Identifying conserved binding sites regulating circadian patterns in *Brassica rapa*

Joan Barreto (jbarreto), Jennie Cheng (chen7312), Ekin Ercetin (ercet001), Plinio Rosales (rosal072),
Grace Wurgler (wurgl007)

Abstract

In plants, studying the circadian rhythms of carbon, starch, and glucose metabolism can give insight into crop productivity in different growth conditions. Response to stress is controlled by circadian genes, and increasing stress conditions threaten crops and food security. Gaining more understanding of the transcriptomic and metabolic responses may lead to the development of crops that are able to better withstand stress conditions. The goal of this project was to use machine learning methods to find possible conserved binding sites regulating circadian patterns in *Brassica rapa* RNAseq data. Four models of conventional machine learning to determine an optimal method to predict the phase of circadian genes included: random forest regressor (RF), support vector regressor (SVR), ridge regression, and XGBoost. In addition, deep neural networks with varying hidden layers with 256 neurons were used. After cross-validation, nonlinear models were determined to predict the phase for circadian genes the best, in particular random forest and XGBoost, respectively. It was found that three sequences namely, AATGGGCA, ATGTGGCG, and TAAACGTC were found to be among the top 10 most important features for the models tested. These sequences were identified as potential binding sites, suggesting that they may play a crucial role in regulating circadian patterns in Brassica RNAseq data. Using the Plant Transcription Factor Database (PTFD), it was found that the sequences may contribute significantly to a transcription factor's ability to bind.

Introduction

The first observation of the circadian rhythm was documented in 1729, by Jean Jacques d'Ortous de Mairan. De Mairan observed a *Mimosa* plant in a light-tight dark room and noticed that its leaves would begin to open when there was light and close when it was dark (Huang, 2018). Circadian rhythms represent a “subset of biological rhythms with period” that are endogenously (internally) generated and are persistent over constant environmental factors, such as temperature or light availability (McClung, 2006). They have become known as “biological clocks.” Circadian rhythms are measured over a period of approximately 24 hours.

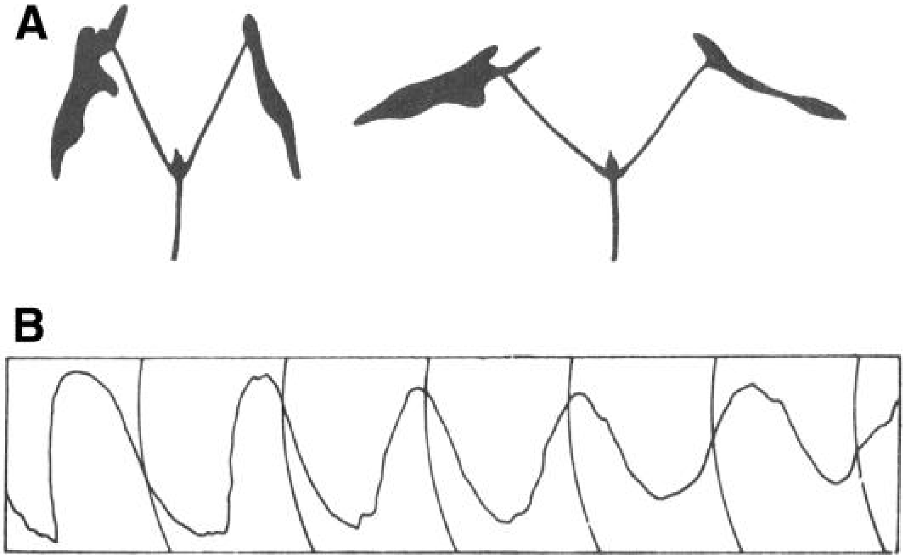


Figure 1. (A) Sleep movements of *Phaseolus coccineus*. The position of the primary leaves of a seedling at night is at the left and during the day is at the right. (B) Circadian rhythm of leaf movements of *P. coccineus* entrained to light/dark cycles and monitored in continuous light. As can be inferred from the leaf positions in (A), the peaks of the curve represent the nighttime leaf position. The vertical lines indicate 24-h intervals. The period for this trace is ~27 h. Retrieved from (McClung, 2006). (A) was originally published as Figure 14 and (B) as Figure 4 in Chapter 2 in Bünning (1973).

Circadian rhythms include physical and physiological changes, such as the leaf movements shown in Figure 1, flower, fragrance emission, or gas exchange (McClung, 2006). Control of transcription via circadian rhythms is well known across organisms, including cyanobacteria, *Neurospora*, *Drosophila*, mice, and *Arabidopsis* (Dunlap, 1999). Labeled “circadian genes,” previous research suggests that there is an extensive list of genes controlled by circadian rhythms, influencing multiple metabolic pathways, and 35% of the *Arabidopsis* transcriptome may show circadian regulation (Michael & McClung, 2003).

Understanding the cause and effect of circadian rhythms is important for understanding crop productivity. Common mechanisms studied from an agricultural lens are glucose, starch, and carbon metabolism (Kim et al., 2017). Rapidly changing climates generate stress conditions that threaten agricultural crops and food security across the globe. Plant response to abiotic stress is influenced by circadian genes, the expression of which regulates metabolic and physiological pathways throughout the day (Greenham et al., 2017). Understanding the diel transcriptomic and metabolomic responses to abiotic stress may facilitate the development of crops that are more resilient to changing climates (Greenham et al., 2017).

The goal of this project is to evaluate computational approaches to identify conserved binding sites for transcription factors that regulate circadian genes. We will focus on the species *Brassica rapa* (field mustard), a crop of economic relevance, and a model for comparative studies as its genus comprises the closest crop relatives to Arabidopsis (Greenham et al., 2017). There are over 4000 known transcription factors in the *B. rapa* genome that are sorted into 58 families (Jin et. al., 2017). The most common families include ERF, C2H2, bHLH, MYB, and NAC (Jin et. al., 2017).

We will use machine learning techniques to identify cis-regulatory elements within a 2 kilobase (kb) upstream region across several morphotypes (conserved). This region will be converted to features for machine learning models to predict specific genes' circadian patterns including phase, peak, and trough.

Materials and Methods

Preprocessing

Raw Data

The raw data for this project was kindly provided by Dr. Katie Greenham from the College of Biological Sciences at the University of Minnesota. This included RNAseq data generated using Illumina NovaSeq 6000 S4 with the objective of identifying differentially expressed genes associated with circadian patterns and cold hardiness, using multiple cultivars of *Brassica rapa*, a 32 h period with 4 h intervals, and four replicates. On April 11, 2023, we obtained the normalized counts ('tpm_counts.txt') as transcript per million (TPM) for 39,581 genes that were mapped to the *B. rapa* subsp. *trilocularis* (Yellow Sarson) variety R500 reference genome (Lou et al., 2020). We also received the metadata for the libraries ('LIBRARIES_KEY.txt') containing information for all genotypes, which enabled us to begin the filtering and preprocessing steps. We later received the genomes files (FASTA) for genotypes R500, L58, W083, A03, VT123 PCGlu and O302V, and their JGI annotations in General Feature Format (GFF3). All raw and processed files can be located at "/home/myersc/jbarreto/CSCI5461" through the Minnesota Supercomputing Institute (MSI), the code for the further steps can be openly accessed from the GitHub repository https://github.com/joanmanbar/CSCI5461_2023_FinalProject and associated forks.

Filtering

For the purpose of this project, we ignored the expression count from the cold treatment and only focused on seven time points on the control data for the genotypes with available genomes. After this filtering step, we added columns with NA values for genotypes missing replicates or time points, as required by MetaCycle(Wu et al., 2016), the R tool we used to identify circadian genes.

Identifying Circadian Genes

We screened for circadian genes, using the "JTK_CYCLE" algorithm with timepoints 17, 21, 25, 29, 33, 37, 41, and the function "meta2d" with default values. The results from the complete analysis comprise of a .txt file per genotype with the gene id, as well as their period, amplitude, and phase, the latter being the phenotype of choice to predict. The output also includes p-values adjusted with the Benjamini-Hochberg method, which we used to determine whether a gene was circadian or not based on a threshold lower than 1%.

Accessing Promoter Regions

Because the expression data were mapped to R500, we required pairwise syntenic hits for the other JGI genotypes of interest. We received this data on April 19, and began the steps to identify promoter regions using the SAMtools (Li et al., 2009) and BEDTools (Quinlan & Hall, 2020) programs, on MSI. For this analysis, we created a .bash file (get_2kb_fasta.sh) that begins with renaming FASTA files for convenience, and loops through each genome to first generate an index file (.fai) using the 'faidx' function from SAMtools that helps generate the file with chromosome sizes ('chrom.sizes'). Next, we read the genome's GFF3 file and filter it to contain only genes, and this was converted to a .bed file with gene coordinates that were used along with the 'chrom.sizes' file as inputs for the 'flank' function in BEDTools that generate the coordinates for the 2kb sequence upstream the promoter region that we aim to extract. On April 25, we finally identified those regions using the as input for the 'getfasta' function from BEDTools the files generated in previous steps.

Feature extraction

Transcription factor binding sites range from 5 to 15 base pairs (bp) within the promoter region or *Arabidopsis* (Fujita et al., 2016), a close relative of *B. rapa*. However, extracting features that represent those potential sites from a DNA sequence is challenging despite the multiple ways to accomplish it. Our approach relied on a sliding window of 8bp across the 2kb sequence to extract sequences with an overlap of 3bp, such that these sequences can then be one-hot-encoded as features representing the presence (one) or absence (zero) of the sequences across genes (see example GeneFeatures_v01.ipynb in Github repository). This resulted in 399 features per gene and a total of 65,375 features assessed for presence or absence across genes, i.e., a 55,894 by 65,375 matrix with binary data. This highly sparse and redundant matrix with more features than observations, had a density distribution ranging from 2-25,000, where the 25th, 50th, and 75th percentiles were 104, 195, and 24,500, respectively. This indicates a need for feature selection, for example, based on density, cosine similarity, or dimensional reduction methods such as principal component analysis, before fitting regression models.

Models

Both conventional machine learning (ML) techniques and deep neural networks (DNN) were used to predict the phase of circadian rhythms for the circadian genes. The input for the models that were tested consisted of a 2kb region upstream for all genes. This region was split and converted into features with a length of eight nucleotides (8bp), overlapping three nucleotides for each adjacent feature. This method of extracting features from a 2kb region resulted in 65,546 features for the 55,893 circadian genes from the 8 genotypes. Thus, the structure of the dataset consisted of 65,564 columns and 55,893 rows. The target for all the models was the phase that had a column title of “JTK_adjphase”. A Python script was written to transform the ~66k features in the 2kb region into a one hot encoded matrix (binary matrix with the 0 or 1 to represent whether or not the specific feature is in the 2kb sequence for each gene) to be used as input for the models. Furthermore, the scikit-learn (sklearn) package, through the function test-train-split, was used to split the data into test and training sets using a 20% test size. The training sets were used as inputs for the following models.

Conventional Machine Learning

Python scripts were written for four conventional machine-learning models using the sklearn package. These models include random forest regressor (RF), support vector regressor (SVR), Ridge regression, and XGBoost. The aim of this part was to compute the predicted phases for each of the models, and the predicted importance of features. The most important features were identified as potential conserved binding sites regulating circadian patterns in *Brassica rapa*.

Random Forest Regressor

Random forest regressor is a form of supervised machine learning which takes in training data, determines characteristic thresholds of the data, and uses those thresholds to create a collection of decision trees that will classify test data. The Python script first one-hot encoded all features of the input gene region so that each gene has an associated target value and a binary array of features (0 for not containing, 1 for containing). It then uses the sklearn python package to split the data into training and testing sets and uses the RandomForestRegressor function on the training set. The number of estimators (number of trees) used in the model was set to 100 and the maximum depth for each tree was set to 10. This algorithm generated target phase prediction values that were measured against the true target phase value to determine the accuracy of the model using both the root-mean-square error (RMSE) and the Pearson correlation coefficient (PCC). The feature importance method implemented was the default method in the RandomForestRegressor function that estimates the importance based on the mean decrease in impurity. Hyperparameter tuning of the number of estimators and max-depth was not performed due to time constraints. Finally, 3-fold cross-validation was used to determine the robustness of this model.

Support Vector Regressor

Support vector regressor (SVR) makes predictions about a dataset based on a computed best line (or hyperplane) of fit given a provided acceptable error threshold. This nonparametric regression method is dependent on the kernel or “smoothing function”. The Python script first one-hot encoded all features of the input gene region so that each gene has an associated target value and a binary array of features (0 for not containing, 1 for containing). It then used the sklearn python package to split the data into training and testing sets and the SVR function from sklearn was used on the training set. The regularization hyperparameter for SVR, “C”, was set to 1 for all trials. The hyperparameter “C” represents the penalty, which is a squared L2 penalty. In addition, two different kernels were tested. The default linear kernel was used, in addition to a Radial Basis function (rbf) kernel. The importances of the features were computed using the sklearn function permutation_importance. Root-mean-square error (RMSE) and the Pearson correlation coefficient (PCC) were used to assess the performance of the models. In addition, 3-fold cross-validation was used to determine the robustness of this model. Finally, due to time constraints, hyperparameter tuning of C was not performed.

Ridge Regression

Ridge regression is a form of linear regression. Unlike ordinary linear regression (OLS) or LASSO, Ridge uses a norm-2 (L2) penalty on the weights. In other words, Ridge adds bias to the predictions using a regularization parameter “alpha” to avoid overfitting. The biological relevance of Ridge can be seen in a plethora of research papers, including a recent paper published by Lee and Myers et al. that uses Ridge to investigate gene expressions to reveal TP53 mutant-like AML with wild-type TP53 and poor prognosis (Lee et al., 2023). Thus, Ridge was determined to have great biological relevance to this project. If the hyperparameter alpha is too large, the results from the model may result in less overfit to the training data, but could risk over-smoothing or underfitting of the model. However, if the hyperparameter alpha is too small, the results from the model may be overfitted to the training set. The Python script first one-hot encoded all features of the input gene region so that each gene has an associated target value and a binary array of features (0 for not containing, 1 for containing). It then used the sklearn python package to split the data into training and testing sets and the SVR function from sklearn was used on the training set. Then, feature importance was determined by retrieving the coefficients of the result of the ridge regression function that describe the weights of each feature in determining the prediction value for the ridge regression model. Similarly, RMSE and PCC were used as metrics to measure the accuracy of the Ridge models. Finally, 10-fold cross-validation was performed to determine the robustness of the model and, due to Ridge being the model that was computationally least expensive, hyperparameter tuning was performed on Ridge varying alpha from 0.1 to 30.

XGBoost
XGBoost is an algorithm that implements the method of gradient boosting, where decision trees are continuously built based on the prediction error of past trees, and the result from the whole collection of trees is a regression prediction. This method boosts various models (trees) that form an ensemble. The Python script first one hot encodes all features of the input gene region so that each gene has an associated target value and a binary array of features (0 for not containing, 1 for containing). The script then uses the sklearn python package to split the data into training and testing sets and the GradientBoosting function from sklearn was used on the training set. The number of estimators was set to 100 trees, and the max depth of each tree was set to 10. Feature importance was found using the 'feature_importances' function from the GradientBoosting class. Additionally, 3-fold cross-validation was performed to determine the robustness of the model. Similarly to the other models, RMSE and PCC were the metrics used to determine the accuracy of the predictions. Finally, hyperparameter tuning of the number of estimators and max-depth of each tree was not performed due to time constraints.

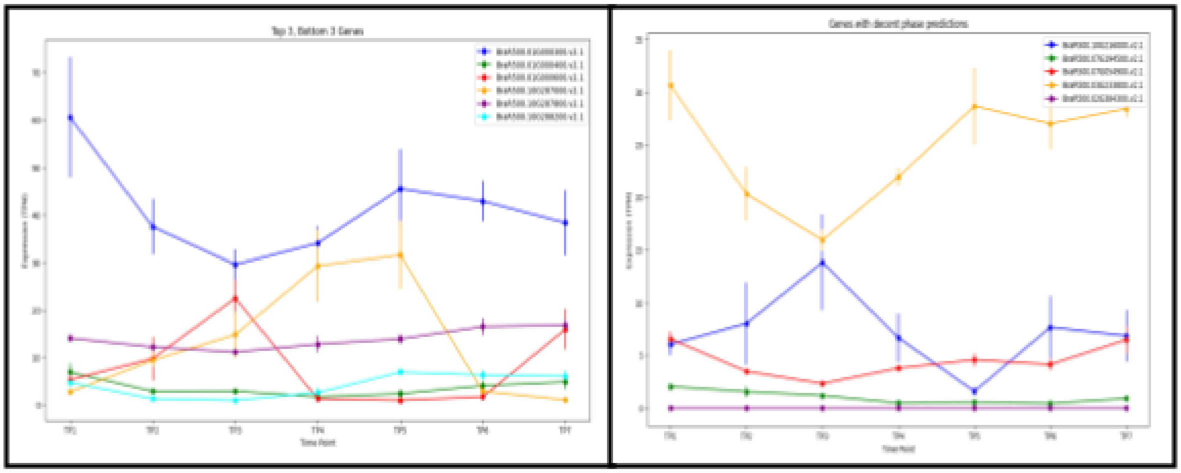
Neural Networks

Similar to the conventional machine learning models, The python script first one-hot encoded all features of the input gene region so that each gene has an associated target value and a binary array of features (0 for not containing, 1 for containing). It then used the sklearn python package to split the data into training and testing sets. The Python package PyTorch was used to implement the scripts to train neural networks. To begin, a class that links together a feature matrix and a target array called "JointDataSet" was developed in addition to a function that loads the data and transforms the arrays (feature matrix and target vector) to torch tensors. This was done because PyTorch strictly works with Torch tensors. In addition, a function that constructs the neural network model was constructed. The inputs include the number of hidden layers, the activation function, the norm, and the dropout rate. For this problem, the activation function was chosen to be the Rectified Linear Unit (ReLu) activation function due to its ability to capture non-linear trends. The RMSE was used as the metric to measure the accuracy of the predictions. A function that trains various epochs for batches of size n was written for both a training and a validation set. The optimizer used for this function was the Adam Optimizer. All these functions were put together into one function. For each hidden layer, 256 neurons were chosen and 100 epochs were trained that tracked the scoring metric (RMSE) and the loss function. The dropout rate and the L2 penalty were set to 0.1 and 0.01, respectively. The learning rate was set to 1e-4. Due to time constraints, all these hyperparameters could not be tuned properly. The output of the neural networks consisted of RMSE scores and loss function scores for both the validation and training sets, in addition to predicted phases for the testing set.

Results

Metacycle Analysis

From the output file generated by MetaCycle, the top three and bottom three clock genes were identified based on their respective scores shown in Figure 2 (b). Specifically, the chosen genes consisted of the highest and lowest scores to include in the analysis. Clock genes were selected based on their phase prediction accuracy in the output file generated by conventional machine learning algorithms shown in Figure 2 (b). Specifically, genes with strong phase predictions were identified to include in the graph. Plotting clock genes based on their expression values can provide insights into the circadian rhythm of plants and how it affects various physiological processes. In plants, the circadian rhythm controls several critical processes, such as photosynthesis, flowering time, and hormone production. By studying the expression patterns of clock genes, researchers can better understand the molecular mechanisms involved in these processes and identify the key regulatory pathways that control the circadian rhythm in plants.



a) b)

Figure 2. The expression levels (y-axis) of clock genes are plotted as a line graph connecting the expression values for every time point (x-axis), providing a visualization of the circadian rhythm in gene expression.

Conventional Machine Learning

For the conventional ML, the XGBoosting and Random Forest models performed the best with test RMSEs of 2.60 and 2.59, respectively. This is equivalent to relative RMSEs of 10%, which was determined to be a decent fit. In addition, the Ridge regression consisted of a test RMSE of 3.30, equivalent to a relative RMSE of 13%. Furthermore, the two different SVR models tested, with linear and radial basis function (rbf) kernels, had RMSEs of 3.15 and 2.63, corresponding to 12% and 10% relative RMSEs, respectively.

Feature importance analysis was used for the RF, XGBoost, Ridge, and SVR models, and each shared important features or sequences. For simplicity, the comparison between XGBoost and Ridge is shown in Figure 2. Three important sequences were found to be among the top 10 most important features for the four models tested. The sequences were AATGGGCA, ATGTGGCG, and TAAACGTC.

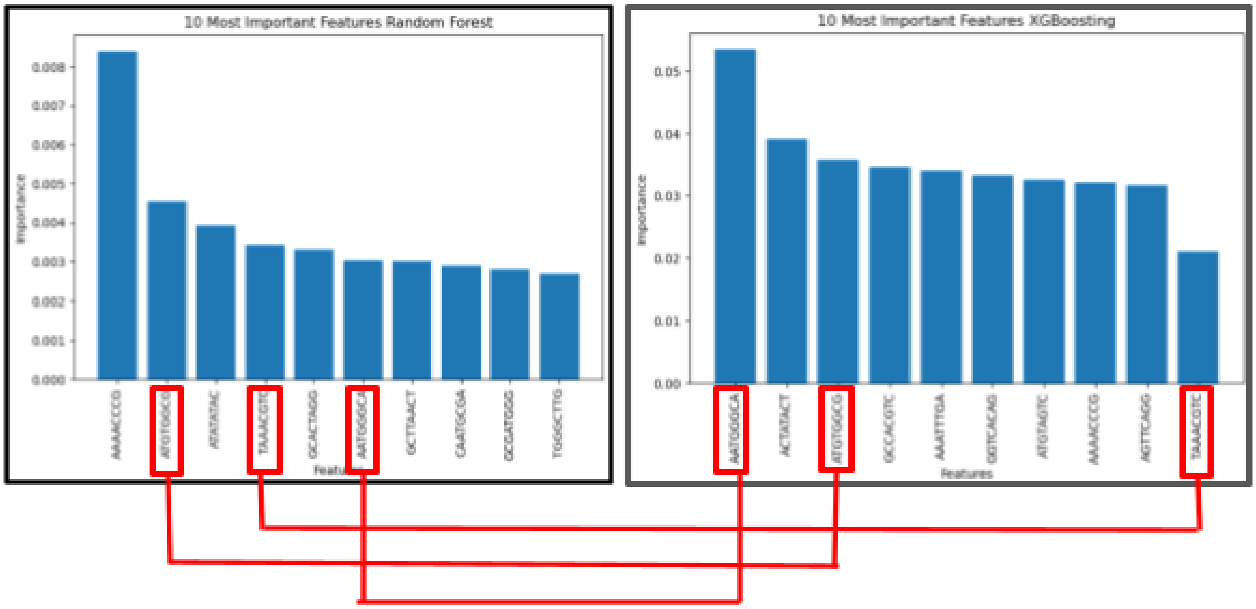


Figure 3. Bar plots depicting the importance of a feature versus the feature name for the Random Forest regressor (left) and XGBoosting (right). In red, three sequences that appear in the models tested as part of the ten most important features in all the models tested. These sequences represent potential binding sites.

In addition to the feature importance analysis evaluated in all the models, hyperparameter tuning was performed on the Ridge regression model. Ten-fold cross-validation was performed, and the regularization hyperparameter alpha was varied from 0.1 to 30. The results of the hyperparameter tuning can be observed in Figure 4. Using the GridSearchCV function, the optimal L2 regularization hyperparameter for Ridge regression was determined to be 5.

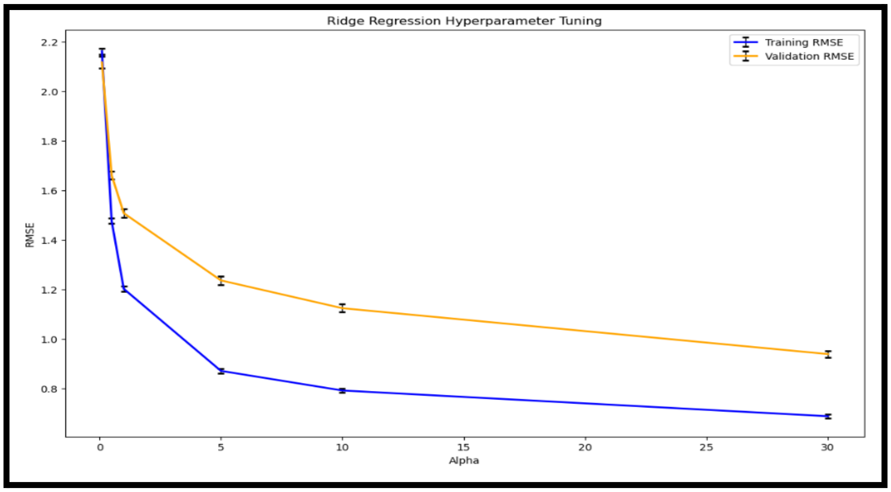


Figure 4. L2 regularization hyperparameter tuning. Using the GridSearchCV function that uses the hyperparameter, model, and scores as inputs, the optimal alpha value was determined to be 5.

Additionally looking at Figure 6, it can be observed that RF and XGB have the highest number of correctly predicted phases, with counts of 3223 and 3217, respectively. Ridge regression and SVR with a radial basis function kernel also perform reasonably well, with counts of 2788 and 2314, respectively. On the other hand, SVR with a linear kernel has the lowest number of correctly predicted phases, with a count of only 438. These results suggest that RF and XGB may be the most effective machine-learning models for predicting the circadian phase of genes.

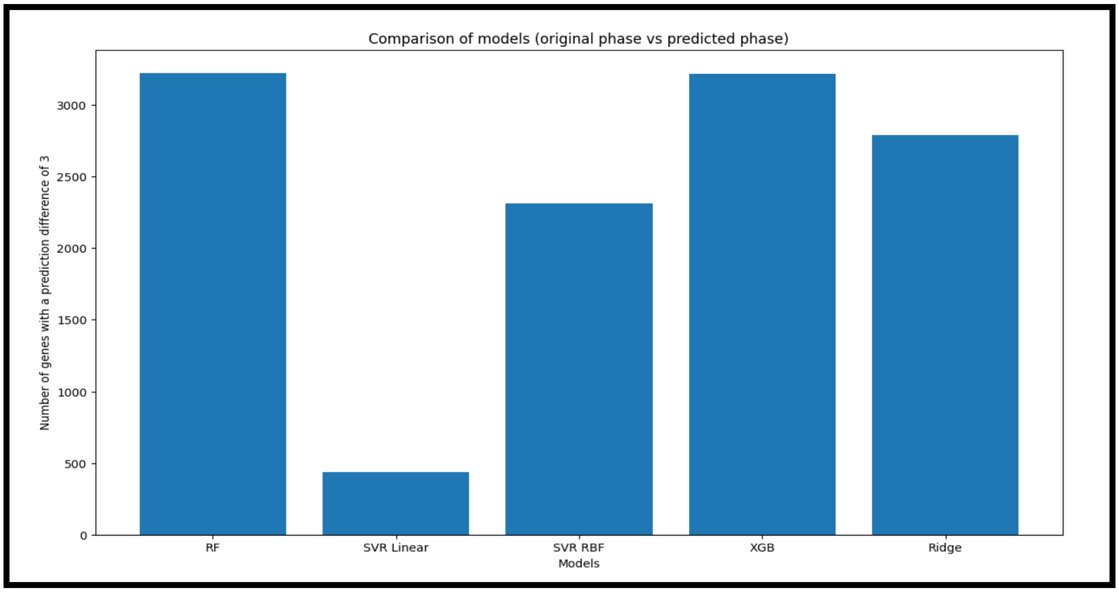


Figure 5. The bar graph compares the performance of several machine learning models in predicting the circadian phase of genes. The x-axis shows the different models being compared, including random forest (RF), support vector regression with linear kernel (SVR Linear), support vector regression with radial basis function kernel (SVR RBF), extreme gradient boosting (XGB), and Ridge regression. The y-axis shows the number of genes whose predicted phase is within 3 hours of original value.

Neural Networks

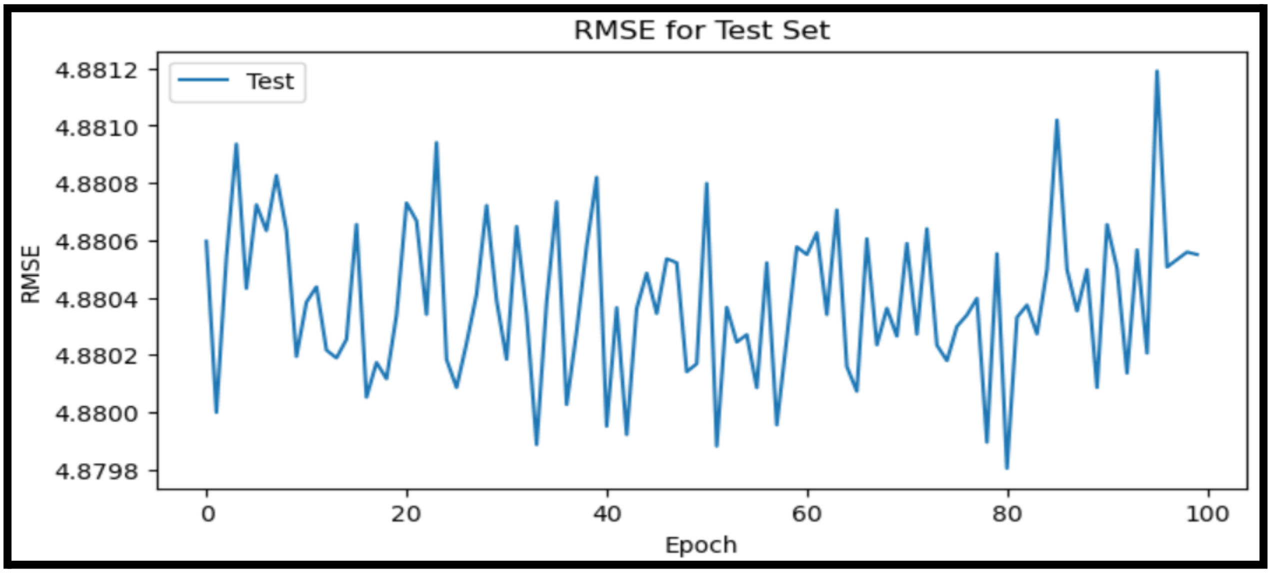


Figure 6. Test RMSE for each Epoch. Three hidden layers with 256 neurons each.

The results of the neural networks can be seen in Figure 6. It can be observed that the RMSE as a function of the epoch tested is random. This indicates that the loss function did not decrease as the number of epochs increased, which is unexpected. The results were similar for all the different structures tested that included one, two, and three hidden layers. In addition, the dropout rate and the L2 penalty were set to 0.8 and 0.75, respectively. Based on these results, it could be seen that the neural network model was overfitted and the loss function did not decrease as expected. Finally, Figure 7 shows a summary of the performance of each of the models tested, including their training and test RMSEs.

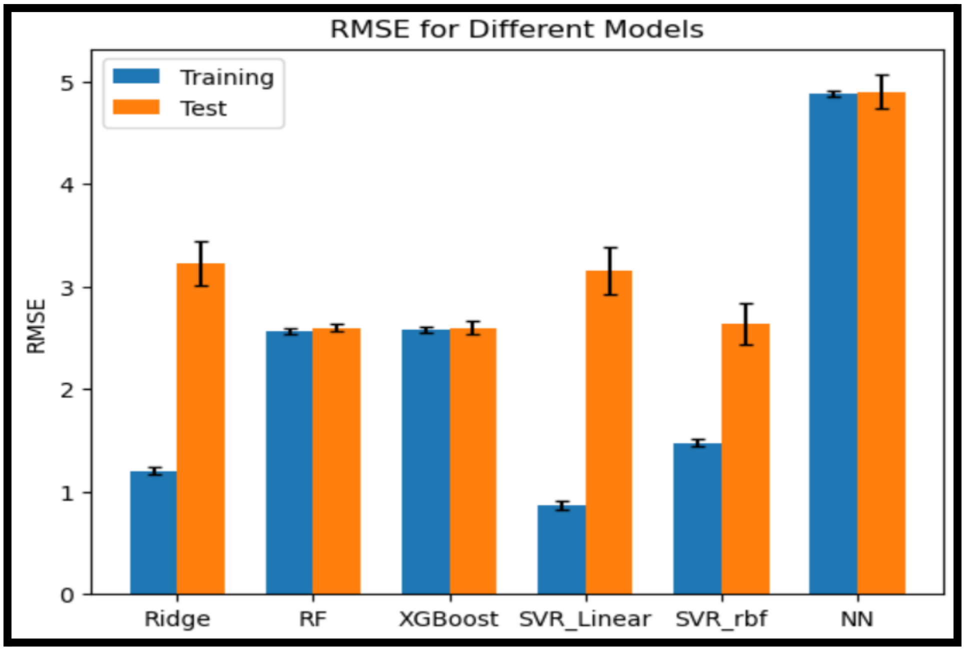


Figure 7. Training and Test RMSEs for each model tested.

Discussion and Conclusion

Based on the results, after cross-validation, the models that performed the best were the random forest and the XGBoosting models with relative RMSEs of around 10%. Although this is a decent fit, a relative RMSE value of less than 5% is desired for a good fit or predictions. This could potentially be achieved after tuning the number of estimators and the max depth of the trees for both of these models. In addition, the Ridge and SVR models were determined to have relative test RMSEs of 13%, but the training scores show overfitting of the data as the training scores are vastly lower than the test scores. Furthermore, the deep learning approach, the neural networks, performed the worst with a lack of loss function decrease. The lack of good performance for the neural networks could be attributed to the lack of hyperparameter tuning that includes parameters such as the number of epochs, the batch size, the number of hidden layers, the dropout rate, and the L2 penalty. Similarly, the performance of the SVR could be increased in the future by hyperparameter tuning the regularization parameter C. Even though Ridge was hyperparameter tuned, due to time constraints, other alpha parameters that could have given more accurate predictions were not tested. It is important to note that the radial basis function kernel performed better than the linear for the SVR model. Similarly, random forest and XGBoost performed better than Ridge. This suggests that nonlinear models might predict the phase of circadian genes better.

The relative RMSEs of the models (ranging from 2.58 RMSE for the random forest and 4.88 RMSE for the neural networks) were adequate to perform feature importance analysis. Based on the feature importance analysis results, three sequences were identified to be among the 10 most important features for the models tested. These included AATGGGCA, ATGTGGCG, and TAAACGTC. Due to their importance in predicting the phase of the circadian genes, these three sequences were identified as potential binding sites, suggesting that they may play a crucial role in regulating circadian patterns.

The most important features identified were further analyzed with the Plant Transcription Factor Database (PTFD) (Jin et. al., 2017). The PTFD allows input of DNA sequences and compares the inputs to known transcription factor binding sites specifically in *B. rapa.* There were 16 total inputs, and only 1 was found to be significant: GGTCACAG. This feature binds a transcription factor in the WRKY family. The protein "interacts specifically with the W box (5'-(T)TGAC[CT]-3'), a frequently occurring elicitor-responsive cis-acting element" (Jin et. al., 2017). GGTCACAG is not one of the most important shared features from the XGBoost and RF results, but it is a known binding site. AATGGGCA, ATGTGGCG, and TAAACGTC may not be the exact binding sites for transcription factors but can still contribute significantly to a transcription factor's ability to bind.

Overall, the results show that the XGBoost and RF models performed the best in predicting the circadian phase, followed by the SVR and ridge regression models. Several significant potential binding sites that may regulate the circadian rhythm in *B. rapa* were found, which can provide insight into the regulation of circadian rhythms and lead to the development of crops that are more resilient to changing climates. As a direction for future research on the topic, it is suggested to investigate potential improvements to the phase prediction accuracy. Specifically, these can be done by exploring hyperparameter tuning for the conventional machine learning algorithms and neural network models utilized in the analysis to address issues related to overfitting. The hyperparameters that need to be tuned include the number of estimators and max tree depth for the XGBoost and the random forest models. In addition, the regularization parameter C from the SVR model should be tuned for future investigations. The neural networks also require parameter tuning. Different learning rates, dropout rates, and L2 penalties should be tested in future work. In addition, a greater number of epochs and batch sizes should be tested to assess model robustness. This will further optimize the methods and potentially improve the accuracy of our results and predictions giving a different insight to other potential binding sites.

References

Bünning, E. (1973). The physiological clock; circadian rhythms and biological chronometry. (Rev. 3d ed., Elberg science library; v. 1). London : New York: English Universities Press; Springer-Verlag.

Dunlap, J. C. (1999). Molecular Bases for Circadian Clocks. *Cell*, *96*(2), 271–290. https://doi.org/10.1016/S0092-8674(00)80566-8

Fujita, T., Fujii, H., & Miyoshi, H. (2016). Characteristics of DNA-binding proteins determine the shapes of binding sites. Scientific Reports, 6, 25164. https://doi.org/10.1038/srep25164

Greenham, K., Guadagno, C. R., Gehan, M. A., Mockler, T. C., Weinig, C., Ewers, B. E., & McClung, C. R. (2017). Temporal network analysis identifies early physiological and transcriptomic indicators of mild drought in Brassica rapa. *ELife*, *6*, e29655. https://doi.org/10.7554/eLife.29655

Huang, R.-C. (2018). The discoveries of molecular mechanisms for the circadian rhythm: The 2017 Nobel Prize in Physiology or Medicine. *Biomedical Journal*, *41*(1), 5–8. https://doi.org/10.1016/j.bj.2018.02.003

Jin JP, Tian F, Yang DC, Meng YQ, Kong L, Luo JC and Gao G. (2017). PlantTFDB 4.0: toward a central hub for transcription factors and regulatory interactions in plants. Nucleic Acids Research 45, D1040-D1045. [full text]

Kim, J. A., Kim, H.-S., Choi, S.-H., Jang, J.-Y., Jeong, M.-J., & Lee, S. I. (2017). The Importance of the Circadian Clock in Regulating Plant Metabolism. *International Journal of Molecular Sciences*, *18*(12), 2680. https://doi.org/10.3390/ijms18122680

Lee, Y., Baughn, L. B., Myers, C. L., & Sachs, Z. (2023). *Machine learning investigation of gene expression datasets reveals TP53 mutant-like AML with wild type TP53 and poor prognosis* (p. 2023.02.22.529592). bioRxiv. https://doi.org/10.1101/2023.02.22.529592

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., ... & 1000 Genome Project Data Processing Subgroup. (2009). The sequence alignment/map format and SAMtools. *bioinformatics*, *25*(16), 2078-2079.

Lou, P., Woody, S., Greenham, K., VanBuren, R., Colle, M., Edger, P. P., ... & McClung, C. R. (2020). Genetic and genomic resources to study natural variation in Brassica rapa. *Plant Direct*, *4*(12), e00285.

McClung, C. R. (2006). Plant Circadian Rhythms. *The Plant Cell*, *18*(4), 792–803. https://doi.org/10.1105/tpc.106.040980

Michael, T. P., & McClung, C. R. (2003). Enhancer Trapping Reveals Widespread Circadian Clock Transcriptional Control in Arabidopsis. *Plant Physiology*, *132*(2), 629–639. https://doi.org/10.1104/pp.021006

Quinlan, A. R., & Hall, I. M. (2010). BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, *26*(6), 841-842.

Wu, G., Anafi, R. C., Hughes, M. E., Kornacker, K., & Hogenesch, J. B. (2016). MetaCycle: An integrated R package to evaluate periodicity in large scale data. *Bioinformatics*, *32*(21), 3351–3353. https://doi.org/10.1093/bioinformatics/btw405