Michael Morales

IST 652

June 8, 2019

**Final Project**

**Data and Domain**

In a nutshell, hiking is walking in nature. More specifically, it is walking on unpaved trails in wilderness or natural preserve areas. For some it is a half mile in nature. For others, it is a six-month trek spanning 14 states along the Appalachian Trail. A select few will expand their horizons even farther by traversing the country again on the Pacific Crest Trail. Even fewer will go on to conquer The Continental Divide Trail, earning the Triple Crown of Hiking. Whether hiking is an hour once a month or a months-long battle against the elements, there is no shortage of participants, as evidenced by an abundance of websites and mobile applications devoted to the sport.

Data for this project can be found at Kaggle. It is a collection of GPS recorded hikes from the European hiking website hikr.org. hikr.org is a platform where registered users can post reports, photos, descriptions, and thoughts about their hikes. Some of the posts contain a GPX location file recorded as the activity took place. The data started as a csv file of 12,000 GPX files scraped from hikr.org in the spring of 2018. The GPX tracks were simplified with gpxpy (prior to posting to Kaggle) to reduce the size of the dataset, and to add other columns such as maximum and minimum elevation, also using gpxpy. The dataset and related kernels can be found at:

<https://www.kaggle.com/roccoli/gpx-hike-tracks>

**Data Preparation and Goal**

**Packages**

Since this project uses prediction models, there were several packages required, both for cleaning and organizing the data and for running the prediction algorithms. The packages were imported as follows:

import json

import nltk

import urllib

import datetime as dt

import pandas as pd

import nltk

from datetime import timedelta

import seaborn as sns

import apyori

from apyori import apriori

From the sklearn group of packages, the following utilities were imported:

KNeighborsClassifier

preprocessing

metrics

cross\_validate

cross\_val\_score

train\_test\_split

GridSearchCV

KFold

GaussianNB

roc\_curve

precision\_recall\_curve

auc

make\_scorer

recall\_score

accuracy\_score

precision\_score

confusion\_matrix

LinearSVC

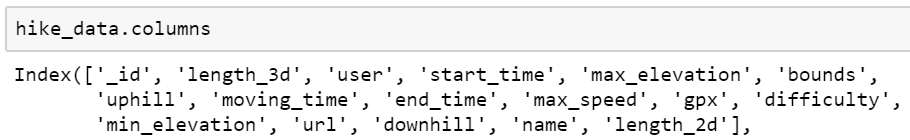
unique\_labels

LogisticRegression

linear\_model

**Data and Descriptions**

The data came in as a csv file with 17 columns.



The ‘difficulty’ column consists of 6 levels of difficulty on the SAC scale for hikes. The Swiss Alpine Club (SAC) established the SAC Hiking Scale in 2002 by publishing the guidelines in “Die Alpen” magazine. The scale provides simple, concise definitions that are easy to understand, and the hike requirements related to the difficulty.

|  |  |  |  |
| --- | --- | --- | --- |
| **SAC Rating** | **Type** | **Path/Terrain** | **Requirements** |
| **T1** | Hiking | Path well beaten and marked, terrain flat or moderately inclined, no danger of falling. | None. Suitable also for sport shoes. Orienting without problem, in principle also without a map. |
| **T2** | Mountain hiking | A continuous path, generally marked, terrain in sections steep, danger of falling not excluded. | Requires a safe step. Trekking shoes recommended. Basic orientation skills required. |
| **T3** | Challenging hiking | Trail not necessarily visible, exposed passages can be protected with cables, to maintain equilibrium one eventually needs hands, usually marked, a danger of falling on exposed passages, gravel slopes, pathless rock slopes. | A very safe step. Good trekking shoes. Orientation skills required constantly. Elementary alpine experiences. |
| **T4** | Alpine hiking | Trail not present, on some places hands are needed to advance, terrain already quite exposed, tricky grassy slopes, steep rocky slopes, easy snow slopes or bare glacier passages. | Experiences with exposed terrain. Stable trekking shoes. Ability of terrain assessment. Good orientation abilities. Alpine experiences. If weather deteriorates, escape can become difficult. |
| **T5** | Sophisticated alpine hiking | Pathless, exposed and difficult terrain, on some places easy climbing sections, steep scramble terrain, snow fields or bare glacier passages where there's danger of sliding. | Mountaineering shoes. Reliable assessment of terrain. Very good orientation abilities. A lot of alpine experiences. Easy climbing skills. Elementary skills of handling with rope and ice pick. |
| **T6** | Difficult alpine hiking | Pathless, not marked, very exposed and difficult terrain. Climbing passages up to UIAA degree II. Tricky, steep rocky terrain, glacier with a higher danger of sliding. | Excellent orientation abilities, mature alpine experiences and mastering of handling with alpine gear. |

The remainder of the columns with hike-specific information are all potentially related to the difficulty since they record such information as the hike’s maximum elevation, distance, the start and end times, the speed of the hiker, and the amount of uphill and downhill.

**Goal**

Data preparation cannot be completed without a goal, so it was crucial to define the intent of the project before moving forward with cleaning and transformation. The goal of the project is to use K-nearest neighbors (KNN), Naïve Bayes, multilinear regression, and Random Forest to attempt to predict hike difficulty by considering the information potentially related to the SAC rating. For instance, a high total hike time (which could be calculated by subtracting end time from start time), could be a predictor of the difficulty, since presumably a difficult hike would take longer to complete. The goal was considered as further data cleaning, transformation, derivation, and organization were completed.

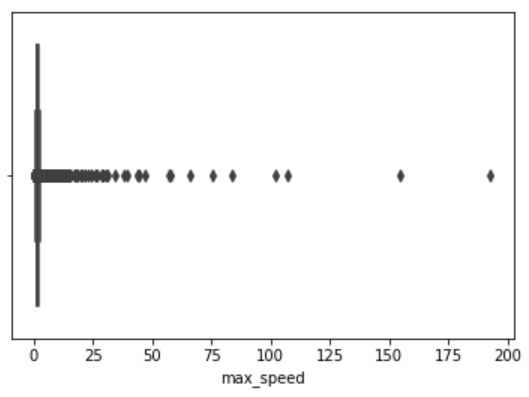
**Cleaning and Transformation**

Michael Morales wrote code to select only the columns needed for the prediction. There were two lengths included in the data set: length\_3d and length\_2d. Michael chose to eliminate length\_2d since it is the measured distance in meters on a two-dimensional map. The project goal was dependent on the actual distance of the hike in three-dimensional space. Other columns such as username and the geolocation codes were not helpful in fulfilling the project mission. Michael then chose to drop any rows with NAs in the time column since time is an important potential predictor, and the size of the dataset allowed for drops. Michael then recast the times as datetime objects, dropped all but the T designation in the SAC Rating column since a data dictionary was readily available online (see table above), and discretized the difficulty value.

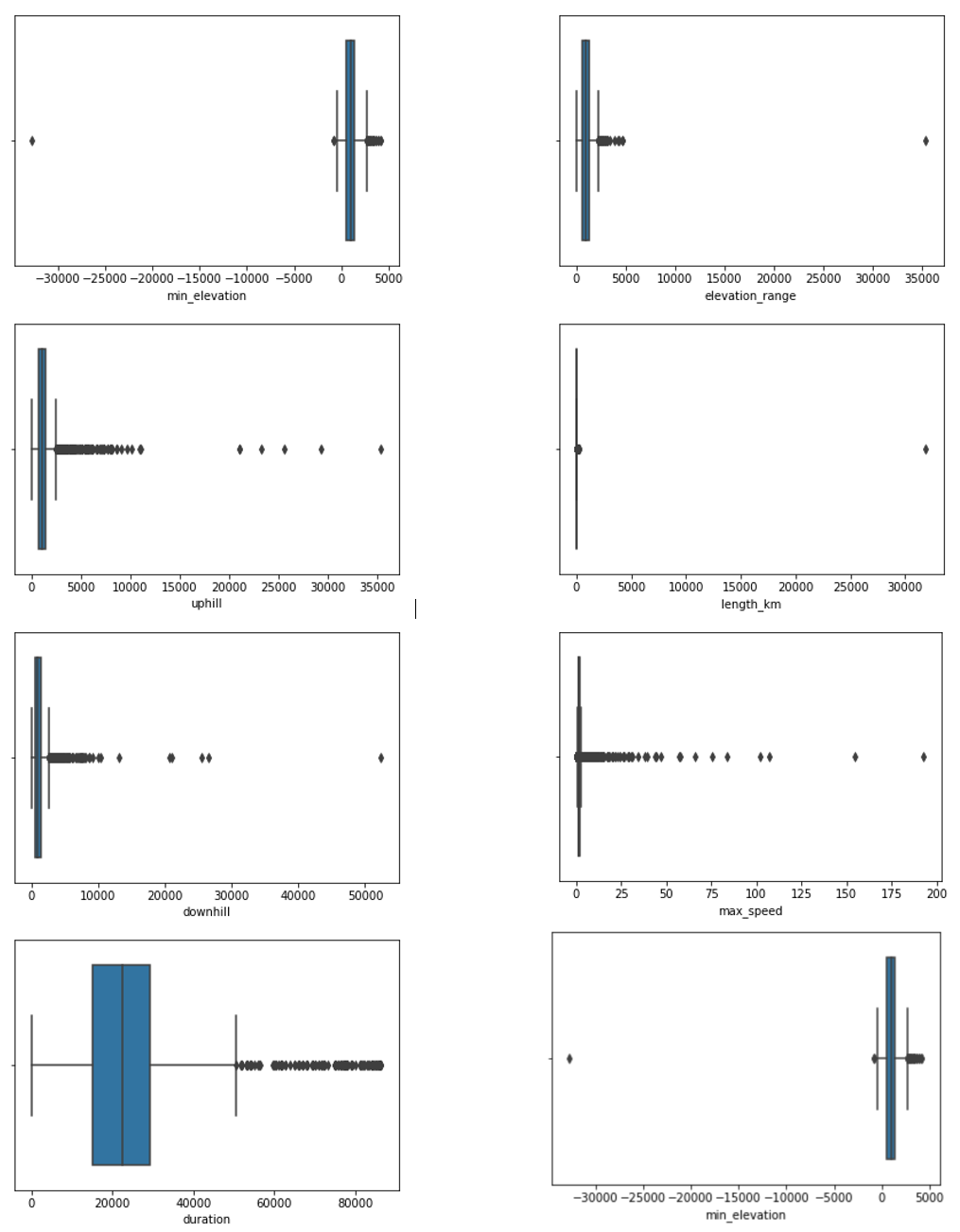
A prediction required that some values be derived from existing columns. Michael Morales calculated duration, the range of elevation (both uphill and downhill changes) in the hike, and the total length in kilometers. Afterward, he removed any remaining rows with null values and dropped duplicates to ensure purity in the data set. After the changes and drops were complete, the data set had 7,346 rows of 12 columns remaining.

**Outliers**

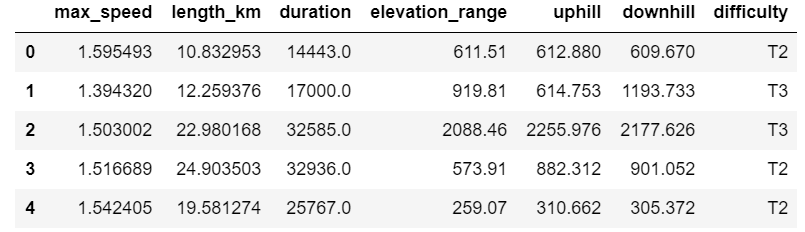
Michael’s next step was to look for outliers in the data. This was an important step because there was no need to keep activities that were potentially not hikes. It seemed that some of the rows may have been activities such as biking or skiing, since the maximum speed was nearly 200 meters per second, as seen in the boxplot below. Accordingly, any speeds higher than 150 meters per second were removed.



Michael also looked at boxplots for the elevations, elevation range, uphill, downhill, duration, and length in kilometers, and found outliers in all of them.

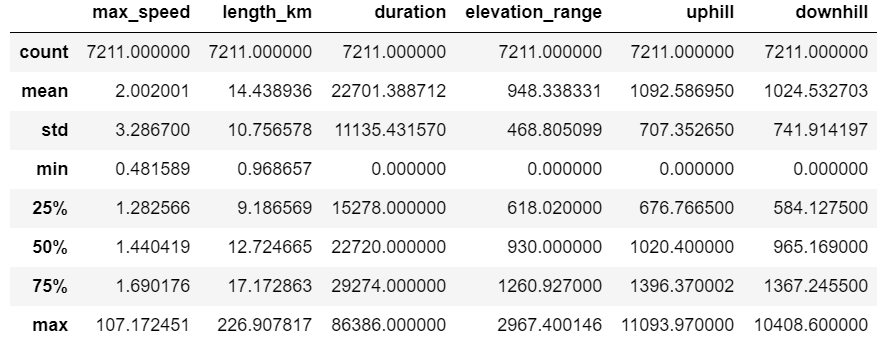


As such, any values beyond the limitations of human ability and earth’s maximum limits were dropped. Examples are a minimum elevation beyond -30,000 meters, and an elevation range of more than 35,000 meters. These values were deemed to be unrealistic and would clearly have a negative impact on the prediction models. Michael dropped values above 300 km for length, greater than 3,000 meters of elevation range, any max speeds equaling zero, and any downhill or uphill greater than 15,000 meters. This left 7,211 rows in the data, which was sufficient for performing the evaluation. The final columns for use in the evaluation were as seen in the figure below.



**Summary Statistics**

As a quick sanity check before running the prediction models, Jack Harris described the data to see the measures of center, quartiles, and minimum and maximum values.

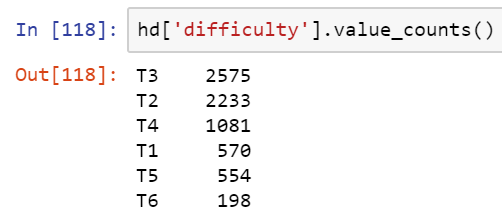


The values suggested that the data had been sufficiently cleaned, structured, and transformed, and that outliers and nulls had been successfully scrubbed.

**Data Splits and Prediction Models**

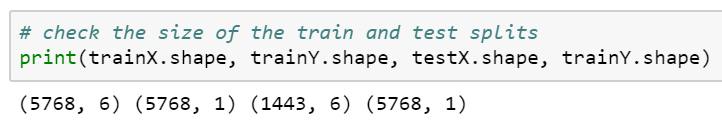
**Data Splits**

Jack proceeded to check the balance of the prediction target by performing unique value counts.



As evidenced by the figure above, the predictor is unbalanced, which could have a negative impact on the ability of the models to make accurate predictions.

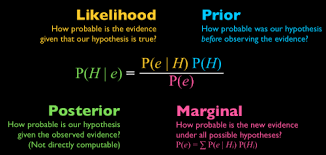
The next step was to split the data into test and train sets. After segregating the target variable, difficulty, Jack split the data into 80% train and 20% test sets, then verified the sizes.



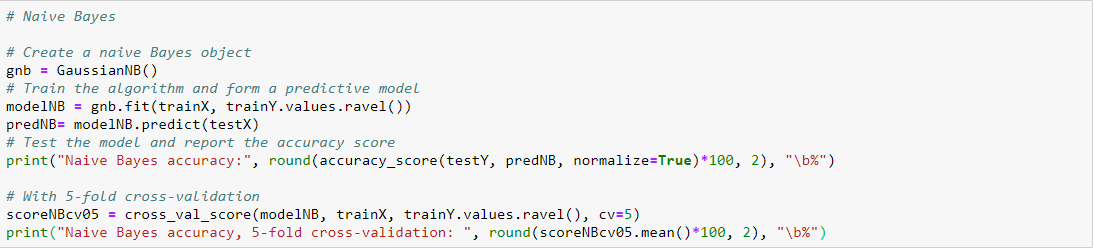
This was the final step before employing the prediction models, all of which were coded by Jack Harris.

**Model Descriptions**

The first model, Naïve Bayes, is a method of classification based on Bayes Theorem. This theorem assumes that each predictor in a dataset is independent of the others. The model is considered “naive” because of this assumption. Even if it is well known that the predictors are related to one another, the model still treats every predictor as entirely independent. These predictors are then used to predict the posterior probability. The assumption of independence allows the model to be run very quickly since it does not have to calculate the probability for every single combination of predictors. However, this speed is a tradeoff for a more accurate model that takes the relationships of predictors into account. Naïve Bayes works using the equation below.

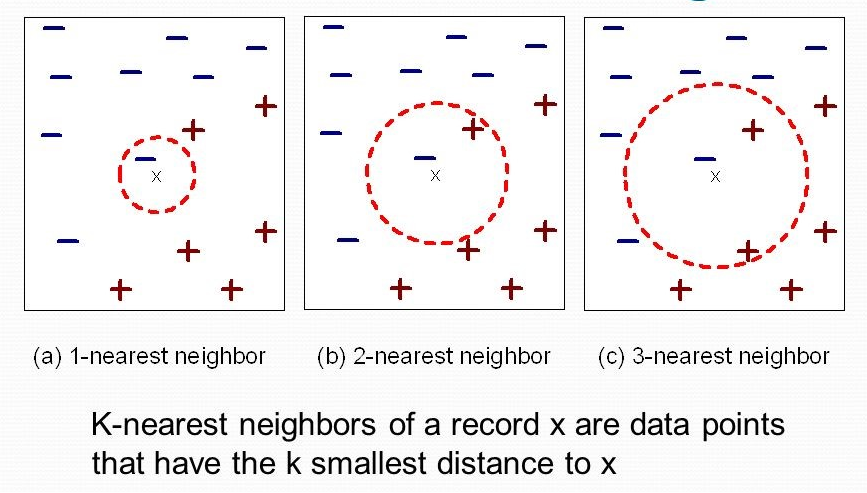


Naïve Bayes was employed in this analysis on the training data to attempt to predict the difficulty rating on the SAC Hiking Scale of each trail. It was not an effective tool in predicting difficulty, as it could only do so 44.14% of the time.



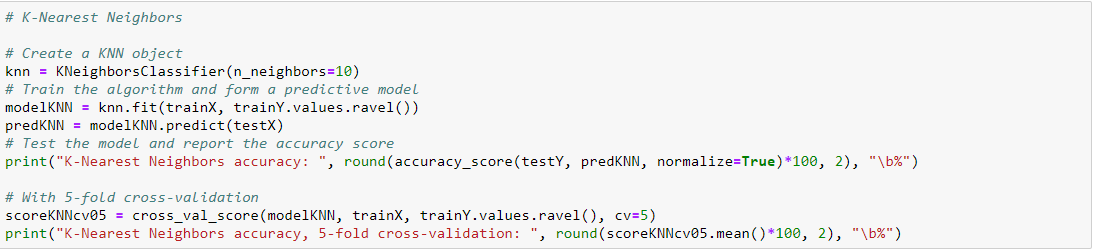


The second model employed was the KNN algorithm. KNN stands for K-Nearest Neighbors. It works in a lazy way, which is to say that it defers computation until the classification stage of the process, only approximating prior to reaching that stage. Another way to express this is to say that it does not train on the data, saving this step until it is ready to make predictions. Other algorithms train themselves on the training data, and then apply the training to the prediction portion of the analysis.



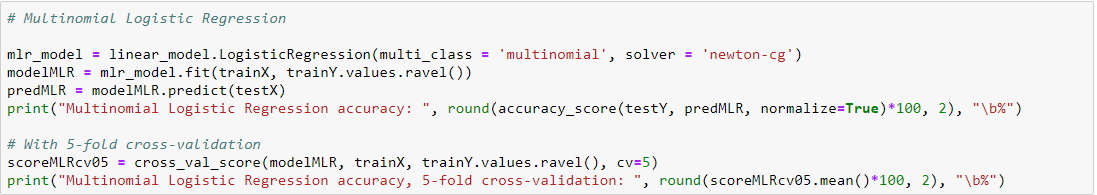
KNN works by finding the nearest neighbors of each classification and weighting them according to their distance to x, an unclassed variable. X can only belong to one class or the other, so the KNN algorithm will draw a circle that includes x and one or more of the other classes to determine how many “votes” are cast for each class. In the example above, the circle only includes the blue negative at first, then includes a vote each for the blue negative and red positive as it expands to the second level. When it reaches the third level of expansion, it includes two votes for red positive and one for blue negative. While one blue negative is closest to x, it seems that further expansion will class it as red negative, although that may not be true. This is KNN in a nutshell. It is not always perfect, but works well when the data are easy to classify.

KNN is supervised, which means the label of the training data is known, and the model is looking to train the on the other features to classify the data on that label. A larger k value can reduce the effects of noisy data. However, using a k value that is too large can diminish the effects of smaller patterns in the data. Ideally, KNN is used for domains like recommendation engines, search engines, and anomaly detection. Unfortunately, KNN was only able to predict difficulty 45.39% of the time.





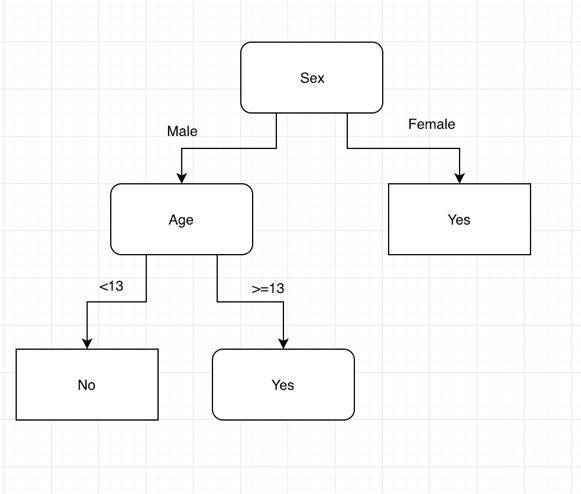
The third prediction model in the project was multinomial regression, which is used to predict a dependent variable given one or more independent variables. It differs from simple binomial regression in that it allows for a dependent variable with more than two categories. Like other regression models, multinomial regression allows for interaction between independent variables to predict the dependent, or target, variable. A multinomial regression might be used to predict what drink diners prefer based on their menu choices, or what snacks sell better at a theater based on the movies shown and the time of year. As such, it seemed to be a very good choice for this project, but was only slightly more successful than the other models. Multinomial regression was only able to predict the target variable 49.69% of the time.

****

****

****

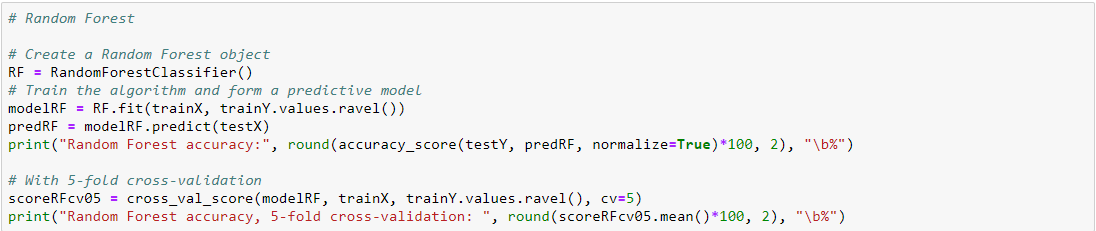
To conclude the predictive model section of the project, Jack used Random Forest (RF). A random forest is a collection of randomly generated decision trees. However, these trees are not generated in the same manner as in a Decision Tree analysis. When creating a single decision tree, the tree is created using all the data and variables and then pruned to generate the best tree. A simple decision tree on a data set of Titanic Survivors is shown below.



In decision tree analysis, each tree is created using a randomly selected group of data and variables. The output from these trees is combined to create the final output of the model. Each tree in the forest gets a “vote” to classify the data point. The label with the highest percentage of votes will classify the data point. This random method leads to a less biased result and typically performs better than a single decision tree. A simplified example from an article on Medium.com is shown below.



RF is able to use features of the variables to differentiate them from each other. In a forest, one may observe Douglas firs, spruce, and pine growing together in a grove. It would be enormously difficult to differentiate the trees without entering the forest and looking closely at the features of each tree. Looking at bark patterns, needles, and cones would give a forester the necessary information to classify the trees, information that could potentially be extrapolated out to the forest as a whole. Using the RF algorithm, the analysis can be tuned by generating features that differentiate variables, giving the model greater power to predict the target variable. While RF is a robust model with the ability to generate excellent results, it was only accurate 44.77% of the time in this analysis.





**Description**

The program first imports all the packages, then the data via a CSV file from Kaggle.com. It then selects the desired columns and derives values from them. Once this is done, it performs several cleaning, organization, and transformation functions to prepare the data for the models. It then visualizes boxplots for outlier identification, removes outliers, and describes the data before splitting the data into training and testing sets. After all the preparatory functions are complete, the program runs the four models used and performs five-fold cross-validation for each one.

**Conclusion**

Because the SAC Hiking Scale rates hikes by difficulty, it seems as though the features of a hike such as length, uphill climbing, elevation range, average speed, and total time would be important factors in the rating. As hikes increase in difficulty on the scale, descriptors such as unsure footing, pathless slopes, trails not clearly marked, and steep grades all seem to be important factors in the total time taken to complete a hike and the speed at which the hiker is moving, as well as the elevation range. On the surface, it would seem that any factors that might be related to the descriptions in the SAC scale would be excellent predictors of the SAC rating, but this does not seem to be the case. The SAC difficulty seems to be more a function of somewhat subjective criteria.

Whether or not a hiker’s footing is unsure could be a consequence of the condition of the hiker’s shoes, yet unsure footing is a feature of the difficulty rating. On icy paths, some hikers may attach crampons to their feet and walk as surely as on pavement, while less-prepared hikers slide down the path unable to continue. And while some paths may be well-marked in spring when annual foliage is thick everywhere but the trail, winter may obscure the path hopelessly. Even weather could affect a trail rating. A surprise summer storm might make a hiker believe she was on a trail of much greater difficulty than the fair-weather reality.

All these dynamic factors are considered in the SAC difficulty rating, but have no clear correlation to the other variables available in the data set. Additionally, the variables used in this analysis are likely collinear. If the project was larger in scope, it would be beneficial to test the variables for multi-collinearity to ascertain whether this was a factor in the model accuracy. Models could also be further tuned and more hikr.org data could be imported, or other sources could be considered in the analysis.