Alexandra Schoenberg

Final Report:
Sakura Bloom Date and Ambient Temperature Analysis

Problem Statement

One of the most beautiful (and popular!) experiences for tourists in Japan and Japanese people alike is viewing the blooming sakura – in English, cherry blossom – flowers. Each spring, these flowers bloom for a relatively short period of time, and their season is beloved and celebrated within Japanese culture. Tourists come from far and wide specifically to see them, and Japanese families father for picnics in their local parks. The exact timing and duration of the sakura's peak blossom depends heavily upon the temperature patterns in the months leading up to early spring, and this project seeks to create a model that can accurately predict when a visitor should plan a trip and reasonably expect to see the flowers at their best for the year. The model aims to answer this question of "best" cherry blossom date for one notable park in Hirosaki City, Japan.

For this project, I used temperature data and flower blooming status for Hirosaki Park from January 1, 1997 to December 31, 2019, which I obtained from Kaggle. The temperature data is sourced from the Japanese Meteorological Agency, and sakura status data is sourced from the Hirosaki City Green Association.

Data Wrangling

The raw dataset from Kaggle contained 9131 rows with 3 columns. These columns were date, temperature, and flower status. Of the latter column, all but 75 values were null, while the former two columns were fully populated. The significance of this is that only 75 days out of the entire dataset of nearly 20 years were labeled as a "bloom" for the cherry blossoms.

Initially, the dataset looked pretty good. Although there were mostly null values in the blossom status column, I understood that to mean that those were days when the sakura were not blooming, and the values could easily be imputed in a subsequent step. My first action to wrangle the data to my specific needs was to separate the date column into three parts – day, month, and year – because in this use case I am looking for granularity of individual days within the blooming window. My second action was to encode and impute the flower status column. I created a for-loop to assign numerical values to each blooming status from the initial dataset: 0 for "bloom"; 1 for "full"; 2 for "scatter"; and 3 for anything else (which, in the case of this dataset, would be null values). Finally, I reduced the DataFrame down to encapsulate only the blooming months of April and May, resulting in a cleaned dataset with 5 columns and 1525 rows.

Exploratory Data Analysis

Although it is evident just from a glance at the magnitude of the dataset versus the magnitude of non-null values in the original blooming column, I wanted to determine what percentage of time during the blooming months we might see blossoming flowers.

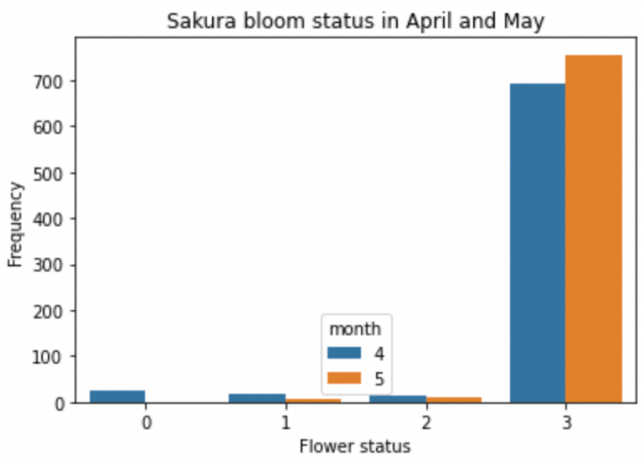

Figure 1: Count Plot of Bloom Status Frequencies

The created count plot confirms that, although the peak blooms are occurring exclusively during April and May in this park, even those months are comprised of mostly non-blooming days. The bloom period each year is quite short. Noting this, I sought to visualize the relationship between temperature and bloom status within these two blooming months.

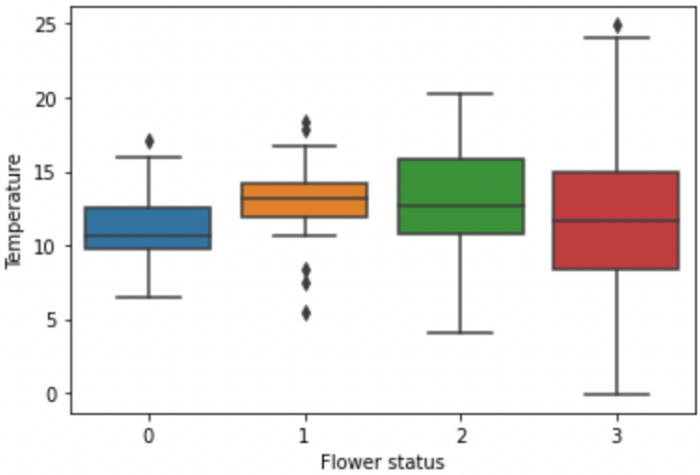

Figure 2: Box & Whisker Chart Temperature vs. Flower Status

From this figure, we can see that the median temperature of all four bloom statuses, from no flowers to full bloom, is relatively similar, clustered between roughly 10-13 degrees Celsius. However, the range of temperatures associated with the non-bloom status is significantly wider than the blooming statuses. If we include the outliers of the boxplot for

status 1, we can note that the temperature range tightens with each step towards the full bloom status. Ultimately, it seemed like temperature and flower status were not very well correlated in the dataset, and a correlation matrix (below) confirmed this assumption.

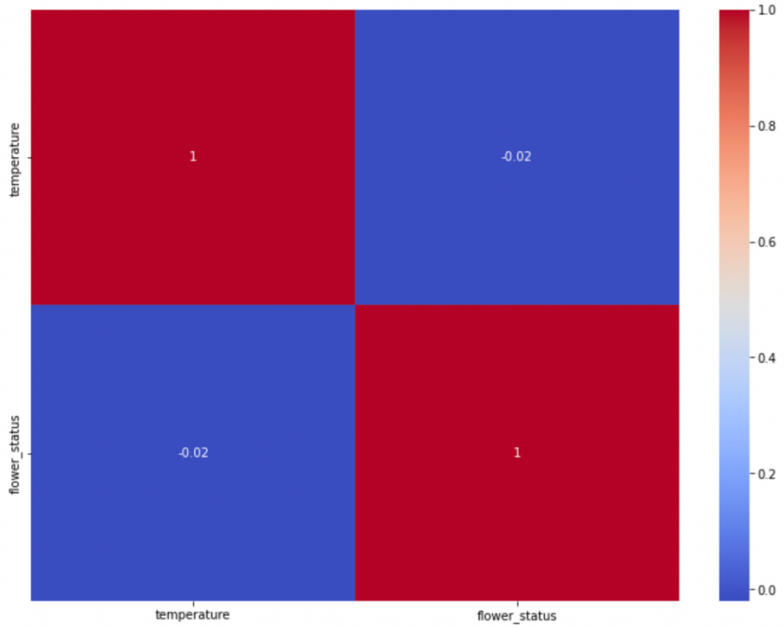

Figure 3: Sakura Bloom Feature Correlation Matrix for Blooming Months

As the dataset contains only bloom status and temperature data as features, the correlation matrix is limited. It is interesting to note that temperature and flower status are not very well correlated, and it opens the door to considering next steps and other data points that could be useful in future iterations of this project. It is a good reminder that weather and nature are complex and that, in the modeling step, we might find that a successful model would need more information and more features.

Modeling

For this classification project, I decided to use a Decision Tree Classifier model, with binary categories of all blooming statuses (0-2) combined and the non-bloom status (3). I applied a min-max scaler to the features of the model before initiating the classifier, in order to ensure that the features were magnitude standardized for analysis. In creating the model, I also performed principal component analysis and output a PCA scatter plot for each model iteration.

The first model was built using the unchanged imbalanced dataset, which was heavily biased towards the non-bloom category. The resulting PCA graph and classification report of this model clearly indicate the degree to which no-bloom dominates, with many more data points and much stronger modeling metrics than the bloom category.

From there, I created three more versions of the model, working to correct the category imbalance and improve the performance as much as possible. The second model oversampled the bloom values and resulted in slight improvements to the overall metrics. The third model undersampled the non-bloom values, resulting in decreased accuracy, precision, and F1 scores

and indicating a less effective model. Because the oversampled model remained superior to the undersampled one, I decided to expand upon it by applying the SMOTE method and checking the results. The SMOTE method, however, did not make improvements upon the oversampled model.

Ultimately, the best model from this project was the second iteration, which has oversampled the blooming data to try to correct the imbalance with non-blooming data. Because the blooming period is such a short segment of the year, there are too few "positive" data points available to build a well-fit model. The overall accuracy score of the final resultant model was 0.94. Precision, recall, and F1 scores for the non-bloom status were 0.96, 0.97, and 0.97 respectively, showing that the model is functioning well for the larger category. Precision, recall, and F1 scores for the bloom status were much less solid – 0.35, 0.32, and 0.33 respectively. More data is needed to improve these scores.

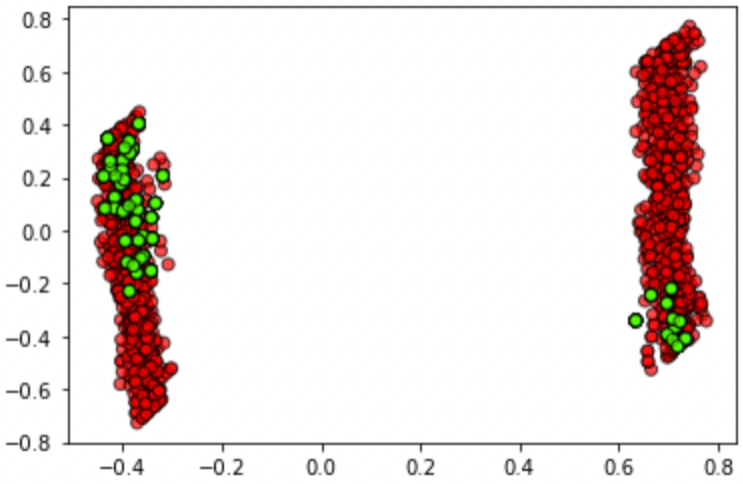

Figure 4: Principal Component Analysis of Best Model

The figure above shows the PCA for the best model. Red points are the non-bloom status, while green indicates the bloom. This model has been oversampled to increase the presence of green.

Takeaways and Future Steps

This project provided a lot of valuable practice in data wrangling, data analysis, and modeling, but also importantly highlighted the value of good data and the lesson that a perfect model might not exist to explain more complex phenomena in such simple terms as x-directly-correlated-to-y. As discussed above, there were simply not enough blooming data points to construct a truly predictive model based only on temperature. Additionally, as mentioned in the original project proposal, weather and nature are complicated and potentially unpredictable. This model takes into account only temperature, and only during the blooming months of April and May, as otherwise the already-imbalanced dataset would be much, much more drastically imbalanced. The flowers' exact blooming timeline could be affected by many more factors,

including but not limited to rainfall, wind, air quality, and even natural disasters, as Japan sits along the Pacific Rim and can be subject to earthquakes, tsunamis, and even volcanic activity.

A further expansion on this project could include many more years' worth of temperature and blossoming data, or otherwise add features such as air quality metrics, rainfall data, UV indices or number/quality of sunny days, etc. in order to flesh out the initial dataset and provide a more comprehensive picture of the natural phenomena occurring around and affecting the flowers.