BIOINFORMATICS IN AGRICULTURE: IDENTIFYING DIFFERENTIALLY EXPRESSED GENES DUE TO ABIOTIC FACTORS IN COTTON USING MULTICLASS CLASSIFICATION

Dhwani Patel – x2020fvk@stfx.ca, St. Francis Xavier University

ABSTRACT

Agriculture is deeply affected by climate change in terms of yield and quality of crops. Bioinformatics has been contributing to the sustainable development of agriculture; it has helped understand the plant's genetic resources and its response to numerous factors. Due to the complexity of stress conditions, there is a crucial need to discover the mechanisms of genetic response against various stress factors[14] and study the patterns of the stress-responsive genome molecular mechanisms [5]. Keeping these responses in mind, crops that can stand stronger in harsh conditions could be reproduced with bioengineering. Previous works include microarray analysis to determine the cross-responsive genes[1-5,15] in plants under influence of biotic and abiotic factors. This study is an effort to perform multiclass classification on the expressed genes. Random Forest algorithm is used to classify differentially expressed genes (DEGs) in cotton influenced by abiotic factors like cold, acid, salinity, and drought using the GEO (Series GSE50770) dataset. The model performance was outlined by a confusion matrix. Future work will be focused on the efficient filtering of the least responsive genes, improving the accuracy of the model, identifying the differentially expressed genes, and deriving relevant conclusions that can contribute to the survival of agricultural crops on the planet.

INTRODUCTION

The adverse effect of the change in climate on agriculture, combined with population growth might result in a shortage of food. It is necessary to address this global issue of hunger and find ways to overcome it. Zero Hunger is one of the goals for the 2030 Agenda for Sustainable Development which was adopted by Canada along with 192 other UN member states in September 2015. The 1st International Electronic Conference on Agronomy in 2021 outlined the main challenge of the constant or decreasing yield of crops due to a lack of resources and rising costs. Hence, in the 2nd International Electronic Conference on Agronomy in 2022, they were interested in research contributing towards yield stability, use of renewable resources, efficient breeding, etc.

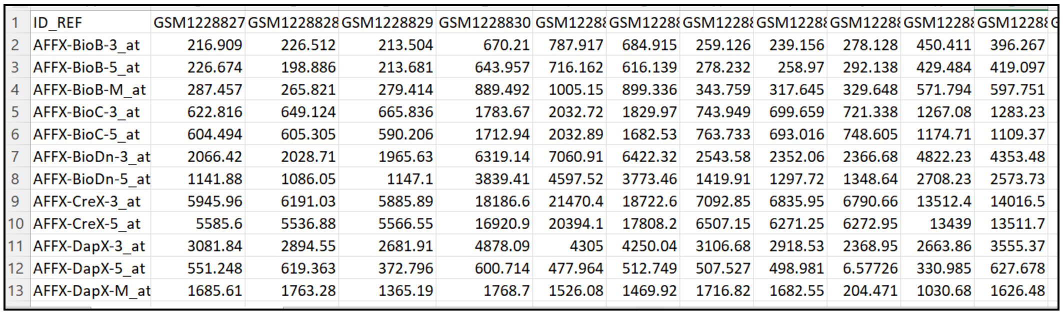
Bioinformatics has been contributing to understanding the issues mentioned above by aiming toward plant genetics. There have been studies of gene modification in crops due to environmental factors that describe the ability of agricultural crops to resist diseases and survive other obstacles. Observing changes at a genomic level like sequencing of microbial genomes can also help in the control levels of greenhouse gases, which plays a significant role in stabilizing climate change globally[3-4]. Bacillus thuringiensis (Bt) genes when transferred to crops like cotton, maize, and potatoes enhance their ability to resist insects, since the pests die when they try to feed on them, which results in less use of insecticides[2-4].

There has been ongoing research and reviews on how Machine Learning and Deep Learning, AI tools have become prominent in data preprocessing and interpretation[12]. These techniques help in modeling of stress responses in plants by observing gene expression, protein expression, genomic variation and metabolite biosynthesis. Literature revision states that there are many instances of supervised algorithms that are implemented to gain insights into the data resources. The method of feature extraction, selection of the output variables, and learning algorithms depend on the kind of data available[11]. Random Forests, Support Vector Machines, and Convolutional Neural Networks have been primarily used for identifying and classifying symptoms of stress or the stress response from image datasets[13]. These models were proved to be successful in portraying the model of stress response in plants.

Studies show that there is significant variation in genes and proteins in presence of biotic and abiotic factors. Biotic factors include insects, microbes, competitive species growing around, fungi, and animal grazing. Abiotic factors would be excess heat, wind, cold, acidity in the soil, salinity, and more. The microarray analysis technique is widely used to generate data by experimenting with DNA, RNA, and protein microarrays. Observing the gene expression in crops influenced by these factors using microarray analysis can be beneficial to recognize stress-responsive genes and crosstalk among them[1][16-20]. A paper from China shows how transcriptome analysis reveals Differentially Expressed Genes (DEGs) in cotton plants when exposed to different abiotic factors[1]. They found around 126 transcripts that were common in every experiment and they noticed that there was a crosstalk of responsive genes along with pathways to multiple biotic and abiotic stresses in cotton crops[1]. They also concluded that the rate of photosynthesis dropped and there was an increase in thiamin which might be developed to fight against abiotic stress[1]. Researching more in this area can help the agricultural field to get stronger crops, crops with greater nutritional value, reduce in using chemicals, and enhance the yield of specific crops with bioengineering. Their observations and conclusions were based on the microarray analysis, while this study aims to deduce similar conclusions using Machine Learning models.

MATERIALS AND METHODS

The GEO dataset (Series GSE50770) is used for multiclass classification in this paper. The dataset was formed by expression profiling and consists of transcripts that were differentially expressed in cotton seedlings under treatment of ABA (1µM Abscisic acid), cold (4°C), drought (200mM mannitol), salinity (200mM NaCl), and alkalinity (pH=11) [1],[6]. The experiment was conducted separately on seedlings 4-day after germination (DAG), and RNA was isolated from specimens of 10-day treatment. The experiment was repeated for around 24k genes three times for ABA, cold, drought, and salinity and two times for alkalinity. Thus, 17 samples were gathered for each gene[1],[6].



1	ID_REF	GSM1228827	GSM1228828	GSM1228829	GSM1228830	GSM1228	GSM1228	GSM1228	GSM1228	GSM1228	GSM1228	GSM1228	G
2	AFFX-BioB-3_at	216.909	226.512	213.504	670.21	787.917	684.915	259.126	239.156	278.128	450.411	396.267	
3	AFFX-BioB-5_at	226.674	198.886	213.681	643.957	716.162	616.139	278.232	258.97	292.138	429.484	419.097	
4	AFFX-BioB-M_at	287.457	265.821	279.414	889.492	1005.15	899.336	343.759	317.645	329.648	571.794	597.751	
5	AFFX-BioC-3_at	622.816	649.124	665.836	1783.67	2032.72	1829.97	743.949	699.659	721.338	1267.08	1283.23	
6	AFFX-BioC-5_at	604.494	605.305	590.206	1712.94	2032.89	1682.53	763.733	693.016	748.605	1174.71	1109.37	
7	AFFX-BioDn-3_at	2066.42	2028.71	1965.63	6319.14	7060.91	6422.32	2543.58	2352.06	2366.68	4822.23	4353.48	
8	AFFX-BioDn-5_at	1141.88	1086.05	1147.1	3839.41	4597.52	3773.46	1419.91	1297.72	1348.64	2708.23	2573.73	
9	AFFX-CreX-3_at	5945.96	6191.03	5885.89	18186.6	21470.4	18722.6	7092.85	6835.95	6790.66	13512.4	14016.5	
10	AFFX-CreX-5_at	5585.6	5536.88	5566.55	16920.9	20394.1	17808.2	6507.15	6271.25	6272.95	13439	13511.7	
11	AFFX-DapX-3_at	3081.84	2894.55	2681.91	4878.09	4305	4250.04	3106.68	2918.53	2368.95	2663.86	3555.37	
12	AFFX-DapX-5_at	551.248	619.363	372.796	600.714	477.964	512.749	507.527	498.981	6.57726	330.985	627.678	
13	AFFX-DapX-M_at	1685.61	1763.28	1365.19	1768.7	1526.08	1469.92	1716.82	1682.55	204.471	1030.68	1626.48	

Fig 1. Dataset

It is important to visualize the dataset to gain a few obvious insights. GEO2R is an interactive web tool that uses R packages and allows comparison among samples in GEO Series[6]. This helps in the identification and visualization of differentially expressed genes.

Data Preprocessing (Log Transformation)

The normalization of microarray data with various transformation techniques facilitates the comparison of gene expression data. Variance can be reduced with techniques like thresholding, scaling, and log transforming. Here, log2 transformation is used to measure the fold change.

Exploratory Data Analysis

GEO2R displays a table showing genes ordered by highest significance. A linear model is used to fit data and the empirical Bayes method is used to shrink the variance, resulting in ranking genes based on their expressional evidence. It also shows the expression level of a specific gene in each sample. Other visualization graphs include volcano plots, mean-variance trends, and density plots.

Here are a few figures that show the relationship between different groups of samples. The volcano plot shows the contrast between the two groups. This helps in identifying the significant genes mostly present at extremes of the plot. All the points in blue represent the significant genes, on the left side, the downregulated genes can be spotted while the upregulated genes are spotted on the right side.

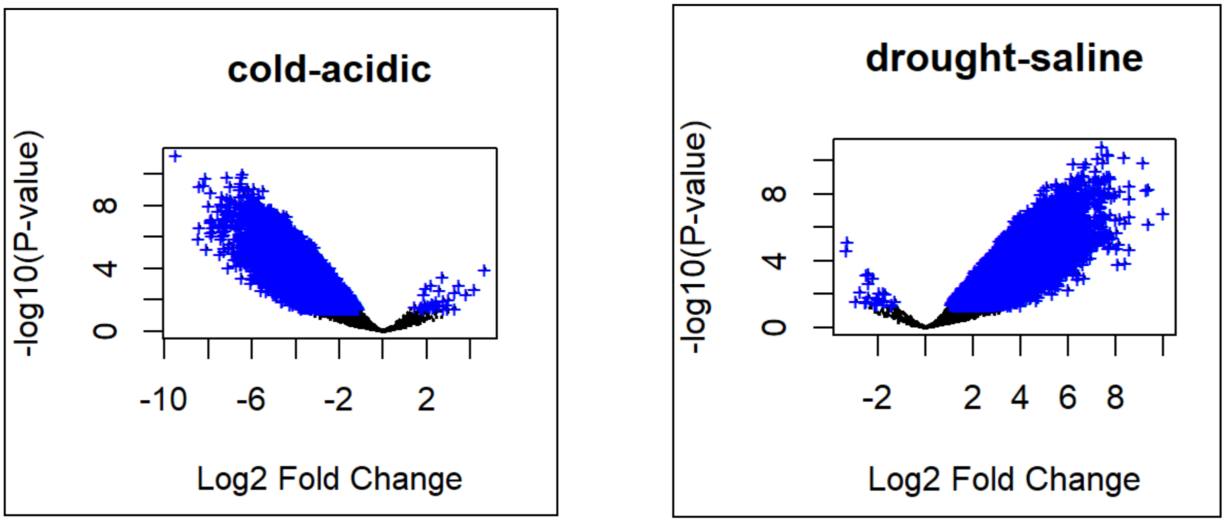


Fig 2. Volcano plots between different conditions

Following are the plots that compare all the six groups. The left image shows the boxplot which explains the distribution of expression values for each sample. The image on the right shows the density of expression values for multiple arrays.

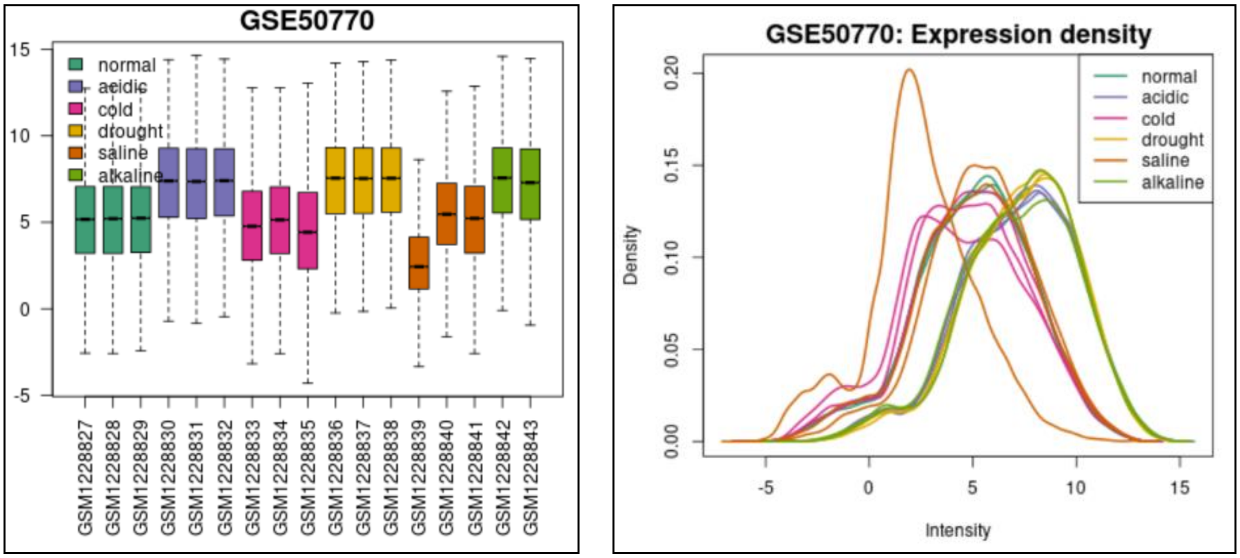
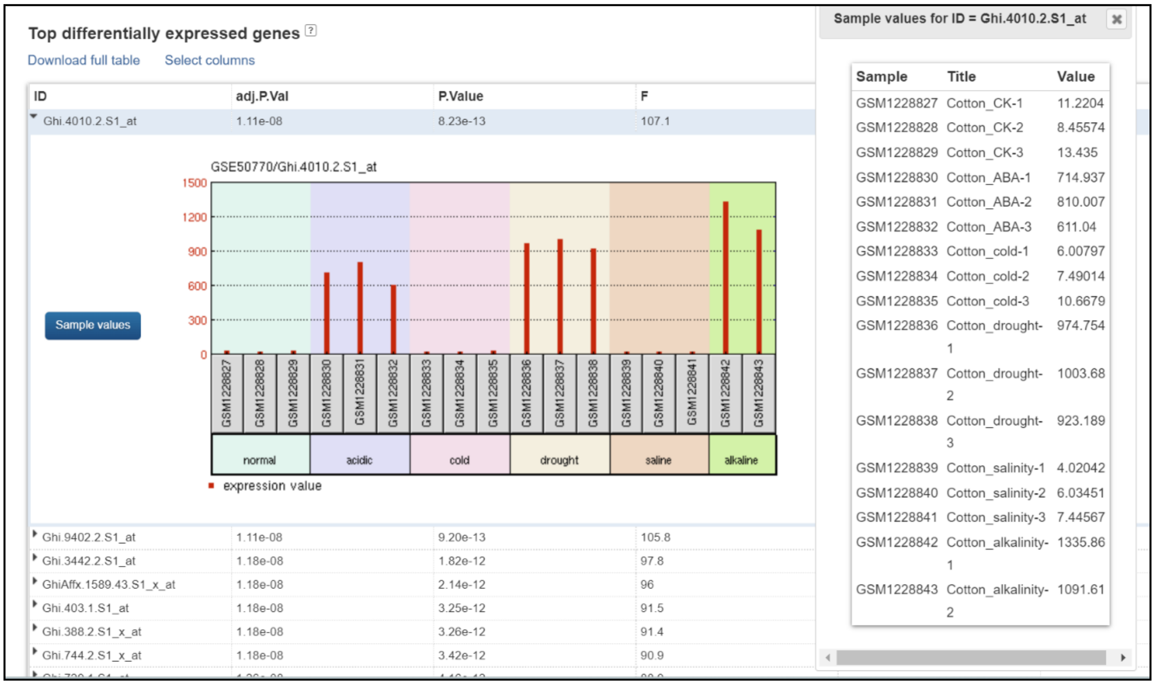


Fig 3. Distribution and density of gene expression

The table showing the most significant genes and details of gene expression values in each sample is shown below, which was generated using the GEO2R tool. It shows how the selected gene has more expression in acidic, drought, and alkaline conditions. This can help in finding genes that are common to all six groups and so their presence indicates harsh conditions and if the crop can be benefitted by induction of those genes.



Top differentially expressed genes

ID	adj.P.Val	P.Value	F
Ghi.4010.2.S1_at	1.11e-08	8.23e-13	107.1
Ghi.9402.2.S1_at	1.11e-08	9.20e-13	105.8
Ghi.3442.2.S1_at	1.18e-08	1.82e-12	97.8
GhiAffx.1589.43.S1_x_at	1.18e-08	2.14e-12	96
Ghi.403.1.S1_at	1.18e-08	3.25e-12	91.5
Ghi.388.2.S1_x_at	1.18e-08	3.26e-12	91.4
Ghi.744.2.S1_x_at	1.18e-08	3.42e-12	90.9

Sample values for ID = Ghi.4010.2.S1_at

Sample	Title	Value
GSM1228827	Cotton_CK-1	11.2204
GSM1228828	Cotton_CK-2	8.45574
GSM1228829	Cotton_CK-3	13.435
GSM1228830	Cotton_ABA-1	714.937
GSM1228831	Cotton_ABA-2	810.007
GSM1228832	Cotton_ABA-3	611.04
GSM1228833	Cotton_cold-1	6.00797
GSM1228834	Cotton_cold-2	7.49014
GSM1228835	Cotton_cold-3	10.6679
GSM1228836	Cotton_drought-1	974.754
GSM1228837	Cotton_drought-2	1003.68
GSM1228838	Cotton_drought-3	923.189
GSM1228839	Cotton_salinity-1	4.02042
GSM1228840	Cotton_salinity-2	6.03451
GSM1228841	Cotton_salinity-3	7.44567
GSM1228842	Cotton_alkalinity-1	1335.86
GSM1228843	Cotton_alkalinity-2	1091.61

Fig 4. Top differentially expressed genes based on P-value

Feature Selection (Filtering Genes)

Genefilter() is a function of the Bioconductor package that is used to filter genes from a given array using the filter functions provided. Genes were selected by specifying the lower and upper bounds of the coefficient of variance. The dimensions of the dataset were now reduced to around 4k genes from 24k genes. pOverA is a filter function that is used to filter based on the value of A, it returns TRUE if the proportion of the elements is larger than A.

Modeling

To classify the gene samples, a label vector was created manually assigning a class to each group. Since there are more than two classes, this problem presents itself as a multiclass classification problem. Various models were tried to get accuracy using a confusion matrix, with data being divided into 75% training and 25% testing.

In an effort to improve the evaluated models, leave one out cross-validation technique was applied. This was performed manually using a ‘for’ loop. Training and testing index were defined in a way that there were always 16 samples in training data and 1 sample in testing data. At each iteration, the training and testing data is changed and the models are trained accordingly.

Models used for classification –

- Random Forest Classifier
- Support Vector Machine (with Radial Kernel)
- Support Vector Machine (with Linear Kernel)
- Decision Tree

All the models were trained using default parameters, it was observed that Random Forest Classifier performed the best with 0.64 accuracy. Since Random Forest performed the best, I tried to improve the model's accuracy.

Hyperparameter Tuning

Optimizing the parameters of a model can result in an increase in accuracy. The process involves understanding data and finding values of parameters that work best to train the model for testing similar kind of data.

tuneRanger is a package that manages the automatic tuning of a random forest classifier to get the best out of the model[7]. The authors use model-based optimization as a tuning strategy and the three parameters min.node.size, sample.fraction, and mtry are tuned all at once.

The sample size parameter is specified to determine the number of observations pulled for the training of each tree's training. Its effect is like the mtry parameter. A decrease in the sample size results in diverse trees and thus lowering the correspondence among the trees, resulting in a positive effect on the accuracy calculation during aggregation of the trees. It has been shown that sampling when done with replacement = True, can result in slight bias when categorical variables with changeable categories are considered[8,9]. The performance of the model in such cases may get impaired due to replacement.

The nodesize parameter is used to specify the minimum number of observations in a terminal node. Its value decides the depth of the tree, lower values lead to deeper trees, meaning more splits till the terminal node is reached.

Mtry is one of the central parameters of random forests, which represents the number of randomly drawn candidate variables through which every split is selected when expanding a tree. Evaluation is based on Out-of-bag predictions which makes it much faster than other packages and tuning strategies. The package supports both classification and regression problems.

The value of number of trees can also be set in the model. It is not a parameter that is taken up for tuning, but it is suggested to be set sufficiently high for the betterment of the OOB curve.

There are various parameters that can be changed in the main function tuneRanger. First argument is the task, it is created using makeClassifTask of mlr. The measure to be optimised should be selected from a list that has specific measures selected from the listMeasures of mlr. Here, the Brier score is selected as a measure, since it is a measure used to compare true observed labels with predicted probabilities in multiclass classification tasks. The number of trees to be trained can be specified in num.trees, num.threads suggests the number of CPU threads used by the ranger. The number of iterations can be mentioned in iters. To manually specify a list of the tuned parameters, argument tune.parameters can be used.

The mean of the top best 5% of iterations is used to recommend the final hyperparameters via a list, which ends up training the random forest model.

Evaluation

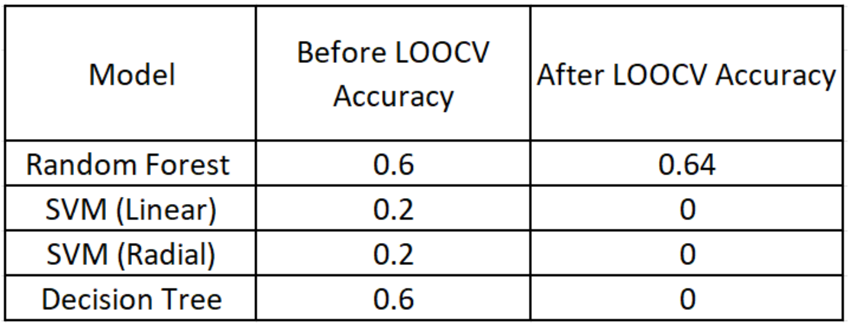
The performance of models was outlined using a confusion matrix table generated from predicted labels and actual labels, the diagonals of that table would suggest correct predictions. Thus, the formula used was, to add the values in diagonals and divide them by the total predictions.

RESULTS AND DISCUSSION

The evaluation of models shows that for this multiclass classification problem, Random Forest Classifier is the best choice with tuned parameters and leave one out cross-validation.

First, all models were trained on 75% of the data and tested on the rest 25%. These accuracies were recorded in the table. Later, leave one out cross-validation was implemented for each model and the new accuracies were quite different from the previous ones.

The table below shows the comparison of the accuracies of different models before and after leave one out cross-validation.



Model	Before LOOCV Accuracy	After LOOCV Accuracy
Random Forest	0.6	0.64
SVM (Linear)	0.2	0
SVM (Radial)	0.2	0
Decision Tree	0.6	0

Fig 5. Model accuracies with default parameters

It was observed that Random forest and Decision tree outperformed the SVM models. After LOOCV, the observed accuracy of SVM(L), SVM(R), and the Decision Tree was 0. This might be due to very less observations at hand.

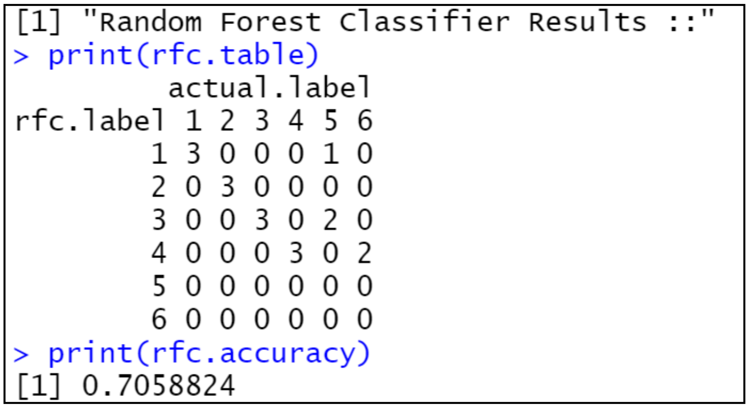
On the other hand, the Random Forest Classifier showed a slight improvement in terms of accuracy. Hence, I decided to work more and enhance the RF model. With hyperparameter tuning using tuneRanger, as discussed in the previous section, there was a hike in accuracy, and now the new accuracy recorded was 0.705.

Random Forest Classifier		
Before LOOCV	After LOOCV	LOOCV with Hyperparameter tuning
0.6	0.64	0.705

Fig 6. Accuracy after parameter tuning

Individual results of models after leave one out cross-validation is as follows –

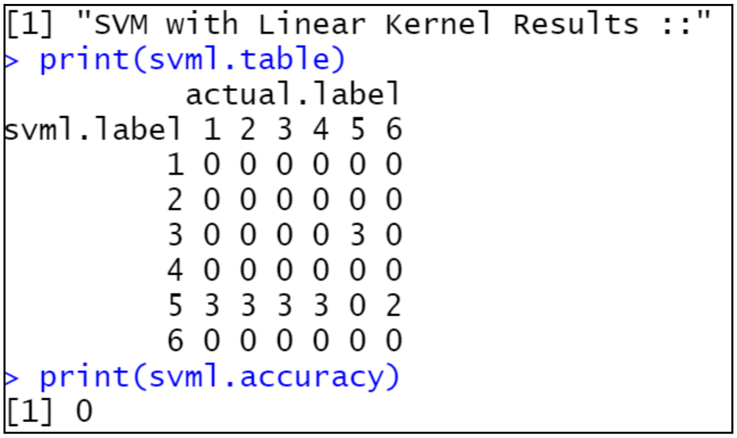
Random Forest Model



```
[1] "Random Forest Classifier Results ::"
> print(rfc.table)
         actual.label
rfc.label 1 2 3 4 5 6
        1 3 0 0 0 1 0
        2 0 3 0 0 0 0
        3 0 0 3 0 2 0
        4 0 0 0 3 0 2
        5 0 0 0 0 0 0
        6 0 0 0 0 0 0
> print(rfc.accuracy)
[1] 0.7058824
```

Fig 7. Confusion matrix and accuracy score for Random Forest

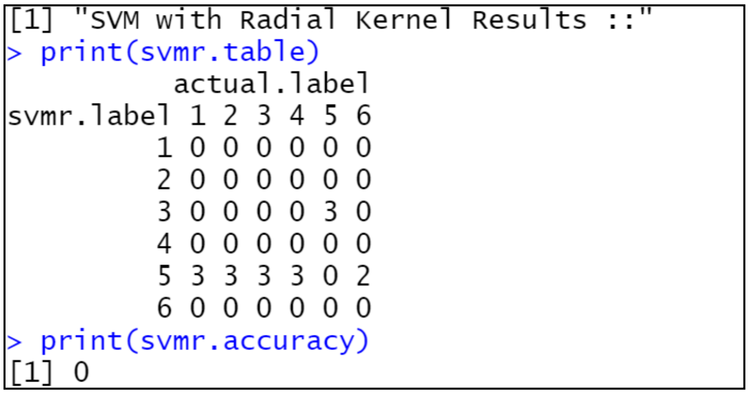
Support Vector Machine (Linear) Model



```
[1] "SVM with Linear Kernel Results ::"
> print(svml.table)
          actual.label
svml.label 1 2 3 4 5 6
         1 0 0 0 0 0 0
         2 0 0 0 0 0 0
         3 0 0 0 0 3 0
         4 0 0 0 0 0 0
         5 3 3 3 3 0 2
         6 0 0 0 0 0 0
> print(svml.accuracy)
[1] 0
```

Fig 8. Confusion matrix and accuracy score for SVM-Linear

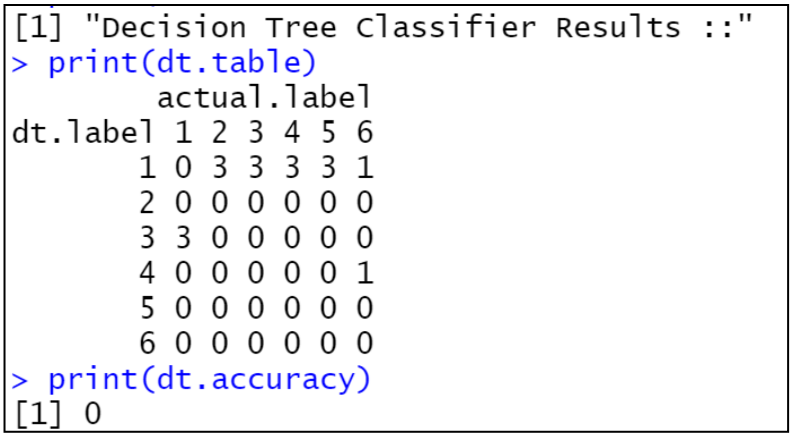
Support Vector Machine (Radial) Model



```
[1] "SVM with Radial Kernel Results ::"
> print(svmr.table)
          actual.label
svmr.label 1 2 3 4 5 6
         1 0 0 0 0 0 0
         2 0 0 0 0 0 0
         3 0 0 0 0 3 0
         4 0 0 0 0 0 0
         5 3 3 3 3 0 2
         6 0 0 0 0 0 0
> print(svmr.accuracy)
[1] 0
```

Fig 9. Confusion matrix and accuracy score for SVM-Radial

Decision Tree Model



```
[1] "Decision Tree Classifier Results ::"
> print(dt.table)
        actual.label
dt.label 1 2 3 4 5 6
       1 0 3 3 3 3 1
       2 0 0 0 0 0 0
       3 3 0 0 0 0 0
       4 0 0 0 0 0 1
       5 0 0 0 0 0 0
       6 0 0 0 0 0 0
> print(dt.accuracy)
[1] 0
```

Fig 10. Confusion matrix and accuracy score for Decision Tree

The performance of Random Forest was outlined using multiple ROC curves to compute the multi-class AUC values. pRoc function performs multi-class AUC as defined by Hand and Till (2001)[10]. A multiclass AUC is a mean of several auc. It can be shown as :–

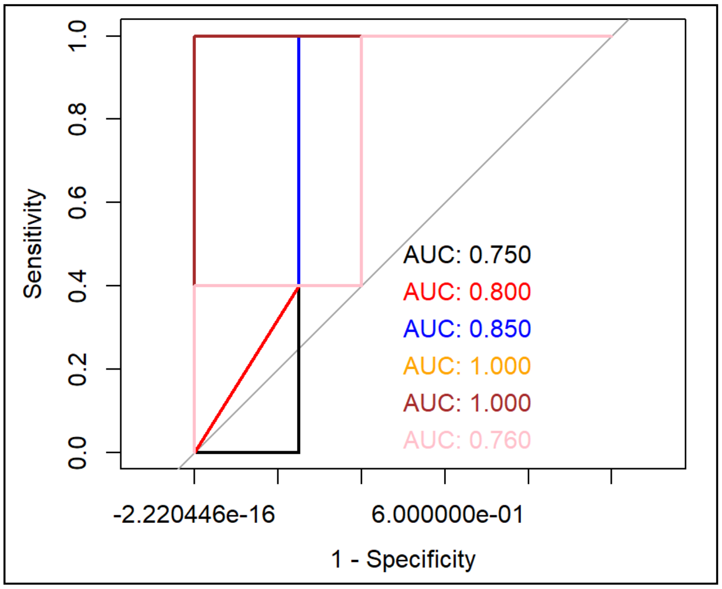


Fig 11. ROC plot with AUC values

CONCLUSION

Random Forest outperforms all the models by achieving accuracy of 70.5%. Other models did not perform well. The inferior performance of the models might be due to presence of very less samples per group. Those when divided for train and test, leaves with 16 samples for training and 1 sample for testing. It can be seen from results that hyperparameter tuning makes quite a difference in performance of model.

Future work would focus on gathering more data based on annotations and type of stress. In terms of models, more efforts would be made to perform loocv on different models like xgBoost, since without loocv, it gave similar results to the random forest model. Trying upsampling techniques to balance the data might enhance the model accuracy.

Identifying significant genes would be another step in the process. Selecting important features might help in understanding gene behaviour and it's contribution to each class can help researchers to develop techniques to induce or reduce the gene expression of identified genes. Classifying the differentially expressed genes would serve as a basis for other research grounded on identifying significant stress-responsive genes to make crops robust in the harsh climate, improve the nutritional value and consequently improve the quality and quantity of crop yield.

Code Availability

The final project code along with the ReadMe file is available on GitHub.

GitHub Link - https://github.com/dhwanipatel23/Bioinformatics-in-Agricultural-Crops

REFERENCES

1. Zhu Y-N, Shi D-Q, Ruan M-B, Zhang L-L, Meng Z-H, et al. (2013) Transcriptome Analysis Reveals Crosstalk of Responsive Genes to Multiple Abiotic Stresses in Cotton (Gossypium hirsutum L.). PLoS ONE 8(11): e80218. doi:10.1371/journal.pone.0080218
2. Zhou B, Zhang L, Ullah A, Jin X, Yang X, Zhang X. Identification of Multiple Stress Responsive Genes by Sequencing a Normalized cDNA Library from Sea-Land Cotton (Gossypium barbadense L.). PLoS One. 2016 Mar 31;11(3):e0152927. doi: 10.1371/journal.pone.0152927. PMID: 27031331; PMCID: PMC4816313.

3. SHARMA, Dr. (2021). Bioinformatics and its applications in environmental science and health and its applications in other disciplines.. 4. 88-93.
4. Ranjan, R. (2019). *APPLICATION OF BIOINFORMATIC TOOLS IN SUSTAINABLE AGRICULTURAL DEVELOPMENT*. www.semanticscholar.org. https://api.semanticscholar.org/CorpusID:215770966
5. Mishra, Divya & Shekhar, Shubhendu & Singh, Deepika & Chakraborty, Subhra & Chakraborty, Niranjan. (2018). Heat Shock Proteins and Abiotic Stress Tolerance in Plants. 10.1007/978-3-319-74715-6_3.
6. Dataset and GEO2R tool - https://www.ncbi.nlm.nih.gov/geo/info/geo2r.html
7. Probst, P. (2018, April 10). Hyperparameters and Tuning Strategies for Random Forest. arXiv.org. https://arxiv.org/abs/1804.03515
8. Janitza, S., Binder, H. and Boulesteix, A.-L. (2016) Pitfalls of hypothesis tests and model selection on bootstrap samples: causes and consequences in biometrical applications. Biometrical Journal, 58, 447–473.
9. Strobl, C., Boulesteix, A.-L., Zeileis, A. and Hothorn, T. (2007) Bias in random forest variable importance measures: Illustrations, sources and a solution. BMC Bioinformatics, 8, 25.
10. David J. Hand and Robert J. Till (2001). A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems. *Machine Learning* **45**(2), p. 171–186. DOI: doi: 10.1023/A:1010920819831.
11. Silva J.C.F., Teixeira R.M., Silva F.F., Brommonschenkel S.H., Fontes E.P. Machine Learning Approaches and Their Current Application in Plant Molecular Biology: A Systematic Review. Plant Sci. 2019;284:37–47. doi: 10.1016/j.plantsci.2019.03.020.
12. Rico-Chávez AK, Franco JA, Fernandez-Jaramillo AA, Contreras-Medina LM, Guevara-González RG, Hernandez-Escobedo Q. Machine Learning for Plant Stress Modeling: A Perspective towards Hormesis Management. Plants (Basel). 2022 Apr 2;11(7):970. doi: 10.3390/plants11070970. PMID: 35406950; PMCID: PMC9003083.
13. Moghimi A., Yang C., Marchetto P.M. Ensemble Feature Selection for Plant Phenotyping: A Journey from Hyperspectral to Multispectral Imaging. IEEE Access. 2018;6:56870–56884. doi: 10.1109/ACCESS.2018.2872801.
14. Seki M, Kamei A, Yamaguchi-Shinozaki K, Shinozaki K (2003) Molecular responses to drought, salinity and frost: common and different paths for plant protection. Curr Opin Biotechnol 14: 194-199. doi:https://doi.org/10.1016/S0958-1669(03)00030-2. PubMed: 12732320.
15. Chinnusamy V, Schumaker K, Zhu JK (2004) Molecular genetic perspectives on cross-talk and specificity in abiotic stress signalling in plants. J Exp Bot 55: 225-236. PubMed: 14673035.

16. Chaudhary B, Hovav R, Flagel L, Mittler R, Wendel JF (2009) Parallel expression evolution of oxidative stress-related genes in fiber from wild and domesticated diploid and polyploid cotton (Gossypium). BMC Genomics 10: 378. doi:https://doi.org/10.1186/1471-2164-10-378. PubMed: 19686594.
17. Christianson JA, Llewellyn DJ, Dennis ES, Wilson IW (2010) Global gene expression responses to waterlogging in roots and leaves of cotton (*Gossypium hirsutum*. p. L.). Plant Cell Physiol 51: 21-37.
18. Rodriguez-Uribe L, Higbie SM, Stewart JM, Wilkins T, Lindemann W et al. (2011) Identification of salt responsive genes using comparative microarray analysis in Upland cotton (*Gossypium hirsutum*. p. L.). Plant Sci 180: 461-469.
19. Yao D, Zhang X, Zhao X, Liu C, Wang C et al. (2011) Transcriptome analysis reveals salt-stress-regulated biological processes and key pathways in roots of cotton (*Gossypium hirsutum*. p. L.). Genomics 98: 47-55.
20. Padmalatha KV, Dhandapani G, Kanakachari M, Kumar S, Dass A et al. (2012) Genome-wide transcriptomic analysis of cotton under drought stress reveal significant down-regulation of genes and pathways involved in fibre elongation and up-regulation of defense responsive genes. Plant Mol Biol 78: 223-246.
21. Conference details - https://www.mdpi.com/journal/agriculture/events