Modeling Gene Expression using Chromatin Features

Chuting Xu

December 18, 2024

1 Project Overview

1.1 Motivation and Importance

The overarching goal of this project is to reproduce a simplified version of the approach described in the referenced paper, "Modeling gene expression using chromatin features in various cellular contexts" (Dong et al., 2012). In that study, the authors demonstrated that the levels and patterns of certain chromatin features (including histone modifications and chromatin accessibility) can be used to predict gene expression levels across different cell lines and experimental conditions.

In this project, we adapt the core two-step modeling approach introduced by the authors, which involves a classification step to determine whether a gene is "on" or "off," followed by a regression step to predict the expression level of genes classified as "on." Unlike the broader scope of the original work, this simplified project focuses exclusively on the genes located on chromosome 1 and uses data from the GM12878 cell line, a type of lymphoblastoid cell line. By narrowing the scope, we aim to verify and validate the fundamental concepts in a controlled setting.

Why Is This Important?

Understanding Gene Regulation: By linking chromatin features to gene expression, we gain insights into epigenetic regulation. The chromatin state plays a crucial role in determining whether genes are turned "on" (expressed) or "off" (not expressed).
Predictive Modeling: Developing accurate predictive models allows researchers to estimate gene expression solely from chromatin data. Such predictive capabilities can facilitate understanding gene regulation when direct expression measurements are not available, helping guide experimental designs and hypotheses.
Benchmarking and Validation: Reproducing a simplified version of the methodology in a tightly controlled setting (here focusing on chromosome 1 of GM12878) helps verify and validate the approach. Ensuring reproducibility and reliability serves as a benchmark for future expansions of the methodology to other chromosomes, cell lines, or more complex conditions.

1.2 Data Used and Preprocessing Steps

1.2.1 Data Sources

Chromatin Features: We utilized BigWig files containing raw signal tracks for multiple histone modifications, one histone variant, and DNase I hypersensitivity data from the GM12878 cell line. Specifically, the histone marks include H3K4me3, H3K9ac, H3K36me3, H3K79me2, H3K4me1, H3K4me2, H3K9me3, H4K20me1, H3K27me3, H3K27ac, and the histone variant H2A.Z, along with DNase I hypersensitivity as an indicator of open chromatin (Table 1). These features serve

as predictors for gene expression, reflecting various regulatory states of the chromatin. While some of these marks (e.g., H3K4me3, H3K9ac, H3K27ac, H2A.Z, and DNase) are primarily associated with promoters and TSS regions, others (e.g., H3K36me3, H3K79me2) are more indicative of transcriptional elongation within gene bodies, and certain marks (e.g., H3K9me3, H3K27me3) can reflect repressive or heterochromatic states that are not restricted to the TSS.

These histone modification data are based on extensive mapping efforts as described in Koch et al. (2007), who characterized the landscape of histone modifications across the human genome.

Predictor	**Associated Regulatory Role**
H3K4me3, H3K9ac, H3K27ac	Active promoters, TSS-proximal regions
H2A.Z	Regulatory elements including promoters
DNase	Chromatin accessibility (often promoter/enhancer regions)
H3K36me3, H3K79me2	Gene body, transcription elongation
H3K4me1	Enhancers/distal regulatory regions
H3K4me2	Promoter-proximal, but less strictly than H3K4me3
H3K9me3, H3K27me3, H4K20me1	Repressive chromatin marks, heterochromatin

Table 1: List of predictors (histone modifications, histone variant, and DNase) and their general regulatory contexts. While TSS-focused binning aligns with CAGE and TSS-associated marks, other marks may be more informative in downstream or distal regions.

Gene Annotations: Gene and transcript structures were obtained from a GTF file (e.g., gencode.v7.annotation). This annotation was used to identify transcription start sites (TSSs) and assign transcripts to genes (Frankish et al., 2019).
Expression Data: CAGE (Cap Analysis of Gene Expression) data in BigWig format was employed to estimate gene expression at TSSs. Since CAGE captures transcription initiation events, it is most relevant to promoter-proximal regulatory features.

1.2.2 Gene Selection and Filtering

Chromosome Filter: To simplify computations and provide a controlled test environment, we focused exclusively on chromosome 1.
Length Filter: Transcripts shorter than a predefined minimum length (e.g., 4,100 bp) were excluded. This length threshold ensured more consistent binning around the gene body and flanking regions.
TSS Definition: For genes with multiple transcripts, we selected the TSS corresponding to the highest CAGE signal, ensuring that the representative transcript reflected the strongest initiation site. This step reduces ambiguity and aligns the predictive modeling with the most biologically relevant start site.

1.2.3 Bin Creation for CAGE-based Expression

Since we are only using CAGE data, which inherently focuses on transcription initiation rather than elongation or termination, we centered our binning strategy around the TSS. Instead of creating bins around both the TSS and transcription termination site (TTS), we created a set of bins that span upstream and downstream regions of the TSS. Specifically, for each selected transcript, we defined a window (e.g., 2,000 bp upstream and 2,000 bp downstream) around the TSS and divided

it into a series of bins (e.g., 40 bins upstream, 40 bins downstream, plus a bin covering the TSS region).

In practice, let L_{up} be the upstream distance and L_{down} the downstream distance relative to the TSS. Then the total number of bins N_{bins} can be expressed as:

$$N_{\text{bins}} = N_{\text{up}} + N_{\text{down}} + 1,$$

where $N_{\text{up}} = 40$ and $N_{\text{down}} = 40$ are the numbers of equally sized bins upstream and downstream, respectively, and the '+1' represents the bin covering the TSS itself. This approach focuses on promoter-proximal features most relevant to transcription initiation, as captured by CAGE. It aligns well with marks known to have strong signals at promoters (e.g., H3K4me3, H3K9ac, H3K27ac, H2A.Z, DNase), but may not fully capture features that are more informative along the gene body or near the TTS (e.g., H3K36me3, H3K79me2, H3K9me3, H3K27me3).

The original article by Dong et al. (2012) employed a more extensive binning strategy: 40 bins around the TSS, 40 bins around the TTS, and one bin covering the remaining gene body, totaling 81 bins. This approach was designed to capture both promoter-proximal signals and features associated with transcription elongation and termination.

In contrast, our simplified approach focuses solely on the TSS region. While this aligns well with CAGE data (which primarily reflects initiation events), it may omit information that could be gleaned from gene-body or TTS-proximal features. Future work could consider integrating a binning scheme more similar to the original study to further improve predictive performance.

1.2.4 Normalization and Transformation

CAGE Normalization: CAGE signals were normalized to Reads Per Million (RPM) to account for differences in sequencing depth. If C_i is the raw CAGE count for a gene i and T is the total number of mapped CAGE reads (in millions), then:

$$\text{CAGE}_{\text{RPM},i} = \frac{C_i}{T}.$$

Chromatin Signal Transformation: Each bin's chromatin signal (for histone modifications and DNase I hypersensitivity) was aggregated and then transformed to a logarithmic scale. Since some bins may have zero counts, we added a small pseudocount α (e.g., $\alpha = 0.1$) before taking the logarithm:

$$\text{Signal}_{\log,i} = \log_2(\text{Signal}_i + \alpha).$$

This ensures that no undefined values occur and that the dynamic range of the data is compressed, making downstream statistical modeling and correlation analysis more stable and robust.

In the original article (Dong et al. 2012), an optimization procedure was used to determine the best pseudocount for log-transformation. In our study, we chose a fixed pseudocount (e.g., $\alpha = 0.1$) after trying a few values and finding that the overall results were very similar. While we did not perform a formal optimization step, this simplified approach did not substantially alter the performance compared to results reported by Dong et al.

1.3 Data Structure and Size

After applying the filtering criteria and selecting the most expressed TSS for each gene, we obtained a total of 2,506 genes on chromosome 1. We then divided these genes into two subsets:

- **D1 (Discovery Set):** One-third of the genes (835 genes) were randomly selected for the purpose of identifying the 'best bin' for each chromatin feature. This step determines which genomic bin around the TSS provides the strongest correlation with gene expression.
- **D2 (Modeling Set):** The remaining two-thirds of the genes (1,671 genes) were used for subsequent model training and evaluation. For these genes, the chromatin feature values were extracted from the best bins identified using D1.

1.3.1 Feature Matrix (X)

For each gene in D2 and each chromatin feature, we took the chromatin signal from the best bin (determined using D1) and applied a logarithmic transformation with a small pseudocount. Stacking all genes as rows and all chromatin features as columns, we formed a feature matrix $\mathbf{X}$.

If we denote the number of genes in D2 by N_{genes} and the number of chromatin features by N_{features}, then:

$$\mathbf{X} \in \mathbb{R}^{N_{\text{genes}} \times N_{\text{features}}}.$$

In our case, $N_{\text{genes}} = 1671$ and $N_{\text{features}} = 12$.

1.3.2 Response Vector (Y)

The response vector $\mathbf{Y}$ contains the gene expression levels derived from CAGE data. We used a log-transform for nonzero values and assigned zero to genes with no expression signal:

$$Y_i = \begin{cases} \log_2(\text{CAGE}_{\text{RPM},i}) & \text{if } \text{CAGE}_{\text{RPM},i} > 0, \\ 0 & \text{otherwise.} \end{cases}$$

1.3.3 Binary Classification Label (is_on)

To implement the two-step modeling strategy, we also defined a binary label is_on to distinguish between "on" (expressed) and "off" (not expressed) genes. A gene is considered "on" if:

$$Y_i > 0.1.$$

This binary label facilitates the initial classification step, after which the regression step is applied only to the genes predicted to be on.

Note that the details of how these data structures are integrated into the modeling pipeline, and how the best bins from D1 are used to train models on D2, will be described in subsequent sections.

2 Model Training and Evaluation

2.1 K-Fold Cross-Validation

To rigorously assess the predictive performance of our models, we employed a k-fold cross-validation (CV) scheme. Specifically, we used $k = 10$ folds, partitioning the set of N_{genes} genes into 10 roughly equal subsets. For each iteration:

1. One subset was held out as the test set.
2. The remaining nine subsets were used to train the model.

3. The trained model was then applied to the held-out test set to obtain predictions.

By cycling through all 10 subsets, every gene served as a test gene exactly once. This approach reduces the chance that results are driven by peculiarities of a single train/test split and provides a more stable estimate of model performance.

2.2 Models and Predictive Framework

The modeling approach is based on a two-step procedure:

1. **Classification Step:** Predict whether a gene is "on" or "off."
2. **Regression Step:** For genes predicted "on," estimate their expression levels.

We considered three classifiers for the first step and four regressors for the second step, resulting in a total of 12 (3×4) model combinations.

2.2.1 Classification Models

Logistic Regression (LR): Logistic regression models the probability that a gene is "on" using the logistic function:

$$P(\text{on} \mid \mathbf{x}) = \frac{1}{1 + e^{-(\beta_0 + \beta^\top \mathbf{x})}}.$$

This model is relatively simple, interpretable, and well-suited for smaller datasets.
Random Forest (RF) Classifier: A random forest classifier builds numerous decision trees and aggregates their predictions. Each tree is grown on a bootstrap sample of the training data and considers a random subset of features at each split. This ensemble approach often improves accuracy and reduces overfitting.
Support Vector Machine (SVM) Classifier: SVMs find a decision boundary (hyperplane) that separates "on" and "off" genes with maximum margin. Nonlinear kernels can capture complex relationships, but SVMs may be less interpretable and could be sensitive to data scale and sample size.

2.2.2 Regression Models

For genes predicted "on," we considered the following regressors:
Lasso Regression: Lasso solves:

$$\min_{\theta} \sum_{i} (Y_i - \theta^\top \mathbf{x}_i)^2 + \lambda \|\theta\|_1,$$

where Y_i is the log-transformed expression. By penalizing the L_1 norm of the coefficients, it enforces sparsity, making it suitable for high-dimensional, low-sample-size settings.
Random Forest (RF) Regressor: A random forest regressor averages predictions from multiple regression trees, each built on a random sample of the data and features. This often yields a robust nonparametric model that can handle nonlinearities and interactions.
Multivariate Adaptive Regression Splines (MARS): MARS approximates functions with piecewise linear segments. It automatically selects which interactions and nonlinearities to model. Though flexible, MARS might be prone to overfitting when the sample size is small.
SVM Regressor: SVM regression fits a function $f(\mathbf{x})$ that deviates from Y_i by at most a certain amount, aiming to find a balance between model complexity and fitting accuracy. Nonlinear kernels can capture complex relationships, but may require careful tuning.

2.2.3 Modeling Complexity and Sample Size Considerations

While some of these models are highly flexible and can model complex gene regulation patterns, the relatively small number of genes (1,671 in the modeling set) may limit the benefits of high complexity. Models like SVMs and MARS may not generalize well without careful tuning and regularization, due to potential overfitting.

In contrast, simpler models like logistic regression or more robust ensemble methods like random forests may perform better under these constraints. As we will show in the results, random forests emerged as top performers among the tested models.

2.2.4 Twelve Model Combinations

Combining the three classification methods (LR, RF Classifier, SVM Classifier) with the four regression methods (Lasso, RF Regressor, MARS, SVM Regressor) resulted in 12 distinct two-step models. Each combination provides a unique balance of complexity, interpretability, and robustness.

We will present performance comparisons of these 12 model combinations in the results section.

3 Classification Performance

3.1 Metrics and Definitions

Two primary metrics were used to evaluate classification performance:

Misclassification Rate: This is the proportion of genes incorrectly classified as "on" or "off." Formally, if we let y_i be the true label for gene i and $\hat{y}_i$ be the predicted label, the misclassification rate is:

$$\text{Misclass} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(y_i \neq \hat{y}_i),$$

where $\mathbb{I}(\cdot)$ is an indicator function and N is the total number of test genes.

Area Under the Receiver Operating Characteristic (ROC) Curve (AUC): AUC measures the ability of the classifier to rank positive examples higher than negative examples. An ROC curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various decision thresholds. The AUC is the area under this curve, with a value of 1.0 indicating perfect classification and 0.5 indicating no better than random guessing.

3.2 Results

Table 2 presents the mean misclassification rates and AUC values computed over 10-fold cross-validation for three classifiers: Logistic Regression, Random Forest (RF) Classification, and Support Vector Machine (SVM) Classification.

Model	Misclassification	AUC
Logistic Regression	0.0987	0.9408
RF Classifier	0.0975	0.9417
SVM Classifier	0.1694	0.9144

Table 2: Average classification performance over 10-fold CV. Lower misclassification and higher AUC indicate better performance.

Both Logistic Regression and the RF Classifier achieved low misclassification rates and high AUCs, indicating strong predictive ability. The SVM Classifier performed slightly worse, exhibiting a higher misclassification rate and a lower AUC. These results suggest that simpler or more robust ensemble methods may be more suitable given the data constraints.

3.3 ROC Curve Example

Figure 1 shows an example ROC curve for the Random Forest classifier from one fold of the cross-validation. The curve demonstrates a high True Positive Rate at relatively low False Positive Rates, consistent with the high AUC observed.

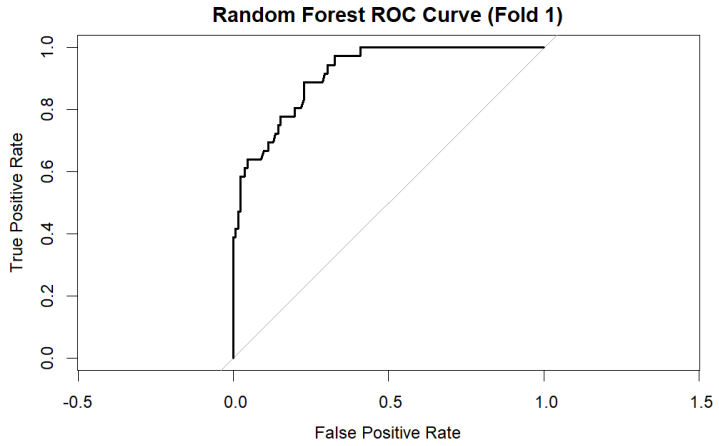

Figure 1: ROC curve for the Random Forest classifier on one fold of the 10-fold CV. The curve approaches the top-left corner, indicating strong discriminative ability.

4 Regression Performance

4.1 Metrics and Interpretations

For the regression component (applied only to genes predicted as "on"), we evaluated two metrics:
Root Mean Squared Error (RMSE_log): RMSE in the log-transformed expression space measures the average deviation between predicted and true log-expression values:

$$\text{RMSE\_log} = \sqrt{\frac{1}{N_{\text{on}}} \sum_{i \in \{\text{on}\}} (Y_i - \hat{Y}_i)^2},$$

where N_{on} is the number of genes considered "on" and $\hat{Y}_i$ is the predicted log-expression. Lower RMSE_log indicates more accurate predictions.
Pearson's Correlation (r): Pearson's r measures the linear correlation between predicted and true log-expression values:

$$r = \frac{\sum (Y_i - \bar{Y})(\hat{Y}_i - \overline{\hat{Y}})}{\sqrt{\sum (Y_i - \bar{Y})^2} \sqrt{\sum (\hat{Y}_i - \overline{\hat{Y}})^2}}.$$

A higher r (closer to 1) indicates stronger linear agreement between predictions and true values.

4.2 Results

Table 3 summarizes the mean RMSE_log and Pearson's r over 10-fold CV for the 12 model combinations. Each combination pairs one of the three classifiers (Logistic, RF, SVM) with one of the four regression models (Lasso, RF Regression, MARS, SVM Regression).

Two-Step Model	**RMSE_log**	**Pearson r**
Logistic + Lasso	2.7381	0.4361
Logistic + RF_Reg	2.7072	0.4682
Logistic + MARS	2.7610	0.4264
Logistic + SVM_Reg	2.8437	0.4032
RF_Clf + Lasso	2.6862	0.4264
RF_Clf + RF_Reg	2.6351	0.4739
RF_Clf + MARS	2.6825	0.4331
RF_Clf + SVM_Reg	2.7590	0.4130
SVM_Clf + Lasso	2.7051	0.4309
SVM_Clf + RF_Reg	2.6982	0.4663
SVM_Clf + MARS	2.7611	0.4116
SVM_Clf + SVM_Reg	2.8435	0.3866

Table 3: Mean regression performance over 10-fold CV for 12 two-step models. Lower RMSE_log and higher Pearson r indicate better performance.

4.3 Analysis of the Results

From these results, two clear patterns emerge:

1. **Impact of the Classification Model on Regression Performance:** Comparing the same regression method paired with different classifiers reveals that combinations using the Random Forest (RF) classifier consistently yield lower RMSE_log and higher Pearson r. For example, when paired with Lasso, the RF classifier results in a better RMSE_log (2.6862) than Logistic (2.7381) or SVM (2.7051) classifiers. This suggests that correctly identifying "on" genes (a step in which the RF classifier excels) leads to a more accurate and stable set of genes for the subsequent regression step.

2. **RF Regression Outperforms Other Regression Methods:** Among the regression models, RF regression tends to provide better performance within each classification category. For instance, the RF classifier combined with RF regression achieves an RMSE_log of 2.6351 and an r of 0.4739, surpassing the other regressors paired with the RF classifier. Similarly, RF regression outperforms Lasso, MARS, and SVM regression across other classifier choices.

In summary, a two-step modeling approach that employs the Random Forest classifier to accurately select "on" genes followed by the Random Forest regressor for predicting their expression levels provides the best predictive accuracy. These results highlight the benefit of selecting robust, ensemble-based methods given the relatively small sample size and complex gene regulation relationships.

4.4 Predicted vs. Actual Expression Example

Figure 2 illustrates the relationship between the predicted and actual log-transformed expression values for the Random Forest classifier + Random Forest regressor combination on a single fold of the 10-fold cross-validation. The red dashed line represents a simple linear fit between predicted and actual values.

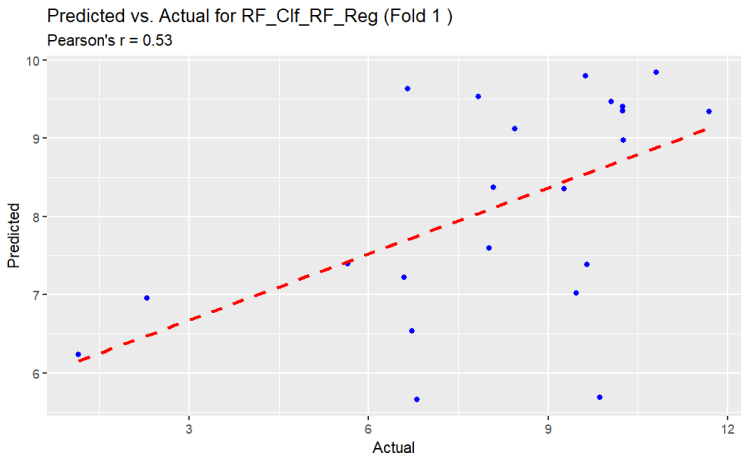

Figure 2: Predicted vs. actual log-expression for the RF_Clf_RF_Reg model on one fold. Pearson's $r \approx 0.53$.

A Pearson correlation of approximately 0.53 indicates a moderate linear relationship. While not extremely high, this level of correlation is not uncommon in genetic data, where biological complexity, technical noise, and limited sample size can constrain predictive performance. As the size of the dataset grows or as more predictive features become available, we might expect this correlation to improve, reflecting a clearer relationship between predicted and actual gene expression levels.

5 Variable Importance Analysis

After identifying the Random Forest (RF) classifier + Random Forest regressor (RF_Clf_RF_Reg) as the best-performing two-step model, we sought to gain insights into which chromatin features most strongly influenced the classification and regression tasks. To do this, we reran the chosen model configuration multiple times (e.g., 10 iterations) on the D1 and D2 datasets. Each run trains an RF model, which internally uses bootstrap samples of the data and random subsets of features at each split, thereby providing estimates of variable importance that are robust to sampling variation.

5.1 Variable Importance for Classification

The RF classifier computes importance scores indicating how much each feature reduces the Gini impurity across the trees. High importance suggests that a feature is frequently used at splits that improve predictive accuracy for distinguishing "on" from "off" genes.

Figure 3 shows the MeanDecreaseGini distributions for all chromatin features over multiple runs. Several key observations emerge:

- **Promoter-Associated Marks Dominate:** Features like H3K9ac and H3K79me2, along with DNase hypersensitivity, rank among the top. H3K9ac is known to be enriched at ac-

tive promoters, and DNase hypersensitivity signifies open chromatin—both of which strongly correlate with transcription initiation.

- **H3K4me3 and H3K27ac also Contribute:** H3K4me3 is a hallmark of active promoters, and H3K27ac is associated with active regulatory elements. Their relatively high importance underscores the focus on transcription start regions inherent in CAGE-based expression quantification.

- **Lower-Ranked Features:** Marks typically associated with gene bodies or repressive states, such as H3K36me3, H3K9me3, or H3K27me3, show lower importance for classification. This is consistent with the TSS-centered binning approach and the nature of the classification task (distinguishing "on" vs. "off" at initiation rather than capturing elongation or repression states).

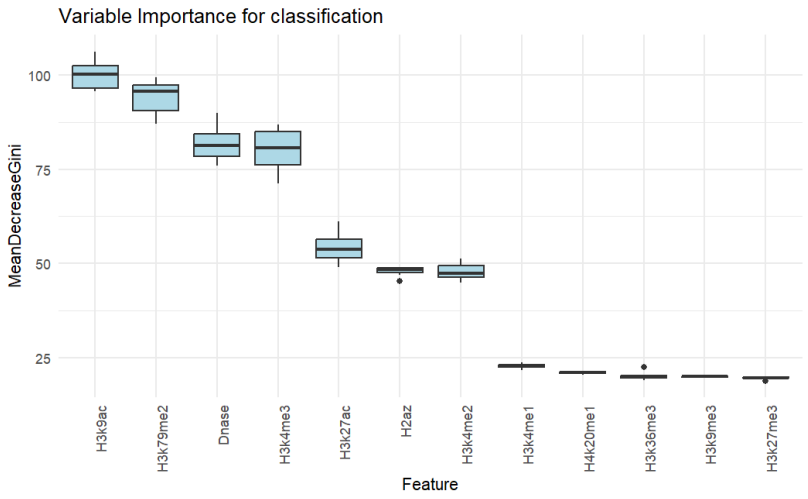

Figure 3: Variable importance (MeanDecreaseGini) for classification over multiple runs of RF_Clf. Higher values indicate greater importance for predicting whether a gene is "on" or "off."

5.2 Variable Importance for Regression

For the regression step, the RF regressor computes importance via the IncNodePurity measure, indicating how much each feature improves node purity (reduces residual variance) across the trees.

Figure 4 shows the IncNodePurity distributions for the same features in the regression stage. Here, the pattern differs:

- **DNase Stands Out:** DNase hypersensitivity emerges as the top predictor for quantitative expression levels among expressed genes. This suggests that once a gene is "on," the extent of open chromatin at or near its TSS may correlate with higher or lower expression magnitude.

- **Active Histone Marks:** Features such as H3K9ac and H3K79me2 also score highly, indicating that they contribute to refining the expression level predictions. H3K79me2, known to be associated with transcription elongation, may be more informative at the quantitative level of expression rather than the binary on/off state.

- **Less Influential Marks:** Similar to classification, repressive and distal marks (H3K27me3, H4K20me1) remain at the bottom, suggesting their limited role in predicting continuous expression levels in this dataset and modeling setup.

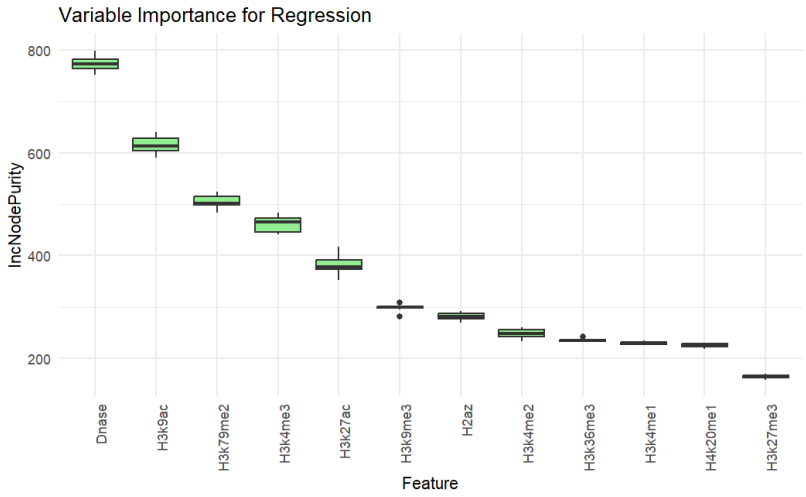

Figure 4: Variable importance (IncNodePurity) for regression over multiple runs of RF_Reg. Higher values indicate greater importance for predicting the actual expression level among genes predicted to be "on."

5.3 Interpretation, Biological Context, and Comparison with Published Findings

The variable importance patterns reflect the biology of transcription initiation and regulation:

- **Classification (On/Off):** Features that define an active promoter environment (e.g., H3K9ac, DNase accessibility, H3K4me3) are most crucial. This aligns with CAGE data, which measures initiation at or near promoters.
- **Regression (Intensity of Expression):** After a gene is classified as on, features that signal ongoing transcription (e.g., DNase and certain active marks) help to refine predictions of expression level. Marks like H3K79me2, associated with elongation, may be more useful at this stage.

Re-running the models multiple times and observing consistent ranking patterns provides confidence that these importance values are not artifacts of a single run. Instead, they represent stable indicators of which chromatin features contribute most to understanding gene expression regulation under this modeling framework.

Notably, our variable importance results show strong parallels with those reported by Dong et al. (2012). In that study, H3K9ac emerged as one of the most influential histone marks for predicting gene expression, a finding that aligns closely with our analysis, where H3K9ac also ranked at or near the top in both classification and regression importance. Beyond H3K9ac, other features such as DNase accessibility, H3K27ac, H3K4me2, H3K79me2, and H2A.Z played prominent roles in both our work and the original article's findings.

For example, H3K79me2, often associated with transcription elongation, consistently contributed to refining expression level predictions in our dataset. Similarly, DNase hypersensitivity,

which indicates open chromatin, and histone marks like H3K27ac and H3K4me3—both associated with active promoters—were identified as important predictors in both analyses. While the exact ranking of relative importance differs slightly between our results and those of Dong et al. (2012), the overall pattern of top predictive features remains remarkably consistent.

This close correspondence suggests that our simplified approach and focus on the GM12878 cell line for genes on chromosome 1 still captures fundamental aspects of the relationship between chromatin features and gene expression identified in a more comprehensive analysis. Thus, the key regulatory marks identified here not only have robust predictive power in our controlled setting but also echo established findings in the broader literature.

References

[Dong et al.(2012)] Dong, X., Greven, M.C., Kundaje, A. et al. (2012). Modeling gene expression using chromatin features in various cellular contexts. *Genome Biology*, 13:R53. https://doi.org/10.1186/gb-2012-13-9-r53

[R Core Team(2012)] R Core Team (2012). *R: A language and environment for statistical computing.* R Foundation for Statistical Computing, Vienna, Austria. http://www.R-project.org/

[Frankish et al.(2019)] Frankish, A., Diekhans, M., Ferreira, A.M., et al. (2019). GENCODE reference annotation for the human and mouse genomes. *Nucleic Acids Research*, 47(D1):D766–D773. https://doi.org/10.1093/nar/gky955

[The ENCODE Project Consortium(2012)] The ENCODE Project Consortium (2012). An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489:57–74. https://doi.org/10.1038/nature11247

[Koch et al.(2007)] Koch, C.M., Andrews, R.M., Flicek, P., Dillon, S.C., Karaöz, U., Clelland, G.K., Wilcox, S., Beare, D.M., Fowler, J.C., Couttet, P., James, K.D., Lefebvre, G.C., Bruce, A.W., Dovey, O.M., Ellis, P.D., Dhami, P., Langford, C.F., Weng, Z., Birney, E., Carter, N.P., Vetrie, D., Dunham, I. (2007). The landscape of histone modifications across 1*Genome Research*, 17(6):691–707. https://doi.org/10.1101/gr.5704207