ReHoney-*Combing* Data and Models
A Forecast of U.S. Honey Production

Cody Fizette (cf2372), Ying Jin (yj1461), Shiqing Li (sl7085), Eidan Maimoni (emm792)
The Bees' Knees
New York University

I. Business Understanding

We are the analyst team of an agricultural company that produces honey under our apiary division. The executives are interested in predicting the future output of honey to better understand which states results in the highest honey production per bee colonies. We are currently focusing on honey production in quantity, based on the apiary status, as well as diseases and the natural conditions for the bees. In addition to predicting the future honey production of each state, we are interested in learning which variables that affect honey production the most. This topic is of concern to us as the majority of our crops, as well as the rest of the world, are pollinated by bees. With the results of the prediction, the company can maximize production and increase profits through better allocating their investments in Apiaries and to build long-term relationships within the state government.

There have been previous attempts and findings at predicting honey production. In the Pakistan Journal of Zoology, researchers Karadas and Kadırhanogullari in Turkey use a multilayer perceptron and data mining techniques such as CHAID to evaluate variables such as beekeeper age, education level, and bee race to determine honey yield. They concluded with the MARS algorithm and their features had the best predictive accuracy with Pearson Correlation Coefficient of 0.913. Our project is targeting on similar issue using different types of features, since we failed to acquire beekeepers' information.

II. Data Understanding

Our data was collected from two sources, the U.S. Department of Agriculture (USDA) and NOAA National Climatic Data Center. Data from USDA were downloaded as csv files. While data from NOAA was fetched using their Web Services API v2.

2.1 Data Collection

The United States Department of Agriculture records, aggregates, and reports yearly reviews of all the States quarterly activity of honey production. The yearly surveys contain files distinguishing the specific year and variables of production. After we download the csv files by passing a scrape function, we proceeded to clean the data by differentiating each line with different characters, each indicating the type of information the row represents. Description, headers, footers and the data itself were the bulk of the content of the files. We observed that the data consists of the following production features: Honey producing colonies, Yield per colony, Production, Stocks December 15, Average price per pound, and Value of production (descriptions in Appendix 6.2). Our target variable lies here in Yield per colony. We separated and placed into a dataframe along with the rest to be later attached to our data on weather and colony features.

The USDA reports also record bee colonies information by each state quarterly ranged from 2015 to the first two quarters of 2018. Once the csv files were downloaded and stored to local, they were passed to the honey_colonies function written to extract relevant information in a similar manner as honey production. Each csv file has 14 - 16 tables depending on the year, but only 8 gave us relevant information: the quarterly colony description tables and quarterly health stressors tables, while the rest of table gives an overall description of all states thus not contributing to our problem. Since the USDA honey production reports were documented in a yearly manner, the quarterly tables were summed and then averaged to ensure consistency. All the non-numerical values in the data were transformed to numerical values according to the documentation (e.g. a notation of (-) means 0, and (Z) means value less than 0.5). Lastly, each index and column was cleaned by removing indent or non-alphabetical characters to ensure consistency and to allow future concatenation.

For each year and state, we had the following colony features: Number of Colonies, Maximum Colonies, Lost Colonies, Percent Lost, Added Colonies, Renovated Colonies, Percent Renovated, Varroa Mites, Other Pests And Parasites, Diseases, Pesticides, Other Diseases and Unknown Diseases (descriptions in Appendix 6.2).

Our weather data was fetched from the Climate Data Online (CDO) Web Services API v2 provided by the National Oceanic and Atmospheric Administration (NOAA). Our original focus on weather data was by either season or year, so we fetched data from their Global Summary of the Month (GSOM) dataset for the convenience of later aggregating upon our need.

By passing in a set of search requirements to the GSOM database, the query returns all of the given type of data from all the measuring stations within the given range of location and time. Under the API restriction of maximum 1000 data points per quest, we built a function that quests data separately for each state and month, aggregates results from all measuring station by taking a average of all the values, then returns a pandas DataFrame with columns of each month and rows for each state. This API is unstable in a way that it would return NaN values irregularly. Therefore, we wrote the function in a way that identifies and fills all the NaN values in a dataframe. Running this function recursively until no more NaNs can be filled allowed us to avoid the API irregularity.

When fetching the weather data, we found that some features (DP05, PSUN, etc.) contain a significant amount of missing data, which were decided to be unusable for our purpose of analysis. For each state and month, we fetched the following weather features: TAVG, PRCP, DP10, SNOW, EVAP and AWND (descriptions in Appendix 6.2).

2.2 Data Cleaning

The only feature that contains a few missing data points was monthly evaporation. Judging from the fact that EVAP data did not vary much from year to year, we filled in the NaNs with the average EVAP from other years.

Since the honey production data were provided by year, we averaged 12-month weather data and summed up quarterly honey production data by year. Furthermore, we aggregated data from Alaska, Connecticut, Delaware, Maryland, Massachusetts, Nevada, New Hampshire, New Mexico, Oklahoma, and Rhode Island to map weather and bee colonies entries with the honey production entries. This was due to the fact that the data from those states were labeled under “Other States” by the USDA. Our dataset was left with 123 instances and 27 features.

III. Data Preparation & Analysis

3.1 Target Variable

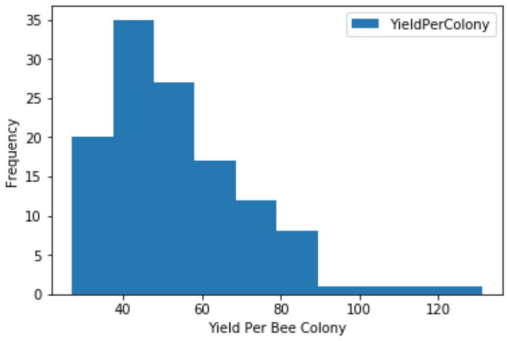


Figure 1: Distribution for yield per colony. It was then turned into four even categories.

For the purpose of having categorical results instead of numerical values on each US state's honey production rate, we decided to separate our numerical value into 4 groups. Looking at the target variable's distribution, we found that the state-wide Yield per Bee colony was approximately a right-skewed gaussian distribution with minimum, mean and maximum of 27, 53.9 and 131 pounds respectively. To ensure an evenly distributed categorical target variable, we separated the numerical values by quartiles using 25%, 50%, 75% of 40.5, 50, and 60.5. Labels for the four quartiles are 0, 1, 2, 3 with 3 being the highest class of honey yield.

3.2 Feature Engineering

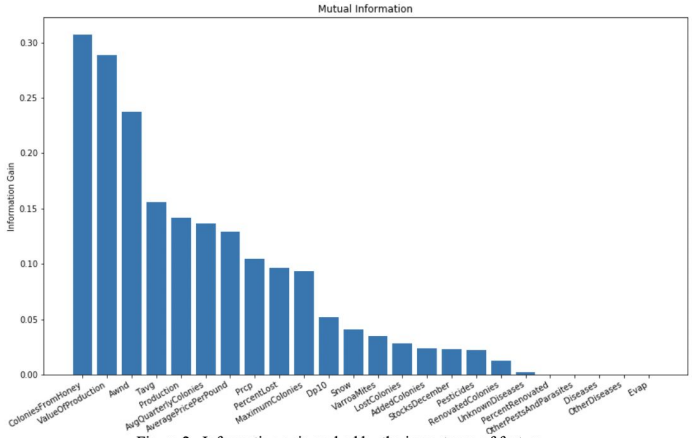


Figure 2. Information gain ranked by the importance of feature.

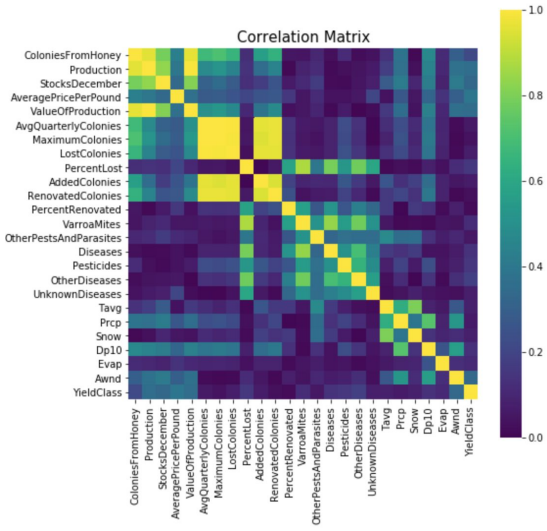
3.2.1 Mutual Information & Correlation

Given the initial complete dataset gathered from the CDO weather service API and USDA releases, we proceeded to feature selection aiming to clean the dataset by eliminating features that 1) were unrelated with targeted variables, 2) were highly correlated with high variances, and 3) would cause data leakage issues. First we implemented mutual information to rank the importance of each features. We found five features that have zero information gain towards the targeted variable. Meaning, adding those five features would not help us understanding the targeted variable. The five features were Percent Renovated, Other Pests and Parasites, Diseases, Other Disease, and Evaporation.

Next, we observed the correlation between each input features. If two features were highly correlated with each other, we could safely remove one of them without losing information and hurting the model accuracy. The

correlation matrix of all our features is presented in Figure 3 below. Features with correlation higher than 0.8 are presented in a reduced correlation matrix, shown in Figure 4.

From the reduced correlation graph, features including Avg Quarterly Colonies, Maximum Colonies, and Renovated Colonies were removed, since they were found to be highly correlated with three or more features on the topic of number of colonies. We found an interesting fact that Percent Lost was highly correlated with features on diseases, such as Varroa Mites, Diseases and Other Diseases, align with our domain knowledge. Moreover, all features on diseases were highly correlated with each other. We believed that keeping just one out of these four features would be enough to represent information on disease. Lastly, Dp10 and Snow were removed from our feature set as they were highly correlated with Prcp and Tavg.



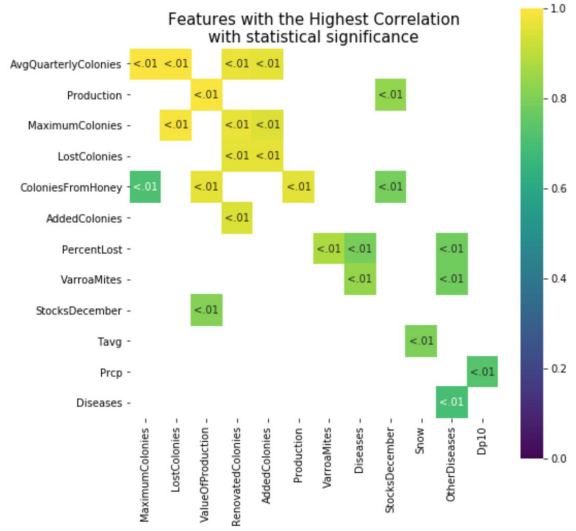


Figure 3: Correlation matrix for all features (left).
Figure 4: Reduced correlation matrix that includes only features with the highest correlation.

After moving features using MI and correlation, in addition to features that will cause data leakage, we were left with 10 features, Value Of Production, Maximum Colonies, Lost Colonies, Added Colonies, Pesticides, Other Diseases, Unknown Diseases, Tavg, Prcp, and Awnd.

3.3.3. Lasso

Lastly, we performed LASSO to compare the set of features derived from mutual information and correlation matrix. The constraint parameter was set to range from 0.1 to 0.8 to avoid convergence, and we found at 0.1 eight potential features to be selected at constraint of 0.1. Similarly, Production and Stocks December were eliminated to

avoid data leakage. With LASSO, we were left with a set of 6 features, Value Of Production, Other Diseases, Awnd, Average Price Per Pound, Lost Colonies, and Prcp. To some extent, features in this set confirms our results using mutual information and correlation.

The above two sets of selected features were then trained in our baseline model Naive Bayes accordingly and performances were compared to select the better feature set.

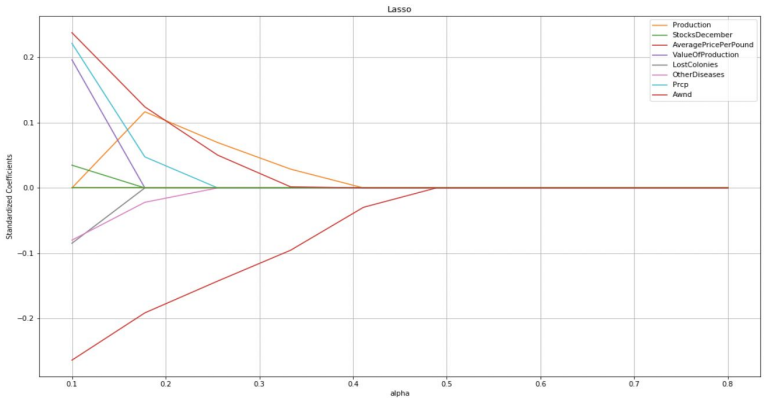


Figure 5. Standardized lasso coefficient, features with non-zero coefficient can be found in the legend.

IV. Modeling & Evaluation

We decided to use Naive Bayes as our baseline model. Upon having a baseline accuracy from the two sets of features, we applied Multinomial Logistic Regression, Support Vector Machine and finally a Random Forest Model using the better set of features. We used the 2015 & 2016 data as training, and 2017 data for testing, a 66/33 split.

4.1 Choice of Algorithms

4.1.1 Naive Bayes

A Gaussian Naive Bayes model was used to explore the performance of the two feature sets as well as a baseline model. The choice of Gaussian Naive Bayes as opposed to other forms of Naive Bayes models was a result of the strictly numerical features in our dataset. Naive Bayes was chosen for the baseline and feature-set exploration due to its simplicity both in concept and in hyperparameter options. This allowed us to quickly iterate over different options before moving on to more time-consuming models.

We began by creating two datasets containing the features chosen from LASSO, covariance, and mutual information analysis. Sklearn's GaussianNB model only has one parameter to tune, var_smoothing. This parameter performs regularization by adding a portion of the largest variance of all features to all the other features. A simple

search with 5-fold cross-validation was performed for var_smoothing values in the range of [1x10^{-15}, 1]. The value that maximized accuracy was chosen for further evaluation.

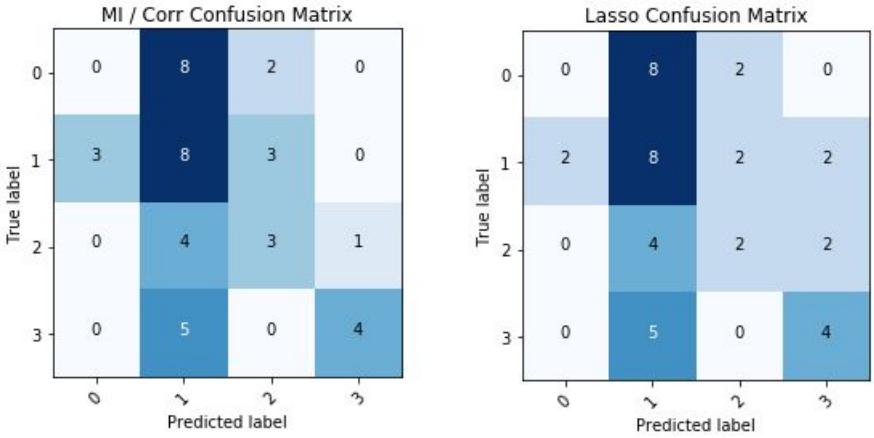


Figure 6: Confusion matrices for both feature sets explored. Titles refer to analysis used to choose feature sets.

Results for this experiment are shown in Figure 6. We can observe that the larger feature set from mutual information and correlation analysis was slightly more accurate than the other set. As a result of this analysis, we decided to train all further models on the larger feature set. Not only does it appear to make slightly better predictions, it is possible that more complex models can make better use of the larger number of features to make better predictions. As of this stage, our dataset was left with 123 instances and 10 features.

4.1.2 Multinomial Logistic Regression

We attempted a Multinomial Logistic Regression model as it is a generally simple and easy-to-interpret model. We first utilized gridsearch to optimize our model by finding the best solver, respective penalties, and constraining regularization parameter C. From the gridsearch, we determined that a Newton Conjugate Gradient method to nonlinear optimization is the best our solver.

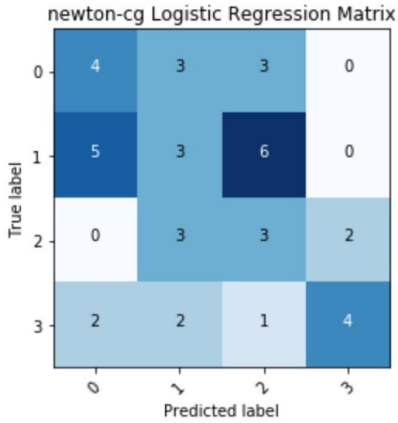


Figure 7: Confusion Matrix of Multinomial Logistic Regression

The grid search returns a relatively large C value with a ~65% training accuracy and ~34% testing accuracy. This unfortunately was poor, so we attempted to expand our parameter space as the model seems to be underfitting. From the result, we see that the accuracy still remained relatively low with a even higher C value. Therefore, we concluded that a logistic regression model was not ideal given our context. Any model adjustments without

modifying the preexisting data would have only increased accuracy by minimal margins. We proceeded to a more complex model to improve our score.

4.1.3 Support Vector Machine

SVM was selected because its versatility. It can accommodate both linear and nonlinear classification problems by setting the features space and kernel of choice. The support vector machine has a regularization constraint C as a parameter which allows the sample to set on the wrong side of hyperplane. The higher C was, the model is more tolerant to error. Lower the C, more sensitive the model is to the support vectors, as it wants to adjust the hyperplane by ensuring all support vectors lie on the correct side. Therefore, C was used to solve bias variance trade off problem encountered in the SVM model.

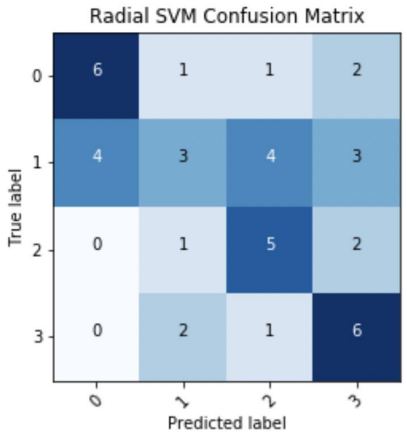


Figure 8. Confusion matrix of the best SVM model after regularization applied.

In our model, we first performed grid search to each kernel to gain understanding the behavior of the dataset. We found our data behaves like a nonlinear radial function as it yields the highest accuracy score. Then parameters C and gamma were passed into the grid search to find the optimum training accuracy, we found that the initial optimum pair of parameters gives the model a 1.00 training accuracy with ~0.5 testing accuracy, which indicated an overfitting pattern with low bias but high variance. Since parameter C from SVM regulates trade-off issue, we updated the model with a lower C, which lower the weight of the support vector acts on the model. By doing so we would expect introducing some bias to the model but hope to reduce the variance. The result showed that although both accuracy scores are lowered, but the training score dropped ~7% to 93%, while the testing accuracy only lowers to ~4.8%. Thus, we concluded that the tuned model had a slightly better performance, with a closer gap between training and testing score.

4.1.4 Random Forest Model

To avoid overfitting and to reduce variance, we decided to use a random forest model with 500 trees. Though random forest is known to be a black box algorithm which is generally harder to interpret than the other models, we are interested in comparing the result of RF with a more explainable model.

Using the random forest algorithm, a grid search was done on max_features, min_samples_leaf and max_depth. We experimented with max_features between 1 to 7, min_samples_leaf of 5 equally spaced values from 3 to half of the size of the training data, and max_depth from 5 to 10. Scoring method for the grid search was set to accuracy, since our target variable was multi-class. Results of the grid search gave out best parameter of max_depth=8, max_feature=5, and min_samples_leaf=3. The confusion matrix in Figure 9 shows that the random forest model was predicting our target variable fairly reasonable, as it never predicts 0 for class 2 or 3, and never predicts 3 for a class 0 or 1. The model results in a prediction accuracy of 51.22% and cross_entropy of 1.22. There were still issues with overfitting, since the model accuracy for training set was 98.78%.

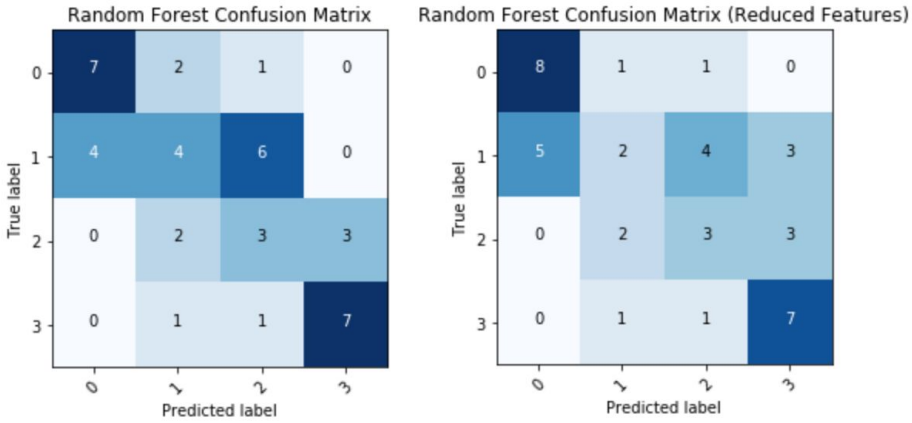


Figure 9: Confusion matrix from random forest model (left).
Figure 10: Confusion matrix from random forest after the features were reduced to the four features with the highest mutual information (right).

We attempted to further reduce overfitting by reducing the number of features based on feature importances. With only four most important features left, another grid search were done on the model, which resulted in the best model prediction accuracy of 48.78% and cross-entropy of 1.29. Though the testing accuracy was not as good as before, the reduced-feature model seems to be less overfitting, since it has a training accuracy of 85.36%. Both training and testing accuracies increased from baseline. Figure 10 shows the confusion matrix for using reduced feature.

4.2 Evaluation

Training and testing accuracies for all four models are in the table on the right. Accuracy was chosen for its ease of

	Training Accuracy	Testing Accuracy
Naive Bayes	0.415	0.366
Logistic Regression	0.659	0.341
SVM	0.939	0.488
Random Forest	0.854	0.488

interpretation and relevance to the business problem. Since this model will be used to assist business decisions, we want our audiences to gain a clear understanding of how likely our predictions are to be correct. Both SVM and RF models showed increase in performance compared to baseline. However, of these two models, RF showed less overfitting. We thus conclude this model to be the best and proceed to deploy it in production.

V. Deployment

Web scraping scripts will be periodically run in order to keep our dataset up to date. When new data arrives, the entire data pipeline will be executed including model fitting. In this way, the model will be kept as up to date and as accurate as possible. The plan is to then use the model to make predictions about each state's honey production at the beginning of each year. These predictions will be used to identify good investment opportunities. For monitoring, predictions will be compared to actual outcomes as well as returns on investment. As we are dealing with a very small data size, we wouldn't expect much deployment issue from scalability.

One important ethical consideration is if our model would reward unethical business practices. The goal of our model is to identify productive apiaries. Steps have to be taken to ensure that these apiaries are not resorting to unethical practices in order to increase efficiency. Prior to investment, an apiary should be thoroughly researched and can be achieved by onsite visits.

One major risk related to our model only measures the quantity, while the quality of honey certainly affects our profit as well. For instance, an apiary produces less in quantity but higher in quality honey may produce more outfit, which would not align with our prediction. To alleviate this, historical quality of production could be included into our feature spaces, although the quality might scored subjectively thus some bias is expected. Another risk is the decisions our company make will affect the following year's training data. For instance, states that initially see investment will tend to perform better thus leading to more investment. Eventually, previous levels of investment should be incorporated as a feature into the model.

VI. Reference

Karadas, Koksal & Kadirhanoğulları, Ibrahim. (2017). Predicting Honey Production using Data Mining and Artificial Neural Network Algorithms in Apiculture. Pakistan Journal of Zoology. 49. 1611-1619. 10.17582/journal.pjz/2017.49.5.1611.1619.

National Agricultural Statistics Service. "Honey Bee Colonies." **Albert R. Mann Library** Cornell University. Accessed November 8, 2018. https://usda.library.cornell.edu/concern/publications/rn301137d?locale=en.

National Agricultural Statistics Service. "Honey Production." **Albert R. Mann Library** Cornell University. Accessed November 8, 2018. https://usda.library.cornell.edu/concern/publications/f1881k888?locale=en.

National Centers for Environmental Information, and NCEI. "Climate Data Online: Web Services Documentation." Climate Data Online. Accessed November 12, 2018. https://www.ncdc.noaa.gov/cdo-web/webservices/v2.

National Centers for Environmental Information, and NCEI. "Climate Data Online: Dataset Discovery." Climate Data Online. Accessed November 12, 2018. https://www.ncdc.noaa.gov/cdo-web/datasets.

scikit-learn developers. "Documentation of scikit-learn 0.20.1." scikit-learn. Accessed November 30, 2018. https://scikit-learn.org/stable/documentation.html.

VII. Appendix

6.1 GitHub Links

Link to our code: https://github.com/emm792/DS-GA-1001-Project

6.2 List of All Features

- Honey producing colonies: The amount of colonies by the thousands per state
- Yield per colony: The ratio between Honey producing colonies and Production, in pounds
- Production: Amount produced by the thousands of pounds per state
- Stocks December 15: The amount of stocks held by the producers per the thousand pounds
- Average price per pound: Price in cents
- Value of production: Production times average price per pound by the thousands of dollars
- NumberOfColonies: The average number of colonies.
- MaximumColonies: The average number of colonies.
- LostColonies: The average number of colonies lost.
- PercentLost:The average percentage calculated from the number of lost colonies divided by the maximum colonies.
- AddedColonies: The average number of added colonies.
- RenovatedColonies: The average number of renovated colonie.
- PercentRenovated: The average percentage calculated from the number of renovated colonies divided by maximum colonies.
- VarroaMites: Average percentage of colonies affected by varroa mites.
- OtherPestsAndParasites: Average percentage of colonies affected by tracheal mitres, nosema, hive beetle, wax moths, and others.
- Diseases: Average percentage of colonies affected by diseases including American and European foulbrood, chalkbrood, stonebrood, paralysis, kashmir, deformed wing, sacbrood, isareli acute paralysis virus , lake sinai II.
- Pesticides: Average percentage of colonies affected by pesticides.
- Other: Average percentages of colonies affected by factors such as weather, starvation, insufficient forage, queen failure, hive damage/destroyed and others.
- Unknown: Average percentages of colonies affected by unstated factors.
- TAVG: Average temperature of the year in Fahrenheit
- PRCP: Average monthly precipitation amount in inches
- DP10: Average number of days in a month with precipitation greater or equal to 1.00 inch
- SNOW: Average monthly snowfall in inches
- EVAP: Average monthly evaporation in inches
- AWND: Average wind speed in miles per hours

6.3 Contributions

Cody Fizette: Scraped honey production data, Naive Bayes model, feature set performance analysis, performance reporting utilities.

Ying Jin: fetch, clean and aggregate weather data from NOAA API; turn target variable to categorical; feature selection using correlation; apply Random Forest model

Shiqing Li: Collect and clean honey colonies data from USDA reports; merge data together; feature selection using Mutual Information and Lasso; apply Support Vector Machine model.

Eidan Maimoni: Initial project idea. Finding pre-existing research solution to problem. Collect and clean honey production data from USDA reports. Ensure same format. Apply Logistic Regression model