Predicting Squirrel Friendliness from Behavior and Characteristics

Xiaoke Song
Data Science Initiative, Brown University
GitHub Repository[1]

1 Introduction

1.1 Motivation

Squirrels are widely admired for their endearing behaviors, making them a favorite among visitors to parks and campuses. However, their responses to human interaction can vary significantly—some may appear curious or friendly, while others may retreat or even behave aggressively. By studying their behaviors and characteristics of friendly squirrels, we can potentially make interactions with these fascinating creatures more enjoyable.

1.2 Central Park Squirrel Census - Squirrel Data

The dataset is derived from the 2018 Central Park Squirrel Census[8], which documented squirrel observations in New York City's Central Park. Collected by volunteers, the dataset includes 3023 data points, each representing an individual squirrel, making it an ideal resource for analyzing squirrel behavior.

1.3 Target Variable and Features

The dataset includes 31 features, with the boolean feature "Approaches" as the target variable. This feature indicates whether a squirrel was observed approaching a human. A value of 'True' identifies the squirrel as "Friendly," while a value of 'False' indicates it was "Unfriendly." This categorization naturally frames the problem as a binary classification task. The feature matrix captures squirrels' characteristics, such as fur color and age, and documents their behaviors, including both physical actions (e.g. chasing, foraging) and sounds made during observations (e.g. Moans, Kuks).

1.4 Current Research

Recent research on the Central Park Squirrel Census dataset explored squirrel friendliness prediction included geographical coordinates in their feature matrix with K-Means clustering technique. They also applied PCA, p-value filtering for feature selection. One study achieved 0.77 accuracy with Logistic Regression[4]; while another reported a 0.46 Precision-Recall AUC score[6], demonstrating the potential to enhance predictive performance.

2 Exploratory Data Analysis

2.1 Feature Analysis

In preparation for preprocessing, I examined the dataset for missing values and analyzed the data types of each feature. To explore potential relationships between the features and the target variable, I created various visualizations to identify correlations. Based on these analyses, I chose to drop features such as 'squirrel_ID' and the geographical coordinates of where squirrels were observed, as they were unlikely to contribute meaningful predictive power. Moreover, this project focuses on predicting squirrels' friendliness based on their characteristics and behaviors, excluding location-based features to emphasize intrinsic traits over environmental factors.

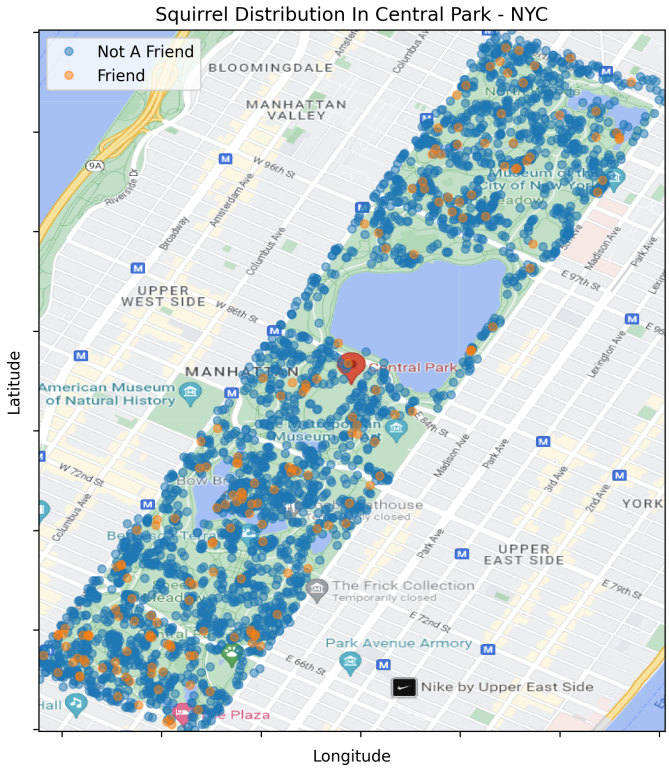

Figure 1: Squirrel distribution in Central Park: friendly squirrels (orange dots) are scattered without a clear pattern, indicating location lacks predictive power.

The stacked bar plots comparing squirrel friendliness by location and time of day suggest interesting behavioral trends. Squirrels found on the ground plane are more likely to be friendly compared to those above ground. Additionally, friendly squirrels are slightly more prevalent in the afternoon than in the morning, possibly due to foraging activity.

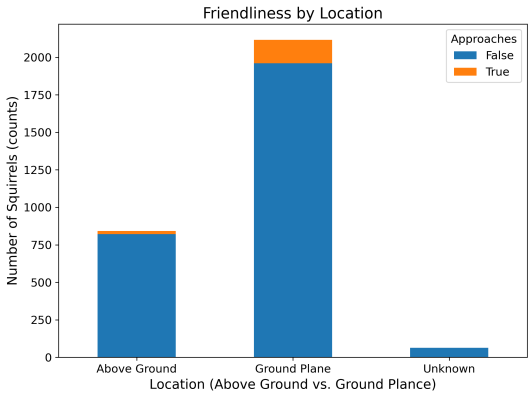

Figure 2: Squirrel distribution by location.

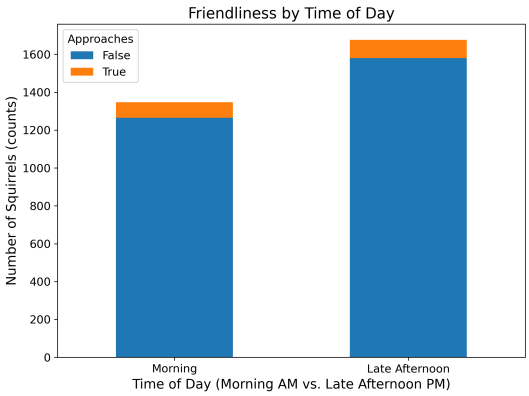

Figure 3: Squirrel distribution by time of day.

The bar plots compare squirrel friendliness based on primary fur color and vocal communication during observations. Figure 4 indicates that friendly squirrels are predominantly found among those with cinnamon or gray fur. From Figure 5, friendly squirrels were only observed making "kukking" sounds, a chirpy vocalization often used various purposes[8].

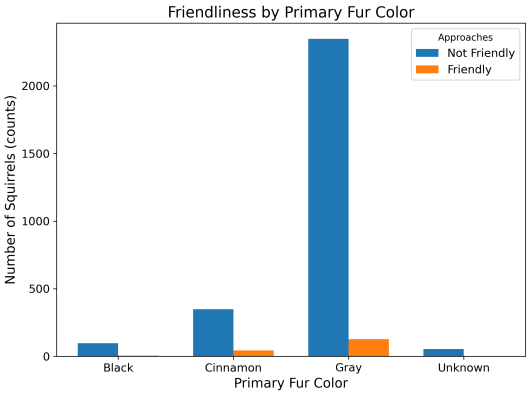

Figure 4: Squirrel distribution by primary fur color.

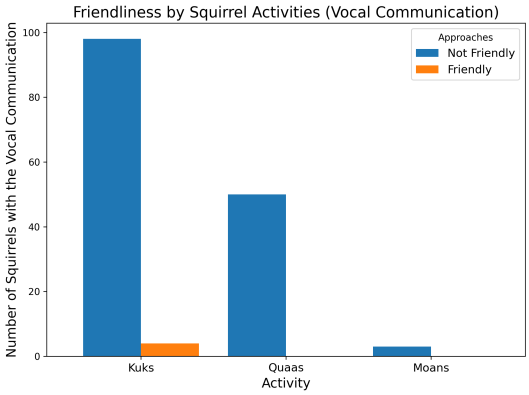

Figure 5: Squirrel distribution by observed vocal communication: "quaaing" indicates the presence of a ground predator such as dog, "moaning" indicates the presence of an air predator such as a hawk[8].

2.2 Target Variable

The target variable in this dataset is "Approaches", which classifies each squirrel as either Friendly ('Approaches = True') or Unfriendly ('Approaches = False'). The pie chart illustrates that unfriendly squirrels make up 94.1% of the population, stating the highly imbalanced nature of the dataset.

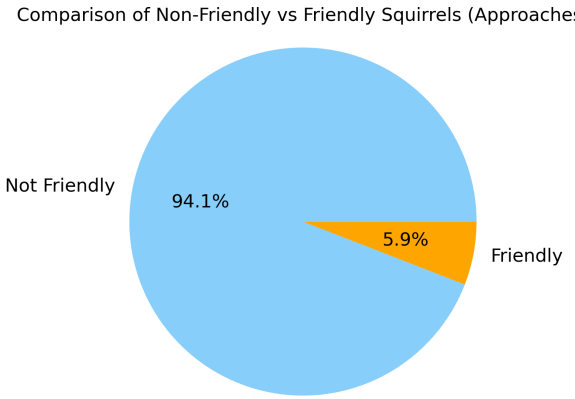

Figure 6: Balance of target variable.

3 Methods

I implemented four machine learning models—Logistic Regression, Random Forest, Support Vector Machine (SVM) and XGBoost—to address the classifiction problem. A key consideration in selecting these models was their ability to handle imbalanced data through 'class_weight' paramter (equivalent 'sample_weight' in XGBoost). This adjustment helps mitigate the challenges posed by the highly imbalanced target variable. Models like K-Nearest Neighbors (KNN) lack built-in mechanisms to address imbalanced data, making them less suitable for this task[2].

3.1 Data Splitting

For all models, the imbalanced dataset was initially split into training (include validation) and test sets using stratified 'train_test_split' with an 8 : 2 ratio. To further optimize model performance, StratifiedKFold with 4 folds was applied to the training and validation set during GridSearch. Using 4-fold stratified cross-validation strikes a balance between effectiveness and computational efficiency, particularly when working with a relatively small dataset.

3.2 Data Preprocessing

My feature matrix consisted of categorical and boolean features. All missing values were present in the categorical data. I applied OneHotEncoder to transform the categorical features, which simultaneously encoded the missing values as a separate category. To standardize the boolean features for better interpretability, I used StandardScaler. After preprocessing, the number of features expanded from 17 to 35, while maintaining the same number of data points: 3023.

3.3 Evaluation Metric

For this analysis, precision was selected as the evaluation metric to assess model performance. Minimizing false positives is critical in this context, as the most undesirable outcome would be the model incorrectly predicting a squirrel as friendly when, in reality, it is not. Such errors could lead to negative consequences, especially if one trusts the model's prediction and approaches an aggressive squirrel. Precision is particularly well-suited for this scenario, as it specifically focuses on reducing false positives and is effective for evaluating imbalanced dataset.

3.4 ML Pipeline

In my model selection, Logistic Regression was the only linear model, while Random Forest, SVM, and XGBoost were non-linear models. In addition to the uncertainty introduced by data splitting, models that rely on random sampling, such as Random Forest, also displayed variability due to their inherent non-deterministic nature. For each algorithm, hyperparameter tuning was performed to optimize performance, as detailed in Table 1.

Logistic Regression	penalty: [`'elasticnet'`] C: [0.01, 0.1, **1**, 10, 100] l1_ratio: [0, 0.25, **0.5**, 0.75, 1]
Random Forest	max_depth: [1, 3, **10**, 30, 100] max_features: [0.1, **0.3**, 0.5, 0.8, 1]
SVM	gamma: [0.01, **0.1**, 1, 10, 100] C: [0.01, 0.1, 1, **10**, 100]
XGBoost	reg_alpha: [1e0, **1e-2**, 1e-1, 1e1, 1e2] max_depth: [1, **3**, 10, 30, 100]

Table 1: Tuned hyperparameter values for each machine learning algorithm, with the bolded parameters representing the optimal choice for each model.

In Logistic Regression, the 'elasticnet' penalty combined with an 'l1_ratio' ranging from 0 to 1 (inclusive) allowed the incorporation of three regularization methods—Lasso, Ridge, and Elastic Net—within a single framework. When defining the hyperparameter ranges for all models, I followed the principle that the optimal parameter values for each fold should not fall at the

boundaries of the range, ensuring convergence within the selected hyperparameter space. Additionally, I included 'early_stopping_rounds' in XGBoost to prevent overfitting by halting training when performance on the validation set stopped improving.

To address uncertainties from non-deterministic models, training and testing were conducted across five different random states, with cross-validation applied to all models. At the end of each loop, the best-performing model and its corresponding test score were saved. The uncertainty due to data splitting was quantified as the standard deviation of the scores across the different random states, providing a measure of variability in the model's performance.

4 Results

The baseline precision for this study is 0.0589, calculated as the ratio of the positive class to the total population. Among the tested models, Random Forest achieved the highest mean precision score of 0.2149. However, it also exhibited the largest uncertainty, with a standard deviation of 0.0513, placing its performance approximately three standard deviations above the baseline. This level of variability is expected, given Random Forest's inherent non-deterministic properties, which contribute to its sensitivity to randomness.

Model	**Mean test score μ**	**Standard deviation σ**	**(μ − baseline) / σ**
Logistic Regression	0.1819	0.0252	4.8921
Random Forest	0.2149	0.0513	3.0378
SVM	0.1978	0.0299	4.6471
XGBoost	0.2089	0.0251	5.9722

Table 2: Comparison of models based on mean test scores, standard deviations, and normalized improvement over the baseline. Higher test scores indicate better precision, while larger standard deviations reflect greater variability due to uncertainty.

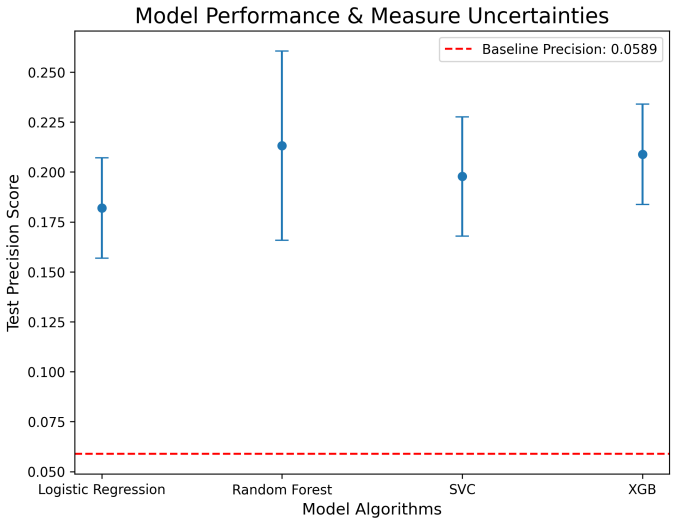

Figure 7: Comparison of model performance based on test precision scores and their uncertainties. All models achieve precision scores around 0.2, reflecting the impact of dataset imbalance on them to correctly predict the minority class.

The confusion matrix of the best-performing Random Forest model, shown in Figure 8, highlights the model's struggle to correctly classify friendly squirrels. This difficulty is evident as the model predominantly predicts the unfriendly class, reflecting the challenges posed by the imbalanced dataset.

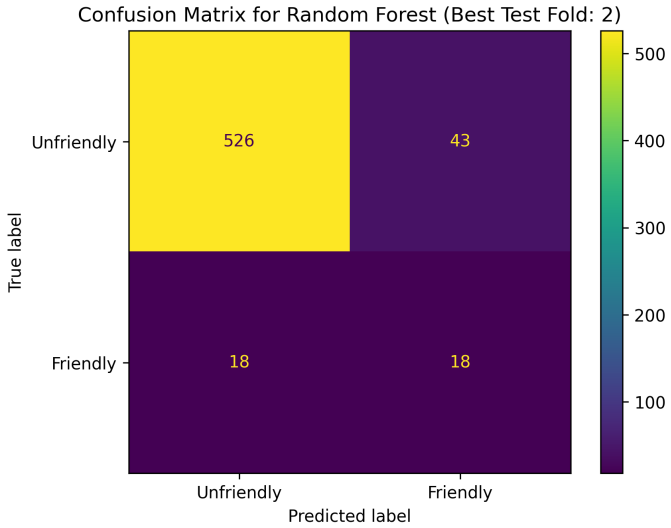

Figure 8: Test set confusion matrix using the best performing Random Forest model.

Permutation importance and Gini importance were used to evaluate feature importance at global level. Both methods identified the boolean feature "Indifferent" (if squirrel showed indifference to human presence[8]) and "Runs from" (whether squirrel ran from human, perceiving them as a threat[8]) as the top features.

"Location" (whether on the "Ground Plane" or "Above Ground"[8]) was significant in permutation importance but not in Gini importance, highlighting a key limitation of permutation importance, it does not account for feature interactions. For example, squirrels on the ground plane often exhibit behaviors like chasing or foraging, which may confound the importance attributed to "Location." Conversely, Gini importance struggles with correlated features, often favoring one while underestimating others, potentially introducing bias[7]. This results demonstrate the complementary strengths and limitation of the two methods.

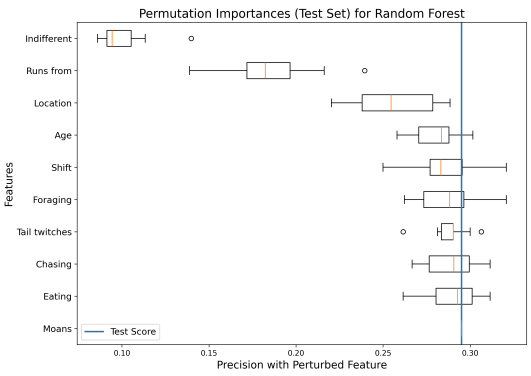

Figure 9: Top 10 Permutation Importance using Random Forest test set.

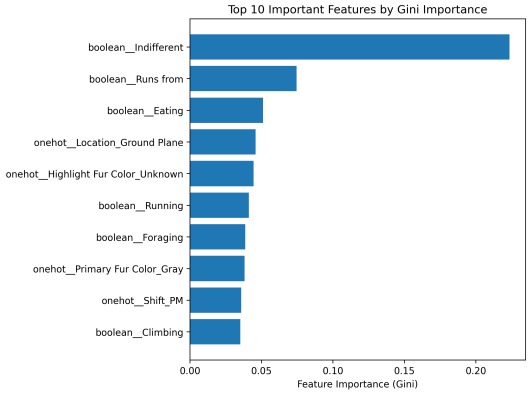

Figure 10: Top 10 Importance according to Gini importance method.

I also utilized a SHAP summary plot to better understand feature importance. As shown in

Figure 11, the boolean feature "Indifferent" is identified as the most influential feature, exhibiting a strong negative correlation with the target variable—higher values significantly decrease the likelihood of friendliness. This is equivalent to saying that squirrels that are not indifferent to human presence are less likely to perceive humans as a threat, making them more likely to be friendly. Other notable features, such as the categorical feature "Location_Ground Plane" and boolean feature "Runs from," align with the importance rankings observed using permutation importance and Gini importance methods, reinforcing their relevance in predicting squirrel friendliness.

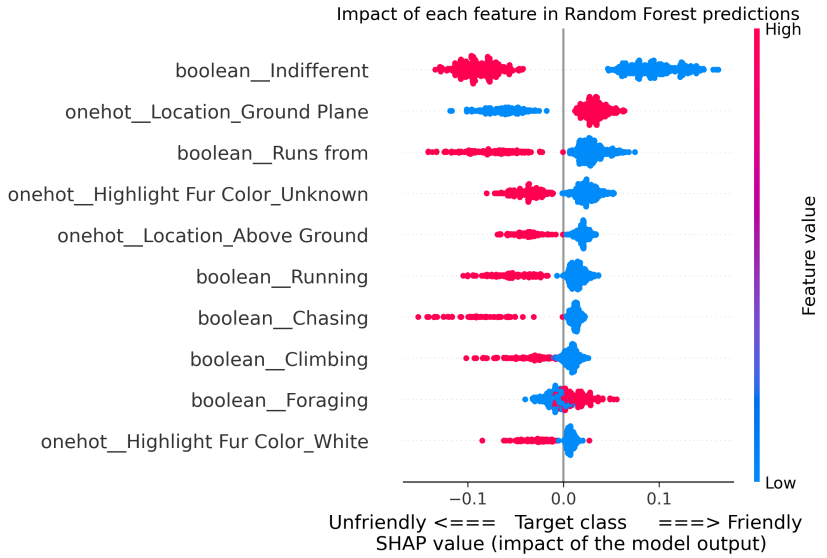

Figure 11: Impact of each feature in Random Forest prediction by SHAP summary plot.

To explore feature importance at the local level, I investigated SHAP contributions for each feature across each individual data point. The baseline probability of the Random Forest model predicting a data point as friendly is approximately 0.25. For data point 120, Figure 12, the predicted probability increased to 0.58, driven by features like "Indifferent," "Runs from," and "Location_Ground Plane." In contrast, for data point 478, Figure 13, the model confidently classified it as unfriendly. In addition to "Indifferent" and "Location_Ground Plane," the boolean feature "Chasing" played a key role in pushing the prediction toward the unfriendly class.

Overall, features identified as important at the global level, such as "Indifferent," "Runs from," and "Location_Ground Plane," also demonstrated strong influence at the local level. Additionally, features like "Chasing" showed their utility in specific instances, highlighting their potential as local predictors of squirrel's friendliness.

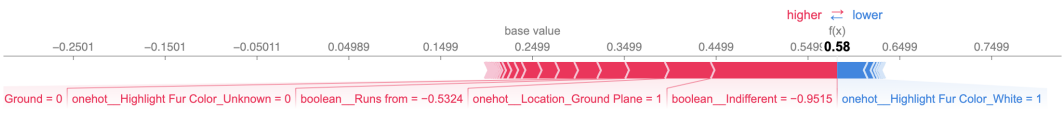

Figure 12: SHAP local value for index 120.

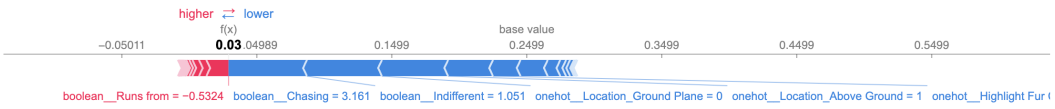

Figure 13: SHAP local value for index 478.

5 Outlook

The most significant challenge in this project is the issue of data imbalance, which causes models to predominantly predict the majority-class, unfriendly squirrels. In previous studies, one approach used random oversampling to increase the representation of the friendly class by duplicating data points[4]. Although this method improved model performance, it changed the original dataset structure, potentially introducing bias and reducing reliability for real-world predictions. Another approach, which I also implemented, involved adjusting the "class_weight" parameter in models[6]. This effectively reduced the tendency to predict only the majority class but came at the cost of sacrificing overall model performance.

Future improvements could include lowering the critical probability of default 0.5 to encourage more predictions of the friendly class.

Since Random Forest is a non-linear model, it does not provide coefficients that can be directly used to evaluate feature importance at global level. Instead, methods like permutation feature importance and Gini importance are commonly used. However, both fail to account for correlations between features, which can lead to biased or misleading interpretations. Similarly, while SHAP contributions offer valuable insight into feature importance at the local level, they can sometimes yield misleading results, especially in the presence of multicollinearity.

To overcome these challenges, it is essential to apply a broader range of interpretability techniques. Partial Dependence Plots (PDPs)[3] and Accumulated Local Effects (ALE) plots[5] are promising tools in this context. These methods address the limitations of traditional approaches by better handling correlated features and providing more reliable interpretations. Incorporating a wider range of interpretation techniques in future work would allow for a deeper understanding of feature importance on the model's predictions.

References

[1] Github repository:. https://github.com/xiaoke-song/1030project_squirrels.git.

[2] Md Eusha Kadir, Pritom Saha Akash, Sadia Sharmin, Amin Ahsan Ali, and Mohammad Shoyaib. A proximity weighted evidential k-nearest neighbor classifier for imbalanced data. *Published under PMCID*, 2020.

[3] Scikit learn Developers. Partial dependence plots (pdps). Scikit-learn Documentation.

[4] Bryan McMaster. A data science story: Squirrel friend. Medium Blog, 2023.

[5] Christoph Molnar. Accumulated local effects (ale) plots. Interpretable ML Book.

[6] Victor Murcia. Squirrelml: Predicting squirrel approach in nyc's central park. Medium Blog, 2023.

[7] Piotr Płoński. The 3 ways to compute feature importance in the random forest. *Towards Data Science*, 2020.

[8] The Squirrel Census Team. 2018 central park squirrel census - squirrel data. https://www.thesquirrelcensus.com/, 2023.