain lightweight model-serving infrastructure for latency-sensitive applications like clinical decision support alongside heavier offline systems for large-scale research workloads.

20.5.3. Intellectual Property, Collaboration, and Regulatory Considerations

GFMs also raise new questions around intellectual property, data sharing, and regulatory expectations. Models trained on proprietary data can be valuable IP assets but are difficult to patent directly. Downstream discoveries (targets, biomarkers) derived from GFMs must be carefully documented for freedom-to-operate analyses.

Joint training or evaluation across institutions may require federated learning or model-to-data paradigms, especially for patient-level data. For biomarkers used in pivotal trials, regulators will expect transparent documentation of model training, validation, and performance across subgroups. Chapter 16 and Chapter 17 highlight confounding and interpretability challenges that become even more acute when models inform trial inclusion or primary endpoints.

Overall, leveraging GFMs in biotech is as much an organizational and regulatory engineering problem as a technical one.

20.6. Forward Look: Toward Lab-in-the-Loop GFMs

A recurring theme across this book is moving from static models to closed loops that integrate foundational representation learning on large unlabeled datasets (genomes, multi-omics), task-specific supervision (disease status, expression, variant effects), and experimental feedback from perturbation assays, functional screens, and clinical trials.

In the drug discovery context, this suggests an evolution toward lab-in-the-loop GFMs. At the hypothesis generation stage, GFMs identify promising targets, variants, and regulatory regions. Graph and set-based models suggest network-level interventions (Jurenaite et al. 2024; Gao et al. 2023; Yuan and Duren 2025).

For experiment design, models propose perturbation libraries (CRISPR, MPRA) that maximize expected information gain. Safety and off-target predictions help filter risky designs before they reach the bench.

During evidence integration and model refinement, screen results feed back into GFMs, improving their local accuracy in disease-relevant regions of sequence space. Clinical trial outcomes update biomarker models and risk predictors for future trials.

Finally, portfolio-level decision support combines genetic and functional evidence from GFMs with classical pharmacology to prioritize or deprioritize programs. Uncertainty estimates and model critique (Chapter 17) help avoid over-confidence in purely model-driven recommendations.

Realizing this vision will require better calibration and uncertainty quantification in GFMs, stronger causal reasoning to distinguish correlation from intervention-worthiness, and careful ethical and equity considerations, especially when models influence who gets access to trials or targeted therapies (Chapter 16).

Yet even in the near term, GFMs already offer tangible value in de-risking targets, enriching cohorts, and interpreting complex functional data. When combined with rigorous experimental design and domain expertise, they can act not as oracle decision-makers, but as force multipliers for human scientists and clinicians.

20.7. Summary

This chapter has sketched how genomic foundation models extend beyond academic benchmarks into practical levers for drug discovery and biotech. GFMs turn variant and regulatory predictions into target discovery and validation pipelines, with workflows that aggregate variant-level scores into gene-level evidence and connect genetic signals to biological mechanisms. They enable the design and interpretation of functional genomics screens that probe mechanism and vulnerability, closing the loop between computational prediction and experimental validation. They support richer biomarkers and patient stratification schemes for trials, moving beyond individual variants to embeddings over high-dimensional genomic and multi-omic profiles. And they provide shared infrastructure for industrial data platforms and MLOps, raising new questions about build-versus-buy strategies, data governance, and regulatory documentation.

The previous chapters on clinical risk prediction (Chapter 18) and pathogenic variant discovery (Chapter 19) use the conceptual toolkit laid out here in more specialized contexts. Together, these

applications illustrate how the representational gains of genomic foundation models connect to the realities of translational research and patient care.

References

Adzhubei, Ivan A., Steffen Schmidt, Leonid Peshkin, Vasily E. Ramensky, Anna Gerasimova, Peer Bork, Alexey S. Kondrashov, and Shamil R. Sunyaev. 2010. "A Method and Server for Predicting Damaging Missense Mutations." *Nature Methods* 7 (4): 248–49. https://doi.org/10.1038/nmeth0410-248.

All of Us Research Program Investigators, All of Us; 2019. "The 'All of Us' Research Program." *New England Journal of Medicine* 381 (7): 668–76. https://doi.org/10.1056/NEJMsr1809937.

Amberger, Joanna S., Carol A. Bocchini, François Schiettecatte, Alan F. Scott, and Ada Hamosh. 2015. "OMIM.org: Online Mendelian Inheritance in Man (OMIM®), an Online Catalog of Human Genes and Genetic Disorders." *Nucleic Acids Research* 43 (D1): D789–98. https://doi.org/10.1093/nar/gku1205.

Auton, Adam, Gonçalo R. Abecasis, David M. Altshuler, Richard M. Durbin, Gonçalo R. Abecasis, David R. Bentley, Aravinda Chakravarti, et al. 2015. "A Global Reference for Human Genetic Variation." *Nature* 526 (7571): 68–74. https://doi.org/10.1038/nature15393.

Avsec, Žiga, Vikram Agarwal, D. Visentin, J. Ledsam, A. Grabska-Barwinska, Kyle R. Taylor, Yannis Assael, J. Jumper, Pushmeet Kohli, and David R. Kelley. 2021. "[Enformer] Effective Gene Expression Prediction from Sequence by Integrating Long-Range Interactions." *Nature Methods* 18 (October): 1196–1203. https://doi.org/10.1038/s41592-021-01252-x.

Avsec, Ziga, Natasha Latysheva, and Jun Cheng. 2025. "AlphaGenome: AI for Better Understanding the Genome." *Google DeepMind*. https://deepmind.google/discover/blog/alphagenome-ai-for-better-understanding-the-genome/.

Benegas, Gonzalo, Carlos Albors, Alan J. Aw, Chengzhong Ye, and Yun S. Song. 2024. "GPN-MSA: An Alignment-Based DNA Language Model for Genome-Wide Variant Effect Prediction." *bioRxiv*, April, 2023.10.10.561776. https://doi.org/10.1101/2023.10.10.561776.

Benegas, Gonzalo, Sanjit Singh Batra, and Yun S. Song. 2023. "[GPN] DNA Language Models Are Powerful Predictors of Genome-Wide Variant Effects." *Proceedings of the National Academy of Sciences* 120 (44): e2311219120. https://doi.org/10.1073/pnas.2311219120.

Benegas, Gonzalo, Gökcen Eraslan, and Yun S. Song. 2025. "[TraitGym] Benchmarking DNA Sequence Models for Causal Regulatory Variant Prediction in Human Genetics." bioRxiv. https://doi.org/10.1101/2025.02.11.637758.

Benegas, Gonzalo, Chengzhong Ye, Carlos Albors, Jianan Canal Li, and Yun S. Song. 2024. "Genomic Language Models: Opportunities and Challenges." arXiv. https://doi.org/10.48550/arXiv.2407.11435.

Bommasani, Rishi, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, et al. 2022. "On the Opportunities and Risks of Foundation Models." arXiv. https://doi.org/10.48550/arXiv.2108.07258.

Brandes, Nadav, Grant Goldman, Charlotte H. Wang, Chun Jimmie Ye, and Vasilis Ntranos. 2023. "Genome-Wide Prediction of Disease Variant Effects with a Deep Protein Language Model." *Nature Genetics* 55 (9): 1512–22. https://doi.org/10.1038/s41588-023-01465-0.

Brixi, Garyk, Matthew G. Durrant, Jerome Ku, Michael Poli, Greg Brockman, Daniel Chang, Gabriel A. Gonzalez, et al. 2025. "[Evo 2] Genome Modeling and Design Across All Domains of

Life with Evo 2." bioRxiv. https://doi.org/10.1101/2025.02.18.638918.

Browning, Brian L., Xiaowen Tian, Ying Zhou, and Sharon R. Browning. 2021. "Fast Two-Stage Phasing of Large-Scale Sequence Data." *American Journal of Human Genetics* 108 (10): 1880–90. https://doi.org/10.1016/j.ajhg.2021.08.005.

Bycroft, Clare, Colin Freeman, Desislava Petkova, Gavin Band, Lloyd T. Elliott, Kevin Sharp, Allan Motyer, et al. 2018. "The UK Biobank Resource with Deep Phenotyping and Genomic Data." *Nature* 562 (7726): 203–9. https://doi.org/10.1038/s41586-018-0579-z.

Camillo, Lucas Paulo de Lima, Raghav Sehgal, Jenel Armstrong, Albert T. Higgins-Chen, Steve Horvath, and Bo Wang. 2024. "CpGPT: A Foundation Model for DNA Methylation." bioRxiv. https://doi.org/10.1101/2024.10.24.619766.

Cao, Zhi-Jie, and Ge Gao. 2022. "[GLUE] Multi-Omics Single-Cell Data Integration and Regulatory Inference with Graph-Linked Embedding." *Nature Biotechnology* 40 (10): 1458–66. https://doi.org/10.1038/s41587-022-01284-4.

Chandak, Payal, Kexin Huang, and Marinka Zitnik. 2023. "[PrimeKG] Building a Knowledge Graph to Enable Precision Medicine." *Scientific Data* 10 (1): 67. https://doi.org/10.1038/s41597-023-01960-3.

Chen, Kathleen M., Aaron K. Wong, Olga G. Troyanskaya, and Jian Zhou. 2022. "[DeepSEA Sei] A Sequence-Based Global Map of Regulatory Activity for Deciphering Human Genetics." *Nature Genetics* 54 (7): 940–49. https://doi.org/10.1038/s41588-022-01102-2.

Cheng, Jun, Guido Novati, Joshua Pan, Clare Bycroft, Akvilė Žemgulytė, Taylor Applebaum, Alexander Pritzel, et al. 2023. "[AlphaMissense] Accurate Proteome-Wide Missense Variant Effect Prediction with AlphaMissense." *Science* 381 (6664): eadg7492. https://doi.org/10.1126/science.adg7492.

Choi, Shing Wan, Timothy Shin-Heng Mak, and Paul F. O'Reilly. 2020. "[PRS] Tutorial: A Guide to Performing Polygenic Risk Score Analyses." *Nature Protocols* 15 (9): 2759–72. https://doi.org/10.1038/s41596-020-0353-1.

Chung, Wen-Hung, Shuen-Iu Hung, Hong-Shang Hong, Mo-Song Hsih, Li-Cheng Yang, Hsin-Chun Ho, Jer-Yuarn Wu, and Yuan-Tsong Chen. 2004. "A Marker for Stevens–Johnson Syndrome." *Nature* 428 (6982): 486–86. https://doi.org/10.1038/428486a.

Clarke, Brian, Eva Holtkamp, Hakime Öztürk, Marcel Mück, Magnus Wahlberg, Kayla Meyer, Felix Munzlinger, et al. 2024. "[DeepRVAT] Integration of Variant Annotations Using Deep Set Networks Boosts Rare Variant Association Testing." *Nature Genetics* 56 (10): 2271–80. https://doi.org/10.1038/s41588-024-01919-z.

Dabernig-Heinz, Johanna, Mara Lohde, Martin Hölzer, Adriana Cabal, Rick Conzemius, Christian Brandt, Matthias Kohl, et al. 2024. "A Multicenter Study on Accuracy and Reproducibility of Nanopore Sequencing-Based Genotyping of Bacterial Pathogens." *Journal of Clinical Microbiology* 62 (9): e00628–24. https://doi.org/10.1128/jcm.00628-24.

Dalla-Torre, Hugo, Liam Gonzalez, Javier Mendoza-Revilla, Nicolas Lopez Carranza, Adam Henryk Grzywaczewski, Francesco Oteri, Christian Dallago, et al. 2023. "Nucleotide Transformer: Building and Evaluating Robust Foundation Models for Human Genomics." *Nature Methods* 22 (2): 287–97. https://doi.org/10.1038/s41592-024-02523-z.

Davydov, Eugene V., David L. Goode, Marina Sirota, Gregory M. Cooper, Arend Sidow, and Serafim Batzoglou. 2010. "Identifying a High Fraction of the Human Genome to Be Under Selective Constraint Using GERP++." *PLOS Computational Biology* 6 (12): e1001025. https://doi.org/10.1371/journal.pcbi.1001025.

DePristo, Mark A., Eric Banks, Ryan Poplin, Kiran V. Garimella, Jared R. Maguire, Christopher Hartl, Anthony A. Philippakis, et al. 2011. "A Framework for Variation Discovery and Genotyping Using Next-Generation DNA Sequencing Data." *Nature Genetics* 43 (5): 491–98.

https://doi.org/10.1038/ng.806.

Edgar, Ron, Michael Domrachev, and Alex E. Lash. 2002. "Gene Expression Omnibus: NCBI Gene Expression and Hybridization Array Data Repository." *Nucleic Acids Research* 30 (1): 207–10. https://doi.org/10.1093/nar/30.1.207.

Fishman, Veniamin, Yuri Kuratov, Aleksei Shmelev, Maxim Petrov, Dmitry Penzar, Denis Shepelin, Nikolay Chekanov, Olga Kardymon, and Mikhail Burtsev. 2025. "GENA-LM: A Family of Open-Source Foundational DNA Language Models for Long Sequences." *Nucleic Acids Research* 53 (2): gkae1310. https://doi.org/10.1093/nar/gkae1310.

Frankish, Adam, Mark Diekhans, Anne-Maud Ferreira, Rory Johnson, Irwin Jungreis, Jane Loveland, Jonathan M Mudge, et al. 2019. "GENCODE Reference Annotation for the Human and Mouse Genomes." *Nucleic Acids Research* 47 (D1): D766–73. https://doi.org/10.1093/nar/gky955.

Gamazon, Eric R., Heather E. Wheeler, Kaanan P. Shah, Sahar V. Mozaffari, Keston Aquino-Michaels, Robert J. Carroll, Anne E. Eyler, et al. 2015. "A Gene-Based Association Method for Mapping Traits Using Reference Transcriptome Data." *Nature Genetics* 47 (9): 1091–98. https://doi.org/10.1038/ng.3367.

Gao, Ziqi, Chenran Jiang, Jiawen Zhang, Xiaosen Jiang, Lanqing Li, Peilin Zhao, Huanming Yang, Yong Huang, and Jia Li. 2023. "[HIGH-PPI] Hierarchical Graph Learning for Protein–Protein Interaction." *Nature Communications* 14 (1): 1093. https://doi.org/10.1038/s41467-023-36736-1.

Garrison, Erik, Jouni Sirén, Adam M. Novak, Glenn Hickey, Jordan M. Eizenga, Eric T. Dawson, William Jones, et al. 2018. "Variation Graph Toolkit Improves Read Mapping by Representing Genetic Variation in the Reference." *Nature Biotechnology* 36 (9): 875–79. https://doi.org/10.1038/nbt.4227.

Georgantas, Costa, Zoltán Kutalik, and Jonas Richiardi. 2024. "Delphi: A Deep-Learning Method for Polygenic Risk Prediction." medRxiv. https://doi.org/10.1101/2024.04.19.24306079.

Goodwin, Sara, John D. McPherson, and W. Richard McCombie. 2016. "Coming of Age: Ten Years of Next-Generation Sequencing Technologies." *Nature Reviews Genetics* 17 (6): 333–51. https://doi.org/10.1038/nrg.2016.49.

Guo, Fei, Renchu Guan, Yaohang Li, Qi Liu, Xiaowo Wang, Can Yang, and Jianxin Wang. 2025. "Foundation Models in Bioinformatics." *National Science Review* 12 (4): nwaf028. https://doi.org/10.1093/nsr/nwaf028.

Gusev, Alexander, Arthur Ko, Huwenbo Shi, Gaurav Bhatia, Wonil Chung, Brenda W. J. H. Penninx, Rick Jansen, et al. 2016. "Integrative Approaches for Large-Scale Transcriptome-Wide Association Studies." *Nature Genetics* 48 (3): 245–52. https://doi.org/10.1038/ng.3506.

He, Shujun, Baizhen Gao, Rushant Sabnis, and Qing Sun. 2023. "Nucleic Transformer: Classifying DNA Sequences with Self-Attention and Convolutions." *ACS Synthetic Biology* 12 (11): 3205–14. https://doi.org/10.1021/acssynbio.3c00154.

Hudaiberdiev, Sanjarbek, D. Leland Taylor, Wei Song, Narisu Narisu, Redwan M. Bhuiyan, Henry J. Taylor, Xuming Tang, et al. 2023. "[TREDNet] Modeling Islet Enhancers Using Deep Learning Identifies Candidate Causal Variants at Loci Associated with T2D and Glycemic Traits." *Proceedings of the National Academy of Sciences* 120 (35): e2206612120. https://doi.org/10.1073/pnas.2206612120.

Jaganathan, Kishore, Sofia Kyriazopoulou Panagiotopoulou, Jeremy F. McRae, Siavash Fazel Darbandi, David Knowles, Yang I. Li, Jack A. Kosmicki, et al. 2019. "[SpliceAI] Predicting Splicing from Primary Sequence with Deep Learning." *Cell* 176 (3): 535–548.e24. https://doi.org/10.1016/j.cell.2018.12.015.

Ji, Yanrong, Zhihan Zhou, Han Liu, and Ramana V Davuluri. 2021. "DNABERT: Pre-Trained Bidirectional Encoder Representations from Transformers Model for DNA-Language in Genome." *Bioinformatics* 37 (15): 2112–20. https://doi.org/10.1093/bioinformatics/btab083.

Jiang, Tao, Yongzhuang Liu, Yue Jiang, Junyi Li, Yan Gao, Zhe Cui, Yadong Liu, Bo Liu, and Yadong Wang. 2020. "Long-Read-Based Human Genomic Structural Variation Detection with cuteSV." *Genome Biology* 21 (1): 189. https://doi.org/10.1186/s13059-020-02107-y.

Jurenaite, Neringa, Daniel León-Periñán, Veronika Donath, Sunna Torge, and René Jäkel. 2024. "SetQuence & SetOmic: Deep Set Transformers for Whole Genome and Exome Tumour Analysis." *BioSystems* 235 (January): 105095. https://doi.org/10.1016/j.biosystems.2023.105095.

Kagda, Meenakshi S., Bonita Lam, Casey Litton, Corinn Small, Cricket A. Sloan, Emma Spragins, Forrest Tanaka, et al. 2025. "Data Navigation on the ENCODE Portal." *Nature Communications* 16 (1): 9592. https://doi.org/10.1038/s41467-025-64343-9.

Karczewski, Konrad J., Laurent C. Francioli, Grace Tiao, Beryl B. Cummings, Jessica Alföldi, Qingbo Wang, Ryan L. Collins, et al. 2020. "The Mutational Constraint Spectrum Quantified from Variation in 141,456 Humans." *Nature* 581 (7809): 434–43. https://doi.org/10.1038/s41586-020-2308-7.

Krusche, Peter, Len Trigg, Paul C. Boutros, Christopher E. Mason, Francisco M. De La Vega, Benjamin L. Moore, Mar Gonzalez-Porta, et al. 2019. "Best Practices for Benchmarking Germline Small Variant Calls in Human Genomes." *Nature Biotechnology* 37 (5): 555–60. https://doi.org/10.1038/s41587-019-0054-x.

Kundaje, Anshul, Wouter Meuleman, Jason Ernst, Misha Bilenky, Angela Yen, Alireza Heravi-Moussavi, Pouya Kheradpour, et al. 2015. "Integrative Analysis of 111 Reference Human Epigenomes." *Nature* 518 (7539): 317–30. https://doi.org/10.1038/nature14248.

Kurki, Mitja I., Juha Karjalainen, Priit Palta, Timo P. Sipilä, Kati Kristiansson, Kati M. Donner, Mary P. Reeve, et al. 2023. "FinnGen Provides Genetic Insights from a Well-Phenotyped Isolated Population." *Nature* 613 (7944): 508–18. https://doi.org/10.1038/s41586-022-05473-8.

Lambert, Samuel A., Laurent Gil, Simon Jupp, Scott C. Ritchie, Yu Xu, Annalisa Buniello, Aoife McMahon, et al. 2021. "The Polygenic Score Catalog as an Open Database for Reproducibility and Systematic Evaluation." *Nature Genetics* 53 (4): 420–25. https://doi.org/10.1038/s41588-021-00783-5.

Landrum, Melissa J, Jennifer M Lee, Mark Benson, Garth R Brown, Chen Chao, Shanmuga Chitipiralla, Baoshan Gu, et al. 2018. "ClinVar: Improving Access to Variant Interpretations and Supporting Evidence." *Nucleic Acids Research* 46 (D1): D1062–67. https://doi.org/10.1093/nar/gkx1153.

Lee, Ingoo, Zachary S. Wallace, Yuqi Wang, Sungjoon Park, Hojung Nam, Amit R. Majithia, and Trey Ideker. 2025. "[G2PT] A Genotype-Phenotype Transformer to Assess and Explain Polygenic Risk." bioRxiv. https://doi.org/10.1101/2024.10.23.619940.

Li, Hao, Zebei Han, Yu Sun, Fu Wang, Pengzhen Hu, Yuang Gao, Xuemei Bai, et al. 2024. "CGMega: Explainable Graph Neural Network Framework with Attention Mechanisms for Cancer Gene Module Dissection." *Nature Communications* 15 (1): 5997. https://doi.org/10.1038/s41467-024-50426-6.

Li, Heng. 2013. "Aligning Sequence Reads, Clone Sequences and Assembly Contigs with BWA-MEM." arXiv. https://doi.org/10.48550/arXiv.1303.3997.

———. 2014. "Towards Better Understanding of Artifacts in Variant Calling from High-Coverage Samples." *Bioinformatics* 30 (20): 2843–51. https://doi.org/10.1093/bioinformatics/btu356.

———. 2018. "Minimap2: Pairwise Alignment for Nucleotide Sequences." *Bioinformatics* 34 (18): 3094–3100. https://doi.org/10.1093/bioinformatics/bty191.

Li, Xiao, Jie Ma, Ling Leng, Mingfei Han, Mansheng Li, Fuchu He, and Yunping Zhu. 2022. "MoGCN: A Multi-Omics Integration Method Based on Graph Convolutional Network for Cancer Subtype Analysis." *Frontiers in Genetics* 13 (February). https://doi.org/10.3389/fgene.2022.806842.

Li, Zehui, Akashaditya Das, William A. V. Beardall, Yiren Zhao, and Guy-Bart Stan. 2023. "Genomic Interpreter: A Hierarchical Genomic Deep Neural Network with 1D Shifted Window Transformer." arXiv. https://doi.org/10.48550/arXiv.2306.05143.

Liao, Wen-Wei, Mobin Asri, Jana Ebler, Daniel Doerr, Marina Haukness, Glenn Hickey, Shuangjia Lu, et al. 2023. "A Draft Human Pangenome Reference." *Nature* 617 (7960): 312–24. https://doi.org/10.1038/s41586-023-05896-x.

Lin, Zeming, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Allan dos Santos Costa, et al. 2022. "[ESM-2] Language Models of Protein Sequences at the Scale of Evolution Enable Accurate Structure Prediction." bioRxiv. https://doi.org/10.1101/2022.07.20.500902.

Linder, Johannes, Divyanshi Srivastava, Han Yuan, Vikram Agarwal, and David R. Kelley. 2025. "[Borzoi] Predicting RNA-Seq Coverage from DNA Sequence as a Unifying Model of Gene Regulation." *Nature Genetics* 57 (4): 949–61. https://doi.org/10.1038/s41588-024-02053-6.

Liu, Zicheng, Siyuan Li, Zhiyuan Chen, Fang Wu, Chang Yu, Qirong Yang, Yucheng Guo, Yujie Yang, Xiaoming Zhang, and Stan Z. Li. 2025. "Life-Code: Central Dogma Modeling with Multi-Omics Sequence Unification." arXiv. https://doi.org/10.48550/arXiv.2502.07299.

Loh, Po-Ru, Petr Danecek, Pier Francesco Palamara, Christian Fuchsberger, Yakir A Reshef, Hilary K Finucane, Sebastian Schoenherr, et al. 2016. "Reference-Based Phasing Using the Haplotype Reference Consortium Panel." *Nature Genetics* 48 (11): 1443–48. https://doi.org/10.1038/ng.3679.

Mallal, Simon, Elizabeth Phillips, Giampiero Carosi, Jean-Michel Molina, Cassy Workman, Janez Tomažič, Eva Jägel-Guedes, et al. 2008. "HLA-B*5701 Screening for Hypersensitivity to Abacavir." *New England Journal of Medicine* 358 (6): 568–79. https://doi.org/10.1056/NEJMoa0706135.

Manzo, Gaetano, Kathryn Borkowski, and Ivan Ovcharenko. 2025. "Comparative Analysis of Deep Learning Models for Predicting Causative Regulatory Variants." *bioRxiv: The Preprint Server for Biology*, June, 2025.05.19.654920. https://doi.org/10.1101/2025.05.19.654920.

Marees, Andries T., Hilde de Kluiver, Sven Stringer, Florence Vorspan, Emmanuel Curis, Cynthia Marie-Claire, and Eske M. Derks. 2018. "[GWAS] A Tutorial on Conducting Genome-Wide Association Studies: Quality Control and Statistical Analysis." *International Journal of Methods in Psychiatric Research* 27 (2): e1608. https://doi.org/10.1002/mpr.1608.

Marquet, Céline, Julius Schlensok, Marina Abakarova, Burkhard Rost, and Elodie Laine. 2024. "[VespaG] Expert-Guided Protein Language Models Enable Accurate and Blazingly Fast Fitness Prediction." *Bioinformatics* 40 (11): btae621. https://doi.org/10.1093/bioinformatics/btae621.

McKenna, Aaron, Matthew Hanna, Eric Banks, Andrey Sivachenko, Kristian Cibulskis, Andrew Kernytsky, Kiran Garimella, et al. 2010. "The Genome Analysis Toolkit: A MapReduce Framework for Analyzing Next-Generation DNA Sequencing Data." *Genome Research* 20 (9): 1297–1303. https://doi.org/10.1101/gr.107524.110.

Medvedev, Aleksandr, Karthik Viswanathan, Praveenkumar Kanithi, Kirill Vishniakov, Prateek Munjal, Clément Christophe, Marco AF Pimentel, Ronnie Rajan, and Shadab Khan. 2025. "BioToken and BioFM – Biologically-Informed Tokenization Enables Accurate and Efficient Genomic Foundation Models." bioRxiv. https://doi.org/10.1101/2025.03.27.645711.

Meier, Joshua, Roshan Rao, Robert Verkuil, Jason Liu, Tom Sercu, and Alexander Rives. 2021. "[ESM-1v] Language Models Enable Zero-Shot Prediction of the Effects of Mutations on Protein Function." bioRxiv. https://doi.org/10.1101/2021.07.09.450648.

Morales, Joannella, Shashikant Pujar, Jane E. Loveland, Alex Astashyn, Ruth Bennett, Andrew Berry, Eric Cox, et al. 2022. "A Joint NCBI and EMBL-EBI Transcript Set for Clinical Genomics and Research." *Nature* 604 (7905): 310–15. https://doi.org/10.1038/s41586-022-04558-8.

Mountjoy, Edward, Ellen M. Schmidt, Miguel Carmona, Jeremy Schwartzentruber, Gareth Peat,

Alfredo Miranda, Luca Fumis, et al. 2021. "An Open Approach to Systematically Prioritize Causal Variants and Genes at All Published Human GWAS Trait-Associated Loci." *Nature Genetics* 53 (11): 1527–33. https://doi.org/10.1038/s41588-021-00945-5.

Naghipourfar, Mohsen, Siyu Chen, Mathew K. Howard, Christian B. Macdonald, Ali Saberi, Timo Hagen, Mohammad R. K. Mofrad, Willow Coyote-Maestas, and Hani Goodarzi. 2024. "[cdsFM - EnCodon/DeCodon] A Suite of Foundation Models Captures the Contextual Interplay Between Codons." bioRxiv. https://doi.org/10.1101/2024.10.10.617568.

Ng, Pauline C., and Steven Henikoff. 2003. "SIFT: Predicting Amino Acid Changes That Affect Protein Function." *Nucleic Acids Research* 31 (13): 3812–14. https://doi.org/10.1093/nar/gkg509.

Nguyen, Eric, Michael Poli, Marjan Faizi, Armin Thomas, Callum Birch-Sykes, Michael Wornow, Aman Patel, et al. 2023. "HyenaDNA: Long-Range Genomic Sequence Modeling at Single Nucleotide Resolution." arXiv. https://doi.org/10.48550/arXiv.2306.15794.

Nielsen, Rasmus, Joshua S. Paul, Anders Albrechtsen, and Yun S. Song. 2011. "Genotype and SNP Calling from Next-Generation Sequencing Data." *Nature Reviews. Genetics* 12 (6): 443–51. https://doi.org/10.1038/nrg2986.

Notin, Pascal, Aaron Kollasch, Daniel Ritter, Lood van Niekerk, Steffanie Paul, Han Spinner, Nathan Rollins, et al. 2023. "ProteinGym: Large-Scale Benchmarks for Protein Fitness Prediction and Design." *Advances in Neural Information Processing Systems* 36 (December): 64331–79. https://papers.nips.cc/paper_files/paper/2023/hash/cac723e5ff29f65e3fcbb0739ae91bee-Abstract-Datasets_and_Benchmarks.html.

Nurk, Sergey, Sergey Koren, Arang Rhie, Mikko Rautiainen, Andrey V. Bzikadze, Alla Mikheenko, Mitchell R. Vollger, et al. 2022. "The Complete Sequence of a Human Genome." *Science* 376 (6588): 44–53. https://doi.org/10.1126/science.abj6987.

O'Connell, Jared, Deepti Gurdasani, Olivier Delaneau, Nicola Pirastu, Sheila Ulivi, Massimiliano Cocca, Michela Traglia, et al. 2014. "A General Approach for Haplotype Phasing Across the Full Spectrum of Relatedness." *PLOS Genetics* 10 (4): e1004234. https://doi.org/10.1371/journal.pgen.1004234.

O'Leary, Nuala A., Mathew W. Wright, J. Rodney Brister, Stacy Ciufo, Diana Haddad, Rich McVeigh, Bhanu Rajput, et al. 2016. "Reference Sequence (RefSeq) Database at NCBI: Current Status, Taxonomic Expansion, and Functional Annotation." *Nucleic Acids Research* 44 (D1): D733–45. https://doi.org/10.1093/nar/gkv1189.

"PacificBiosciences/Pbsv." 2025. PacBio. https://github.com/PacificBiosciences/pbsv.

Padyukov, Leonid. 2022. "Genetics of Rheumatoid Arthritis." *Seminars in Immunopathology* 44 (1): 47–62. https://doi.org/10.1007/s00281-022-00912-0.

Pasaniuc, Bogdan, and Alkes L. Price. 2016. "Dissecting the Genetics of Complex Traits Using Summary Association Statistics." *Nature Reviews Genetics* 18 (2): 117–27. https://doi.org/10.1038/nrg.2016.142.

Patterson, Nick, Alkes L. Price, and David Reich. 2006. "Population Structure and Eigenanalysis." *PLOS Genetics* 2 (12): e190. https://doi.org/10.1371/journal.pgen.0020190.

Pejaver, Vikas, Alicia B. Byrne, Bing-Jian Feng, Kymberleigh A. Pagel, Sean D. Mooney, Rachel Karchin, Anne O'Donnell-Luria, et al. 2022. "Calibration of Computational Tools for Missense Variant Pathogenicity Classification and ClinGen Recommendations for PP3/BP4 Criteria." *American Journal of Human Genetics* 109 (12): 2163–77. https://doi.org/10.1016/j.ajhg.2022.10.013.

Poplin, Ryan, Pi-Chuan Chang, David Alexander, Scott Schwartz, Thomas Colthurst, Alexander Ku, Dan Newburger, et al. 2018. "[DeepVariant] A Universal SNP and Small-Indel Variant Caller Using Deep Neural Networks." *Nature Biotechnology* 36 (10): 983–87. https://doi.org/10.

1038/nbt.4235.

Rakowski, Alexander, and Christoph Lippert. 2025. "[MIFM] Multiple Instance Fine-Mapping: Predicting Causal Regulatory Variants with a Deep Sequence Model." medRxiv. https://doi.org/10.1101/2025.06.13.25329551.

"RealTimeGenomics/Rtg-Core." 2025. Real Time Genomics. https://github.com/RealTimeGenomics/rtg-core.

Regev, Aviv, Sarah A Teichmann, Eric S Lander, Ido Amit, Christophe Benoist, Ewan Birney, Bernd Bodenmiller, et al. 2017. "The Human Cell Atlas." Edited by Thomas R Gingeras. *eLife* 6 (December): e27041. https://doi.org/10.7554/eLife.27041.

Rehm, Heidi L., Jonathan S. Berg, Lisa D. Brooks, Carlos D. Bustamante, James P. Evans, Melissa J. Landrum, David H. Ledbetter, et al. 2015. "ClinGen — The Clinical Genome Resource." *New England Journal of Medicine* 372 (23): 2235–42. https://doi.org/10.1056/NEJMsr1406261.

Rentzsch, Philipp, Daniela Witten, Gregory M Cooper, Jay Shendure, and Martin Kircher. 2019. "CADD: Predicting the Deleteriousness of Variants Throughout the Human Genome." *Nucleic Acids Research* 47 (D1): D886–94. https://doi.org/10.1093/nar/gky1016.

Rives, Alexander, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, et al. 2021. "[ESM-1b] Biological Structure and Function Emerge from Scaling Unsupervised Learning to 250 Million Protein Sequences." *Proceedings of the National Academy of Sciences of the United States of America* 118 (15): e2016239118. https://doi.org/10.1073/pnas.2016239118.

Robinson, James, Dominic J Barker, Xenia Georgiou, Michael A Cooper, Paul Flicek, and Steven G E Marsh. 2020. "IPD-IMGT/HLA Database." *Nucleic Acids Research* 48 (D1): D948–55. https://doi.org/10.1093/nar/gkz950.

Sakaue, Saori, Saisriram Gurajala, Michelle Curtis, Yang Luo, Wanson Choi, Kazuyoshi Ishigaki, Joyce B. Kang, et al. 2023. "Tutorial: A Statistical Genetics Guide to Identifying HLA Alleles Driving Complex Disease." *Nature Protocols* 18 (9): 2625–41. https://doi.org/10.1038/s41596-023-00853-4.

Sanabria, Melissa, Jonas Hirsch, Pierre M. Joubert, and Anna R. Poetsch. 2024. "[GROVER] DNA Language Model GROVER Learns Sequence Context in the Human Genome." *Nature Machine Intelligence* 6 (8): 911–23. https://doi.org/10.1038/s42256-024-00872-0.

Schiff, Yair, Chia-Hsiang Kao, Aaron Gokaslan, Tri Dao, Albert Gu, and Volodymyr Kuleshov. 2024. "Caduceus: Bi-Directional Equivariant Long-Range DNA Sequence Modeling." arXiv. https://doi.org/10.48550/arXiv.2403.03234.

Schubach, Max, Thorben Maass, Lusiné Nazaretyan, Sebastian Röner, and Martin Kircher. 2024. "CADD V1.7: Using Protein Language Models, Regulatory CNNs and Other Nucleotide-Level Scores to Improve Genome-Wide Variant Predictions." *Nucleic Acids Research* 52 (D1): D1143–54. https://doi.org/10.1093/nar/gkad989.

Shafin, Kishwar, Trevor Pesout, Pi-Chuan Chang, Maria Nattestad, Alexey Kolesnikov, Sidharth Goel, Gunjan Baid, et al. 2021. "Haplotype-Aware Variant Calling with PEPPER-Margin-DeepVariant Enables High Accuracy in Nanopore Long-Reads." *Nature Methods* 18 (11): 1322–32. https://doi.org/10.1038/s41592-021-01299-w.

Sherry, S. T., M.-H. Ward, M. Kholodov, J. Baker, L. Phan, E. M. Smigielski, and K. Sirotkin. 2001. "dbSNP: The NCBI Database of Genetic Variation." *Nucleic Acids Research* 29 (1): 308–11. https://doi.org/10.1093/nar/29.1.308.

Siepel, Adam, Gill Bejerano, Jakob S. Pedersen, Angie S. Hinrichs, Minmei Hou, Kate Rosenbloom, Hiram Clawson, et al. 2005. "[PhastCons] Evolutionarily Conserved Elements in Vertebrate, Insect, Worm, and Yeast Genomes." *Genome Research* 15 (8): 1034–50. https://doi.org/10.1101/gr.3715005.

Sirugo, Giorgio, Scott M. Williams, and Sarah A. Tishkoff. 2019. "The Missing Diversity in Human

Genetic Studies." *Cell* 177 (1): 26–31. https://doi.org/10.1016/j.cell.2019.02.048.

Smolka, Moritz, Luis F. Paulin, Christopher M. Grochowski, Dominic W. Horner, Medhat Mahmoud, Sairam Behera, Ester Kalef-Ezra, et al. 2024. "Detection of Mosaic and Population-Level Structural Variants with Sniffles2." *Nature Biotechnology* 42 (10): 1571–80. https://doi.org/10.1038/s41587-023-02024-y.

Sollis, Elliot, Abayomi Mosaku, Ala Abid, Annalisa Buniello, Maria Cerezo, Laurent Gil, Tudor Groza, et al. 2023. "The NHGRI-EBI GWAS Catalog: Knowledgebase and Deposition Resource." *Nucleic Acids Research* 51 (D1): D977–85. https://doi.org/10.1093/nar/gkac1010.

Song, Li, Gali Bai, X. Shirley Liu, Bo Li, and Heng Li. 2022. "T1K: Efficient and Accurate KIR and HLA Genotyping with Next-Generation Sequencing Data." bioRxiv. https://doi.org/10.1101/2022.10.26.513955.

"The Genome Aggregation Database (gnomAD)." n.d. Accessed July 3, 2025. https://www.nature.com/immersive/d42859-020-00002-x/index.html.

The GTEx Consortium. 2020. "The GTEx Consortium Atlas of Genetic Regulatory Effects Across Human Tissues." *Science* 369 (6509): 1318–30. https://doi.org/10.1126/science.aaz1776.

The Tabula Sapiens Consortium. 2022. "The Tabula Sapiens: A Multiple-Organ, Single-Cell Transcriptomic Atlas of Humans." *Science* 376 (6594): eabl4896. https://doi.org/10.1126/science.abl4896.

Trop, Evan, Yair Schiff, Edgar Mariano Marroquin, Chia Hsiang Kao, Aaron Gokaslan, McKinley Polen, Mingyi Shao, et al. 2024. "The Genomics Long-Range Benchmark: Advancing DNA Language Models," October. https://openreview.net/forum?id=8O9HLDrmtq.

Van der Auwera, Geraldine A., Mauricio O. Carneiro, Christopher Hartl, Ryan Poplin, Guillermo del Angel, Ami Levy-Moonshine, Tadeusz Jordan, et al. 2018. "From FastQ Data to High-Confidence Variant Calls: The Genome Analysis Toolkit Best Practices Pipeline." *Current Protocols in Bioinformatics* 43 (1): 11.10.1–33. https://doi.org/10.1002/0471250953.bi1110s43.

Verma, Anurag, Jennifer E. Huffman, Alex Rodriguez, Mitchell Conery, Molei Liu, Yuk-Lam Ho, Youngdae Kim, et al. 2024. "Diversity and Scale: Genetic Architecture of 2068 Traits in the VA Million Veteran Program." *Science* 385 (6706): eadj1182. https://doi.org/10.1126/science.adj1182.

Vilhjálmsson, Bjarni J., Jian Yang, Hilary K. Finucane, Alexander Gusev, Sara Lindström, Stephan Ripke, Giulio Genovese, et al. 2015. "Modeling Linkage Disequilibrium Increases Accuracy of Polygenic Risk Scores." *American Journal of Human Genetics* 97 (4): 576–92. https://doi.org/10.1016/j.ajhg.2015.09.001.

Võsa, Urmo, Annique Claringbould, Harm-Jan Westra, Marc Jan Bonder, Patrick Deelen, Biao Zeng, Holger Kirsten, et al. 2021. "Large-Scale Cis- and Trans-eQTL Analyses Identify Thousands of Genetic Loci and Polygenic Scores That Regulate Blood Gene Expression." *Nature Genetics* 53 (9): 1300–1310. https://doi.org/10.1038/s41588-021-00913-z.

Wenger, Aaron M., Paul Peluso, William J. Rowell, Pi-Chuan Chang, Richard J. Hall, Gregory T. Concepcion, Jana Ebler, et al. 2019. "Accurate Circular Consensus Long-Read Sequencing Improves Variant Detection and Assembly of a Human Genome." *Nature Biotechnology* 37 (10): 1155–62. https://doi.org/10.1038/s41587-019-0217-9.

Whirl-Carrillo, M, E M McDonagh, J M Hebert, L Gong, K Sangkuhl, C F Thorn, R B Altman, and T E Klein. 2012. "Pharmacogenomics Knowledge for Personalized Medicine." *Clinical Pharmacology & Therapeutics* 92 (4): 414–17. https://doi.org/10.1038/clpt.2012.96.

Wu, Yang, Zhili Zheng, Loic Thibaut2, Michael E. Goddard, Naomi R. Wray, Peter M. Visscher, and Jian Zeng. 2024. "Genome-Wide Fine-Mapping Improves Identification of Causal Variants." *Research Square*, August, rs.3.rs–4759390. https://doi.org/10.21203/rs.3.rs-4759390/v1.

Yan, Binghao, Yunbi Nam, Lingyao Li, Rebecca A. Deek, Hongzhe Li, and Siyuan Ma. 2025. "Recent

Advances in Deep Learning and Language Models for Studying the Microbiome." *Frontiers in Genetics* 15 (January). https://doi.org/10.3389/fgene.2024.1494474.

Yuan, Qiuyue, and Zhana Duren. 2025. "[LINGER] Inferring Gene Regulatory Networks from Single-Cell Multiome Data Using Atlas-Scale External Data." *Nature Biotechnology* 43 (2): 247–57. https://doi.org/10.1038/s41587-024-02182-7.

Yun, Taedong, Helen Li, Pi-Chuan Chang, Michael F Lin, Andrew Carroll, and Cory Y McLean. 2021. "Accurate, Scalable Cohort Variant Calls Using DeepVariant and GLnexus." *Bioinformatics* 36 (24): 5582–89. https://doi.org/10.1093/bioinformatics/btaa1081.

Zheng, Rongbin, Changxin Wan, Shenglin Mei, Qian Qin, Qiu Wu, Hanfei Sun, Chen-Hao Chen, et al. 2019. "Cistrome Data Browser: Expanded Datasets and New Tools for Gene Regulatory Analysis." *Nucleic Acids Research* 47 (D1): D729–35. https://doi.org/10.1093/nar/gky1094.

Zheng, Zhenxian, Shumin Li, Junhao Su, Amy Wing-Sze Leung, Tak-Wah Lam, and Ruibang Luo. 2022. "Symphonizing Pileup and Full-Alignment for Deep Learning-Based Long-Read Variant Calling." *Nature Computational Science* 2 (12): 797–803. https://doi.org/10.1038/s43588-022-00387-x.

Zhou, Jian, Chandra L. Theesfeld, Kevin Yao, Kathleen M. Chen, Aaron K. Wong, and Olga G. Troyanskaya. 2018. "[Expecto] Deep Learning Sequence-Based Ab Initio Prediction of Variant Effects on Expression and Disease Risk." *Nature Genetics* 50 (8): 1171–79. https://doi.org/10.1038/s41588-018-0160-6.

Zhou, Jian, and Olga G. Troyanskaya. 2015. "[DeepSEA] Predicting Effects of Noncoding Variants with Deep Learning–Based Sequence Model." *Nature Methods* 12 (10): 931–34. https://doi.org/10.1038/nmeth.3547.

Zhou, Zhihan, Yanrong Ji, Weijian Li, Pratik Dutta, Ramana Davuluri, and Han Liu. 2024. "DNABERT-2: Efficient Foundation Model and Benchmark For Multi-Species Genome." arXiv. https://doi.org/10.48550/arXiv.2306.15006.

Zook, Justin M., Jennifer McDaniel, Nathan D. Olson, Justin Wagner, Hemang Parikh, Haynes Heaton, Sean A. Irvine, et al. 2019. "An Open Resource for Accurately Benchmarking Small Variant and Reference Calls." *Nature Biotechnology* 37 (5): 561–66. https://doi.org/10.1038/s41587-019-0074-6.

A. Deep Learning Primer for Genomics

> ⚠ Warning
>
> **TODO:**
>
> - ...
> - ...

This appendix gives a compact introduction to deep learning for readers who are comfortable with genomics but less familiar with modern neural networks. The goal is not to replace a full machine learning textbook, but to provide enough background to make the models in Chapters 5–19 feel intuitive rather than magical.

We focus on:

- How deep models are structured (layers, parameters, activations)
- How they are trained (loss functions, gradients, optimization)
- Core architectures that appear throughout the book (CNNs, Transformers)
- Concepts like self-supervised pretraining and transfer learning

Where possible, we connect directly to the genomic case studies in the main text (DeepSEA, ExPecto, SpliceAI, Enformer, genomic language models, and GFMs).

A.1. From Linear Models to Deep Networks

A.1.1. Models as Functions

At its core, a predictive model is just a function:

$$f_\theta : x \mapsto \hat{y} \tag{A.1}$$

where:

- x is an input (e.g., a one-hot encoded DNA sequence, variant-level features, or a patient feature vector).
- $\hat{y}$ is a prediction (e.g., probability of a histone mark, gene expression level, disease risk).
- θ are the parameters (weights) of the model.

In classical genomics workflows, f_θ might be:

- **Logistic regression** (for case–control status)
- **Linear regression** (for quantitative traits)
- **Random forests** or **gradient boosting** (for variant pathogenicity scores)

Deep learning keeps the same basic structure but allows f_θ to be a much more flexible, high-capacity function built by composing many simple operations.

A.1.2. Linear Models vs Neural Networks

A simple linear model for classification looks like:

$$\hat{y} = \sigma(w^\top x + b),$$

where w and b are parameters and $\sigma(\cdot)$ is a squashing nonlinearity (e.g., the logistic function). The model draws a single separating hyperplane in feature space.

A **neural network** generalizes this by stacking multiple linear transformations with nonlinear activation functions:

$$\begin{aligned} h_1 &= \phi(W_1 x + b_1) \\ h_2 &= \phi(W_2 h_1 + b_2) \\ &\vdots \\ \hat{y} &= g(W_L h_{L-1} + b_L) \end{aligned}$$

where:

- Each W_ℓ, b_ℓ is a layer's weight matrix and bias.
- $\phi(\cdot)$ is a nonlinear activation (e.g., ReLU).
- $g(\cdot)$ is a final activation (e.g., sigmoid for probabilities, identity for regression).

The key idea:

> By composing many simple nonlinear transformations, deep networks can approximate very complex functions.

In Chapters 5–7, DeepSEA, ExPecto, and SpliceAI implement exactly this pattern, but with **convolutional** layers (Section 4) tailored to 1D DNA sequence instead of dense matrix multiplications (J. Zhou and Troyanskaya 2015; J. Zhou et al. 2018; Jaganathan et al. 2019).

A.2. Training Deep Models

A.2.1. Data, Labels, and Loss Functions

To train a model, we need:

- A dataset of examples $\{(x_i, y_i)\}_{i=1}^N$
- A model f_θ
- A **loss function** $L(\hat{y}, y)$ that measures how wrong a prediction is

Common loss functions:

- **Binary cross-entropy** (for yes/no labels, e.g., "is this ChIP–seq peak present?"):

$$L(\hat{p}, y) = -(y \log \hat{p} + (1 - y) \log(1 - \hat{p}))$$

- **Multiclass cross-entropy** (for one-of-K labels)
- **Mean squared error (MSE)** (for continuous outputs, e.g., gene expression)

The **training objective** is to find θ that minimizes the average loss:

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^{N} L(f_\theta(x_i), y_i).$$

A.2.2. 2.2 Gradient-Based Optimization

Deep networks may have millions to billions of parameters. We can't search over all possibilities, but we can follow the gradient of the loss with respect to θ:

- **Gradient descent** updates:

$$\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}(\theta),$$

  where η is the learning rate.

In practice, we use:

- **Mini-batch stochastic gradient descent (SGD)**: Compute gradients on small batches of examples (e.g., 128 sequences at a time) for efficiency and better generalization.
- **Adaptive optimizers** like Adam, which adjust learning rates per parameter.

You never compute gradients by hand; modern frameworks (PyTorch, JAX, TensorFlow) use **automatic differentiation** to efficiently compute $\nabla_\theta \mathcal{L}$ even for very complex architectures.

A.2.3. Backpropagation in One Sentence

Backpropagation is just the chain rule of calculus applied efficiently through the layers of a network. It propagates "blame" from the output back to each weight, telling us how changing that weight would change the loss.

A.3. Generalization, Overfitting, and Evaluation

A.3.1. Train / Validation / Test Splits

Deep networks can memorize training data if we're not careful. To evaluate generalization, we typically split data into:

- **Training set** – used to fit parameters
- **Validation set** – used to tune hyperparameters (learning rate, depth, etc.) and perform early stopping
- **Test set** – held out until the end to estimate performance on new data

In genomics, **how we split** matters as much as **how much data** we have:

- Splitting by **locus or chromosome** (to test cross-locus generalization)
- Splitting by **individual or cohort** (to avoid leakage between related samples)
- Splitting by **species or ancestry** when evaluating transfer

These issues are developed in more depth in the evaluation and confounding chapters (Chapter 15 and Chapter 16).

A.3.2. Overfitting and Regularization

Signs of overfitting:

- Training loss keeps decreasing, but validation loss starts increasing.
- Metrics like AUROC or AUPRC plateau or drop on validation data even as they improve on training data.

Common regularization techniques:

- **Weight decay / L2 regularization** – penalize large weights.
- **Dropout** – randomly zero out activations during training.
- **Early stopping** – stop training when validation performance stops improving.
- **Data augmentation** – generate more training examples by transforming inputs, e.g.:
  - Reverse-complement augmentation for DNA sequences (treat sequence and its reverse complement as equivalent).
  - Window jittering: randomly shifting the sequence window around a target site.

A.3.3. Basic Metrics

You'll encounter metrics such as:

- **AUROC (Area Under the ROC Curve)** – how well the model ranks positives above negatives.
- **AUPRC (Area Under the Precision–Recall Curve)** – more informative when positives are rare.
- **Calibration metrics** (e.g., Brier score) and reliability diagrams – especially for clinical risk prediction (Chapter 18).

The model and application chapters provide details about which metrics are appropriate for which tasks. See Chapter 15 for more on evaluation metrics.

A.4. Convolutional Networks for Genomic Sequences

Convolutional neural networks (CNNs) are the workhorse architecture in early genomic deep learning models like DeepSEA, ExPecto, and SpliceAI (J. Zhou and Troyanskaya 2015; J. Zhou et al. 2018; Jaganathan et al. 2019).

A.4.1. 1D Convolutions as Motif Detectors

For a 1D DNA sequence encoded as a matrix $X \in \mathbb{R}^{L \times 4}$ (length L, 4 nucleotides), a **convolutional layer** applies a set of filters (kernels) of width k:

- Each filter is a small matrix $K \in \mathbb{R}^{k \times 4}$.

- At each position, the filter computes a dot product between K and the corresponding k-length chunk of X.
- Sliding the filter along the sequence creates an activation map that is high wherever the motif encoded by K is present.

Intuitively:

> A 1D convolutional filter learns to recognize sequence motifs (e.g., transcription factor binding sites) directly from data.

A.4.2. Stacking Layers and Receptive Fields

Deeper convolutional layers allow the model to "see" longer-range patterns:

- **First layer**: short motifs (e.g., 8–15 bp).
- **Higher layers**: combinations of motifs, motif spacing, and local regulatory grammar.
- **Pooling layers** (e.g., max pooling) reduce spatial resolution while aggregating features, increasing the **receptive field**.

In DeepSEA, stacked convolutions and pooling allow the model to use hundreds of base pairs of context around a locus to predict chromatin state (J. Zhou and Troyanskaya 2015). ExPecto extends this idea by mapping sequence to tissue-specific expression predictions (J. Zhou et al. 2018). SpliceAI uses very deep dilated convolutions to reach ~10 kb of context for splicing (Jaganathan et al. 2019).

A.4.3. Multi-Task Learning

Early sequence-to-function CNNs are almost always **multi-task**:

- A single input sequence is used to predict many outputs simultaneously (e.g., hundreds of TF ChIP–seq peaks, histone marks, DNase hypersensitivity tracks).
- Shared convolutional layers learn **common features**, while the final layer has many output units (one per task).

Benefits:

- Efficient use of data and compute
- Better regularization: related tasks constrain each other
- Natural interface for variant effect prediction: you can see how a mutation affects many functional readouts at once

---

A.5. Beyond CNNs: Recurrent Networks (Briefly)

Before Transformers dominated sequence modeling, **recurrent neural networks (RNNs)**—especially LSTMs and GRUs—were the default architecture for language and time series.

Conceptually:

- An RNN processes a sequence one position at a time.
- It maintains a hidden state that is updated as it moves along the sequence.
- In principle, it can capture arbitrarily long-range dependencies.

In practice, for genomic sequences:

- Very long-range dependencies (tens to hundreds of kilobases) are difficult to learn with standard RNNs.
- Training can be slow and unstable on very long sequences.
- CNNs and attention-based models have largely displaced RNNs in genomic applications.

You may still see RNNs in some multi-modal or temporal settings (e.g., modeling longitudinal clinical data), but they are not central to this book's architectures.

A.6. Transformers and Self-Attention

Transformers, introduced in natural language processing, have become the dominant architecture for sequence modeling. In this book, they underpin protein language models, DNA language models (DNABERT and successors), and long-range models like Enformer (Ji et al. 2021; Ž. Avsec et al. 2021).

A.6.1. The Idea of Self-Attention

In a **self-attention** layer, each position in a sequence can directly "look at" and combine information from every other position.

For an input sequence represented as vectors $\{x_1, \ldots, x_L\}$:

1. Each position is mapped to **query** (q_i), **key** (k_i), and **value** (v_i) vectors via learned linear projections.

2. The attention weight from position i to position j is:

$$\alpha_{ij} \propto \exp\left(\frac{q_i^\top k_j}{\sqrt{d}}\right),$$

   followed by normalization so that $\sum_j \alpha_{ij} = 1$.

3. The new representation of position i is a weighted sum of all value vectors:

$$z_i = \sum_{j=1}^{L} \alpha_{ij} v_j.$$

Key properties:

- **Content-based**: Interactions are determined by similarity of representations, not just distance.
- **Global context**: Each position can, in principle, attend to any other position.
- **Permutation-aware via positional encodings**: Additional information (sinusoidal or learned) encodes position so the model knows order.

A.6.2. Multi-Head Attention and Transformer Blocks

Real Transformer layers use **multi-head attention**:

- The model runs self-attention in parallel with multiple sets of (Q, K, V) projections (heads).
- Different heads can specialize in different patterns (e.g., local motif combinations, long-range enhancer–promoter contacts).

A typical Transformer block has:

1. Multi-head self-attention
2. Add & layer normalization
3. Position-wise feed-forward network
4. Another add & layer normalization

Stacking many blocks yields a deep Transformer.

A.6.3. Computational Cost and Long-Range Genomics

Naive self-attention has $O(L^2)$ cost in sequence length L. For genomic sequences, where we might want 100 kb–1 Mb contexts, this is expensive.

Long-range genomic models like Enformer and HyenaDNA address this with:

- **Hybrid designs** (CNNs + attention) to reduce sequence length before applying global attention (Ž. Avsec et al. 2021).
- **Structured state space models (SSMs)** and related architectures that scale more gracefully with length (Nguyen et al. 2023).

These details are treated in depth in the long-range modeling chapters; here it suffices to know that Transformers give flexible global context at the cost of higher computational complexity.

A.7. Self-Supervised Learning and Pretraining

A central theme of this book is **pretraining**: training a large model once on a broad, unlabeled or weakly-labeled task, then re-using it for many downstream problems.

A.7.1. Supervised vs Self-Supervised

- **Supervised learning**: Each input x comes with a label y. Examples:
  - Predicting chromatin marks from sequence (DeepSEA).
  - Predicting splice junctions (SpliceAI).
  - Predicting disease risk from features (Chapter 18).
- **Self-supervised learning**: The model learns from raw input data without explicit labels, using some **pretext task** constructed from the data itself. Examples:
  - Masked token prediction (BERT-style): hide some nucleotides and train the model to predict them from surrounding context.
  - Next-token prediction (GPT-style): predict the next base given previous ones.
  - Denoising or reconstruction tasks.

In genomics, self-supervised models treat DNA sequences as a language and learn from the vast amount of genomic sequence without needing curated labels.

A.7.2. Masked Language Modeling on DNA

DNABERT applied BERT-style masked language modeling to DNA sequences tokenized as overlapping k-mers (Ji et al. 2021). The model:

- Reads sequences as k-mer tokens.
- Randomly masks a subset of tokens.
- Learns to predict the masked tokens given surrounding context.

Benefits:

- Uses essentially unlimited unlabeled genomic data.
- Learns rich representations that can be fine-tuned for tasks like promoter prediction, splice site detection, and variant effect prediction.

Chapter 10 generalizes this story to broader DNA foundation models, including alternative tokenization schemes and architectures.

A.7.3. Pretraining, Fine-Tuning, and Probing

After pretraining, we can use a model in several ways:

- **Fine-tuning**: Initialize with pretrained weights, then continue training on a specific downstream task with task-specific labels.
- **Linear probing**: Freeze the pretrained model, extract embeddings, and train a simple linear classifier on top.
- **Prompting / adapters**: Add small task-specific modules (adapters) while keeping most of the model fixed.

These patterns reappear across protein LMs, DNA LMs, variant effect models, and GFMs in Chapters 9–16.

A.8. Foundations for Evaluation and Reliability

While the main book has dedicated chapters for evaluation (Chapter 15), confounding (Chapter 16), and clinical metrics (Chapter 18), it's useful to have a few basic concepts in mind.

A.8.1. Distribution Shift

A model is trained under some data distribution (e.g., certain assays, cohorts, ancestries) and then deployed under another (e.g., a different hospital system or population). When these differ, we have **distribution shift**, which can degrade performance.

Typical genomic shifts include:

- New sequencing technologies or lab protocols
- New ancestries or populations
- New tissues, diseases, or phenotypes

A.8.2. Data Leakage

Data leakage occurs when information from the test set "leaks" into training (e.g., through overlapping loci or related individuals), leading to overly optimistic estimates of performance. Chapter 15 and Chapter 16 discuss strategies for leak-resistant splits in detail.

A.8.3. Calibration and Uncertainty

For many applications, especially in the clinic, we care not just about whether the model is *correct*, but whether its probabilities are **well calibrated** and whether we know when the model is uncertain. Calibration and uncertainty quantification are covered in Chapter 18; here, the main takeaway is that **perfect AUROC does not imply perfect clinical utility**.

A.9. A Minimal Recipe for a Genomic Deep Learning Project

To make the abstractions more concrete, here is a lightweight "recipe" that roughly mirrors what the case-study chapters do.

1. **Define the prediction problem**
    - Input: e.g., 1 kb sequence around a variant, or patient-level features.
    - Output: e.g., presence of a chromatin mark, change in expression, disease risk.
2. **Choose an input representation**
    - One-hot encoding or tokenization scheme for sequences (see Chapter 8).
    - Encodings for variants, genes, or patients (e.g., aggregate from per-variant features).
3. **Pick a model family**

- CNN for local sequence-to-function (Chapters 5–7).
- Transformer or SSM for long-range or language model-style tasks (Chapters 8–11).
- Pretrained GFM + small task-specific head (Chapters 12–16).

4. **Specify the loss and metrics**
   - Cross-entropy for binary classification, MSE for regression, etc.
   - Metrics like AUROC, AUPRC, correlation, calibration.
5. **Set up data splits and evaluation**
   - Decide whether to split by locus, individual, cohort, or species.
   - Hold out a test set and use validation data to tune hyperparameters.
6. **Train with regularization and monitoring**
   - Use an optimizer (SGD or Adam-like) with a learning rate schedule.
   - Apply regularization (dropout, weight decay, augmentation).
   - Monitor training and validation curves for overfitting.
7. **Inspect and stress-test**
   - Check performance across subgroups (e.g., ancestries, assays, cohorts).
   - Use interpretability tools (Chapter 17) to see what patterns the model is using.
   - Run robustness checks and ablations.
8. **Iterate**
   - Adjust architecture, add more data, refine labels, or incorporate pretrained backbones.
   - Move from model-centric tuning to system-level considerations (data quality, deployment environment, feedback loops).

A.10. How This Primer Connects to the Rest of the Book

This appendix gives you the minimum vocabulary to navigate the rest of the text:

- **Chapters 5–7** show how CNNs on one-hot sequence learn regulatory code, expression, and splicing.

- **Chapters 8–11** extend these ideas to richer sequence representations, Transformers, and long-range sequence models.

- **Chapters 12–16** frame these models as genomic foundation models, introduce evaluation, interpretability, and multi-omics.

- **Chapters 17–19** show how these ingredients are assembled into clinical, discovery, and biotech applications.

You don't need to internalize every detail here. The goal is simply that when you see terms like "convolution," "attention," "pretraining," or "fine-tuning" in the main chapters, they feel like familiar tools rather than mysterious jargon.

B. Additional Resources

> ⚠ Warning
>
> **TODO:**
>
> - ...
> - ...

B.1. Genomics & Human Genetics

- **Thompson & Thompson Genetics and Genomics in Medicine (9th ed.)**
  Ronald Cohn, Stephen Scherer, Ada Hamosh. Clinical-focused overview of human genetics and genomics for medicine, great for grounding in clinical genomics.
- **Human Molecular Genetics (5th ed.)**
  Tom Strachan, Andrew Read. Higher-level molecular genetics/genomics text with strong coverage of mechanisms, technologies, and disease applications.

B.2. Immunology

- **Janeway's Immunobiology (10th ed.)**
  Kenneth M. Murphy, Casey Weaver, Leslie J. Berg. Standard comprehensive immunology textbook, excellent for understanding immune system biology relevant to genomics and disease.

B.3. Machine Learning & Deep Learning

- **Deep Learning**
  Ian Goodfellow, Yoshua Bengio, Aaron Courville. Comprehensive deep learning textbook; free online: https://www.deeplearningbook.org/
- **Dive into Deep Learning (D2L)**
  Aston Zhang et al. Interactive deep learning book with Jupyter notebooks and multi-framework code; free online: https://d2l.ai/
- **An Introduction to Statistical Learning (ISLR, 2nd ed.)**
  Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. Gentle introduction to statistical learning methods used in ML, available free online: https://www.statlearning.com/

- **The Elements of Statistical Learning (ESL)**
  Trevor Hastie, Robert Tibshirani, Jerome Friedman. More advanced, theory-heavy companion to ISLR; free PDF: https://hastie.su.domains/ElemStatLearn/

C. Glossary

C.1. CH 01

C.1.1. Sequencing Technologies & Data

Next-generation sequencing (NGS)

High-throughput DNA sequencing technologies that allow rapid…stretches of DNA, producing millions of short reads in parallel….

Illumina sequencing

A widely used NGS technology that utilizes reversible dye-terminators to sequence DNA by synthesis

Short reads / Paired-end reads

DNA sequences generated by NGS technologies, typically rangi… a DNA fragment, providing additional information for alignment.

Long-read sequencing (PacBio HiFi, Oxford Nanopore)

DNA sequencing technologies that produce longer reads, typic…ases, allowing for better resolution of complex genomic regions.

Circular consensus sequencing

A sequencing method used by PacBio to generate highly accura… long reads by repeatedly sequencing the same DNA molecule.

Base calling

The process of determining the nucleotide sequence from raw sequencing data.

FASTQ

A file format that stores both nucleotide sequences and their corresponding quality scores.

Read depth / Coverage (e.g., 30×, 100×)

The number of times a particular nucleotide is sequenced, indicating the reliability of the sequencing data.

C.1.2. Targeting Strategies

Targeted gene panel

A sequencing approach that focuses on a specific set of gene…or cost-effective analysis of known disease-associated variants.

Whole-exome sequencing (WES)

A sequencing approach that targets all protein-coding region…the genome, providing a comprehensive view of coding variants.

Whole-genome sequencing (WGS)

A sequencing approach that captures the entire genome, inclu…ng regions, providing the most comprehensive view of genetic var

Capture efficiency

The effectiveness of a targeted sequencing approach in enric…interest, impacting the overall quality and coverage of the data.

C.1.3. Alignment & Processing

Read alignment / Mapping

The process of aligning sequencing reads to a reference genome to determine their genomic origin.

Seed-and-extend alignment

An algorithmic approach for read alignment that first identi…ences (seeds) and then extends the alignment around these seeds.

PCR duplicates

Identical sequencing reads that originate from the same DNA …esulting from PCR amplification, which can bias variant calling.

Base quality score recalibration (BQSR)

A process that adjusts the quality scores of sequencing read…or systematic errors made by the sequencer.

Mapping quality

A measure of the confidence that a read is correctly aligned to the reference genome.

Reference bias

The tendency for sequencing and alignment processes to preferentially detect alleles present in the reference genome.

C.1.4. Variant Calling

Variant calling

The process of identifying variants from sequencing data by comparing it to a reference genome.

Genotype likelihood

The probability of observing the sequencing data given a particular genotype.

Pair-HMM (pair hidden Markov model)

A statistical model used in variant calling to calculate the likelihood of different alignments between reads and the reference genome.

Joint genotyping / Cohort calling

The process of simultaneously calling variants across multiple samples to improve accuracy and consistency.

gVCF (genomic VCF)

A variant call format that includes information about both va…sites, allowing for joint genotyping.

VCF (variant call format)

A standardized file format for storing variant information, including SNPs, indels, and structural variants.

VQSR (Variant Quality Score Recalibration)

A method for improving the accuracy of variant calls by mode…lationship between variant quality scores and various annotatio

Pileup

A summary of the base calls at each position in a set of align…ing reads, used for variant calling and visualization.

C.1.5. Phasing

Haplotype phasing

The process of determining which variants are inherited together on the same chromosome.

Read-backed phasing

A method of phasing that uses sequencing reads that span multiple variants to determine their phase.

Statistical phasing

A method of phasing that uses population-level genotype data and statistical models to infer haplotypes.

Compound heterozygosity

The presence of two different variants at a particular gene locus, one on each chromosome of a pair.

Cis vs. trans configuration

Describes the relative arrangement of two variants on the same chromosome (cis) or on different chromosomes (trans).

C.1.6. Variant Types

SNV (single nucleotide variant)

A variation in a single nucleotide that occurs at a specific position in the genome.

Indel

An insertion or deletion of bases in the genome of an organism.

Structural variant

A large-scale alteration in the genome, such as a deletion, duplication, inversion, or translocation.

Multi-nucleotide variant (MNV)

A variation that affects multiple consecutive nucleotides in the genome.

Mosaic variant

A genetic variant that is present in some but not all cells of an organism, often arising during development.

Somatic variant

A genetic variant that occurs in non-germline cells and is not inherited, often associated with cancer.

Germline variant

A genetic variant that is present in the egg or sperm and can be passed on to offspring.

De novo variant

A genetic variant that arises spontaneously in an individual and is not inherited from either parent.

C.1.7. Difficult Regions

Segmental duplication

Large, highly similar sequences in the genome that can complicate read alignment and variant calling.

Paralog / Paralogous gene

A gene that is related to another gene in the same organism due to a duplication event.

Homopolymer

A sequence of identical nucleotides in a row, which can be prone to sequencing errors.

Low-complexity region

A region of the genome with a simple sequence composition, which can be challenging for alignment and variant calling.

HLA region / MHC

The human leukocyte antigen (HLA) region, also known as the major histocompatibility complex (MHC), is a highly polymorphic region involved in immune response.

C.1.8. Benchmarking

Precision (positive predictive value)

The proportion of true positive variant calls among all positive calls.

Recall (sensitivity)

The proportion of true positive variant calls detected among all actual variants.

F1 score

The harmonic mean of precision and recall, providing a single metric for evaluating variant calling performance.

True positive (TP) / False positive (FP) / False negative (FN)

Metrics used to evaluate the accuracy of variant calls, where TP represents correctly identified variants, FP represents incorrectly identified variants, and FN represents missed variants.

High-confidence region

A region of the genome where variant calls are considered to b… reliable, often used for benchmarking and validation.

C.1.9. Key Resources/Tools (may warrant brief glossary entries)

GIAB (Genome in a Bottle)

A consortium that develops reference materials and data for benchmarking genome sequencing and variant calling.

DeepVariant

A deep learning-based variant caller developed by Google that identifies genetic variants from sequencing data.

GLnexus

A tool for joint variant calling across multiple samples, designed to work with DeepVariant outputs.

HaplotypeCaller

A variant caller from the Genome Analysis Toolkit (GATK) that uses local de-novo assembly of haplotypes to call variants.

C.2. CH 02

C.2.1. Reference & Coordinate Systems

Reference genome/assembly

A digital nucleic acid sequence database, assembled as a repre…example of a species' set of genes. Multiple versions exist.

GRCh37

The 37th version of the Genome Reference Consortium human genome assembly.

GRCh38

The 38th version of the Genome Reference Consortium human genome assembly.

T2T-CHM13

The Telomere-to-Telomere (T2T) CHM13 human genome assembly, r…ting a complete, gapless sequence of a human genome.

Pangenome reference

A reference that represents the genetic diversity of a species, rather than a single individual.

Gene model

A representation of the structure of a gene, including its exons, introns, and regulatory elements.

Canonical transcript

The most biologically relevant transcript of a gene, often used as the reference for annotation.

Alternative transcript/isoform

Different versions of a transcript produced from the same gene due to alternative splicing or other mechanisms.

MANE Select

Matched Annotation from NCBI and EMBL-EBI (MANE) Select is a …cripts that are consistently annotated across databases.

C.2.2. Variant Types & Properties

Allele frequency

The proportion of a specific allele among all alleles of a gene in a population.

MAF (minor allele frequency)

The frequency at which the less common allele occurs in a given population.

rsID

A unique identifier assigned to a single nucleotide polymorphism (SNP) in the dbSNP database.

Loss-of-function (LoF) variant

A genetic variant that results in reduced or abolished protein function.

Ultra-rare variant

A genetic variant that is extremely uncommon in the population, often with a frequency of less than 0.01%.

C.2.3. Population Genetics Metrics

Linkage disequilibrium

A non-random association of alleles at different loci in a given population.

pLI (probability of being loss-of-function intolerant)

A metric that estimates the likelihood that a gene is intolerant to loss-of-function variants.

LOEUF (loss-of-function observed/expected upper bound fraction)

A metric that quantifies the observed versus expected number of loss-of-function variants in a gene.

Constraint metrics

Metrics that assess the tolerance of a gene to functional genetic variation.

Imputation

The process of inferring unobserved genotypes in a study sample based on observed genotypes and a reference panel.

C.2.4. Functional Genomics

ChIP-seq

Chromatin Immunoprecipitation followed by sequencing, a method used to analyze protein-DNA interactions.

DNase-seq

A method to identify regions of open chromatin by sequencing DNA fragments generated by DNase I digestion.

ATAC-seq

Assay for Transposase-Accessible Chromatin using sequencing, a technique to study chromatin accessibility.

Hi-C

A method to study the three-dimensional architecture of genomes by capturing chromatin interactions.

Chromatin accessibility

The degree to which DNA is exposed and available for binding by proteins, often assessed by DNase-seq or ATAC-seq.

Histone modification

Chemical modifications to histone proteins that can influence chromatin structure and gene expression.

Peak calling

The process of identifying regions of the genome with signific…ment of sequencing reads, often used in ChIP-seq and ATAC-seq analyses.

Signal track

A graphical representation of sequencing data across the genom… intensity of signals such as read coverage or enrichment.

C.2.5. Expression Genetics

eQTL (expression quantitative trait locus)

A genomic locus that explains variation in gene expression levels.

Splicing QTL

A genomic locus that affects the splicing of pre-mRNA.

Molecular QTL

A quantitative trait locus that influences molecular traits such as gene expression, protein levels, or metabolite concentrations.

Cis-regulatory

Referring to regulatory elements, such as promoters or enhanc…ated on the same molecule of DNA as the gene they regulate.

Colocalization

The occurrence of two or more genetic signals at the same genomic location, suggesting a shared causal variant.

Dropout (single-cell context)

The failure to detect a transcript in a single-cell RNA-seq ex…often due to low mRNA capture efficiency.

C.2.6. Clinical Interpretation

ACMG/AMP criteria

A set of guidelines developed by the American College of Medic…ogy (AMP) for the interpretation of sequence variants. These c…vide a framework for classifying variants into categories such path

Pathogenicity

The ability of a genetic variant to cause disease.

Haploinsufficiency

A condition in which a single functional copy of a gene is in…maintain normal function, leading to a disease phenotype.

Triplosensitivity

A condition in which an extra copy of a gene leads to a disease phenotype.

Gene-disease validity

The strength of evidence supporting a relationship between a gene and a disease.

Pharmacogenomics

The study of how genetic variation affects an individual's response to drugs.

Diplotype

The combination of alleles at multiple loci on a single chromosome that are inherited together.

C.2.7. Study Designs & Statistics

GWAS summary statistics

Aggregated data from genome-wide association studies, typicall…ormation on the association between genetic variants and traits across the genome.

Fine-mapping

The process of identifying the specific causal variants within…omic region associated with a trait.

Effect size

A measure of the strength of the relationship between a genetic variant and a trait.

Ascertainment bias

A systematic distortion in the estimation of genetic effects d…non-random sampling of individuals or variants.

C.3. CH 03

C.4. CH 04

C.5. CH 05

C.6. CH 06

C.7. CH 07

C.8. CH 08

C.9. CH 09

C.10. CH 10

C.11. CH 11

C.12. CH 12

C.13. CH 13

C.14. CH 14

C.15. CH 15

C.16. CH 16

C.17. CH 17

C.18. CH 18

C.19. CH 19

C.20. CH 20

C.21. APX A

C.22. APX B