Supervised ML for Honeybee Queen Treatments

Oscary Dotel  
Undergraduate Computer Science Capstone  
Capstone Advisor: Their Name Here  
SUNY Polytechnic Institute  
Spring 2024

# Introduction

With the assistance of GC-MS (Gas Chromatography Mass spectrometry), an instrument to identify the structure of molecular components was utilized to quantify key molecules typically present in Honeybee queen’s cuticle. These molecules consist of cuticular hydrocarbon (chc) and lipids or esters. These chc and lipids are responsible for crucial biological functionality within queen bees and are indicators to fertility and stressors. With a dataset reflective of five diets in relation to human impacts on nature; the goal was to construct a machine learning model that would be effective in identifying the diet/treatment of a queen bee. At first, the algorithm ADABOOST was used, and unsatisfactory accuracy shed some light on the issues of the dataset. Analyzing the data and its properties was crucial for success.

After a few trials and errors with different ML models, it was understood that the dataset was too meager in size for a classifier ML model. Augmentation of chemistry data was mandatory at this stage. Of course, the approach couldn’t be random and needed to be augmented through constraints that were sensible to the chemistry present in the species of honeybees. In essence, this project is a semi supervised learning model since the augmented data were mere constructs of tangible data. As technology evolves, we humans try to exploit its power to form introspection of our physical world and the effects we impose on it. The study of biology is no different in wanting to use this power. Some insights are difficult to see in the perspective of a microscope. Machine learning algorithms provide a magnification that is not only significant in founding sought out questions but revealing those questions we never asked. As a species, our evolutionary dominance has made the smaller parts of our world enigmatic but no matter how small, it doesn’t take away its role in a stable ecosystem.

# Related work

Cross-Validation techniques are a widely used method for ML for its flexible use of the dataset for less biased outputs. Different styles of cross-validation for every purpose such as Hold-Out Cross Validation that splits training and testing into ratios that would be more ideal for certain datasets [1]. See figure 1 for an illustration of HO CV

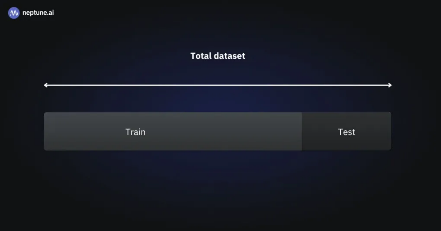
The particular use is dependent on the properties of the dataset and how conceptually certain models can be applied. For example, for my purpose, the variability of the dataset and size made it difficult for the trivial solution. Furthermore, Cross-Validation had been the suggested approach with Leave-One-Out [1] as the recommended technique. See figure 2 for an illustration of LOO CV. Training and providing a prediction on every iteration, Leave-One-Out trains itself on *n – 1* sample, n being the number of total samples. This essentially means that the prediction would be done on one sample from the dataset, providing less worry about the lack of samples during the training phase.

Figure 1. Hold-Out Cross-Validation [1]



Figure 2 Leave-One-Out Cross-Validation [1]

# Method

The experimental project was carried out in three stages. First, I exercised trivial approaches using ADABOOST in conjunction with decision tree classifier, and Gradient Boost for output comparison. Secondly, augmentation of the dataset with two fabrications of the dataset were constructed with different sizes. Lastly, utilization of Leave-One-Out Cross-validation on three datasets, original dataset, small, augmented data, and the large, augmented data for accuracy comparison.

3.1. Phase One – Trivial Approach

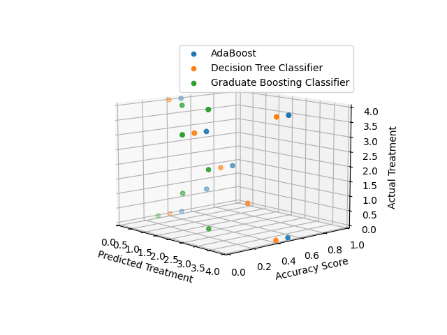
To maximize the weak classifiers within ADABOOST algorithm, I included the Decision Tree classifier with max depth of 24 to reflect the number of classifications. Since ADABOOST performs best with binary data and the dataset are floats, I include this to combat the variability within each classification. With one-hundred weak learners for both GradientBoost and ADABOOST. For a visual representation look at figure 3.

Figure 3. Matplotlib - 3D Accuracy of ADABOOST/GRADIENTBOOST/DECISIONTREE classifier.

Trivial approach with a ratio of 1:1 for the training/testing portion of this phase. Half the dataset is being fed into the different training models and the other half for predictions. As part of the identification process, a function designed to reshape the treatment group labels into integers for the sake of finding overlap would later be improved as the accuracy score of the model improved.

3.2. Phase Two – Augmentation

For more desirable results, augmentation of the data was necessary. The process of augmentation needed to be sensible to the chemical signature present within each treatment group. To combat the variability, standardizing the the data based on each treatment group. The reason for this is to avoid underfitting. A function that would take the totality of samples from one treatment group and then normalize the data. Furthermore, randomness came from subtracting or adding from the median of each chemical, and condition if the value reduced itself below zero to set it to zero as negative of any chemical signatures isn’t sensible. Cuticular hydrocarbon 21 for example is one of the chemical signatures and the presence of it in each treatment group is due to the treatment’s effects on their social harmony and maturing state. All these chemicals are associated with some functionality of cuticular or hormone communication. To keep these representative functionalities consistent within the augmentation this was the approach that was adopted. See figure 5&6 for reference of this concept of c21 in the original and augmented data.[3]. The first augmented data contains 140 samples, 40 of which are from the original data. The second augmented at contains 540 samples, 40 of which are from the original data as well.

A graph of different colored dots

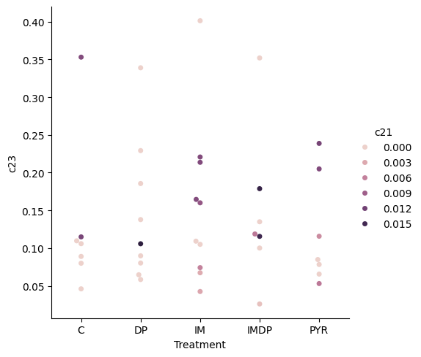
Description automatically generated with medium confidence

Figure 6. Cuticular Hydrocarbon 21 presence in Augmented Dataset

Figure .5. Cuticular Hydrocarbon 21 presence in Original Dataset

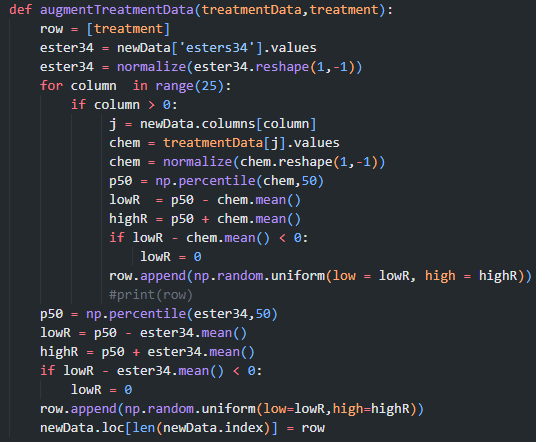


Figure 4. Augmentation Function.

3.3 Phase Three – Leave-One-Out Cross Validation

Finally, converting the treatment groups into matrix values to find overlap between the groups and Leave-One-Out CV to efficiently train our model. See figure 7 for reference on matrix reshape function.A screen shot of a computer program

Description automatically generated

Figure 7. Matrix Reshape Function

The accuracy was measured on multiple iteration of LOO. A function after training the models would append the accuracy of the current iteration being processed. Since LOOCV would only provide predictions on one sample data it was important to extract predictions on all the samples. See figure 8 for illustrations of LOOCV. [2] To establish contrast between augmented data and original data were included in this process. In addition, the learning rate of all the algorithms were constant throughout the iterations. With cross-validation, one can change the learning rate and create a neurol network of sorts.

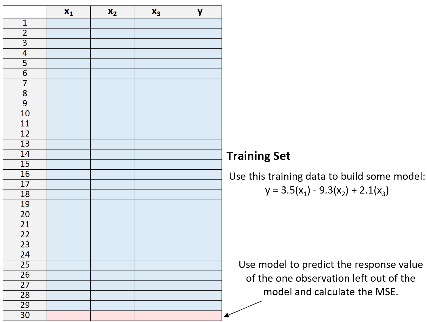


Figure 8. LOOCV [2]

# Results

The results of the project varied across methods and dataset used. The results from the trivial solution when using the original data were unsatisfactory with accuracy below <50%. When training on the augmented dataset. The outputs were satisfactory with an accuracy score reaching above >85% for the140 sample dataset and >90% for the 540 dataset. For the results are shown below in figures 9, 10, and 11.



Figure 9. Trivial Approach Original Data Accuracy Score



Figure 10. Trivial Approach Augmented Data 140 Samples Accuracy Score



Figure 11. Trivial Approach Augmented Dat 540 Samples Accuracy Score

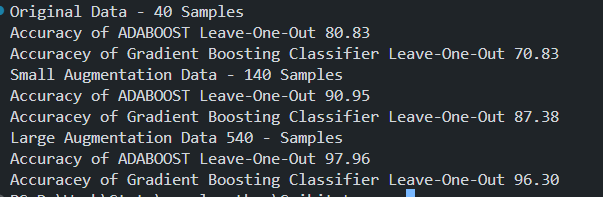
Finally, the results from LOOCV technique were much more promising than the trivial. Especially when it came to the original dataset.

Figure 12. LOOCV Accuracy Score.

# Discussion

Much of the work I did was trying to understand the capabilities of the dataset by modifying the ADABOOST and Gradient Boost algorithm parameters in different increments. What I found was that both algorithms produced exact or similar accuracy scores that were not satisfactory. Of course, this had to do with the lack of samples of the treatment groups but to ensure that wasn’t the case, I ran different experiments to see that the variability within each chemical classification didn’t skew the output. Still, that approach produced similar classifications predictions for the classes that I picked to test. In addition, training the model with only two treatment groups that are closely related in terms of physical conditions produced unsatisfactory predictions. Furthermore, it became apparent to me that with the techniques at play to have an outcome that would be successful – dissection of the chemistry was mandatory. At this stage of the project, I began to use other techniques such as Support Vector Classification (SVC)and Leave-One-Out (LOO). The idea for SVC is that it works well when the dataset has more dimensions than samples. Meaning it’d ideally perform better than previous models since the ratio of sample/classification was 40:26 and 40 of those samples should be broken down into 5 different categories reflective of the treatment groups. Leave-One-Out on the surface also provided optimism because it allowed the model to train on 96% of the dataset while providing predictions classifications on one sample. The interaction of the data with these different methods provided insight on the optimal approach which meant Leave-One-Out is the main gear for this operation. At last, using all these methods and finding little success with them led me to construct a program for the augmentation of the dataset.

# Conclusion and Future Work

Once augmentation was complete, the previously mentioned ML models had more than 500 samples to work with which gave a lot of flexibility. Ultimately, the accuracy score for all the models naturally went up but with some minor differences. I enjoyed seeing the accuracy score significantly increase also with more samples the visualization aspect gain insight on what roles these treatments have on the quantity of specific chemicals such as hydrocarbon 21. Further experimentation on the contrasting between the classifications and their quantities also would be fascinating to embark on. Since these classifications are tied to applications of social harmony within honeybee insects, it would be interesting to see how the evidence connects back to colony behavior. Perhaps making a neurol network that can backtrack its way to predicting the social dynamics of colony development since that data is already known.

# References

|  |  |
| --- | --- |
| [1]1] | V. Lyashenk and A. J. , "Neptune.ai," Ml Model Development, 12 April 2024. [Online]. Available: https://neptune.ai/blog/cross-validation-in-machine-learning-how-to-do-it-right. [Accessed 5 May 2024]. |
| [2] | Z. Bobbit, "Statology," A Quick Intro to Leave-One-Out Cross-Validation (LOOCV), 3 November 2020. [Online]. Available: https://www.statology.org/leave-one-out-cross-validation/. [Accessed 4 5 2024]. |
| [3] | C. L. Vernier, J. J. Krupp, K. Marcus, A. Hefetz, J. D. Levine and Y. Ben-Shahar, "National Library of Medicine," The cuticular hydrocarbon profiles of honey bee workers develop via a socially-modulated innate process, 5 February 2019. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6382352/#:~:text=As%20a%20bee%20ages%20it,the%20environment%20within%20the%20hive.. [Accessed 6 May 2024]. |

# Appendix

Python Dependencies include pandas, Scikit, Matplotlib, and lastly Seaborn.

Last two libraries were strictly for data visualization purposes and are not mandatory for execution.­

Method of installation used pip command for each dependency. (e.g “pip install scikit”)

Link to repository: [DOTEL0Y/napchc-Chemical-Signaling: Using Scikit-Learn - ADABOOST (github.com)](https://github.com/DOTEL0Y/napchc-Chemical-Signaling)