Big Mountain Resort, a ski resort in Montana, is trying to determine strategies that aid in modifying its pricing strategy in order to make up the increase in operating costs due to the addition of a new chair lift. They want to see if they can increase ticket prices while still being a competitive player in the ski market across the country, and they want to examine what resort features are most correlated with ticket price.

After procuring the data from the database manager, my first task was to look over it, checking for missing values, invalid entries, and duplicated rows. I determined that I would use either the adult weekend or the adult weekday ticket price as my target variable. I plotted both ticket prices against their values per state (with a groupby) in a horizontal barplot and a Seaborn boxplot in order to understand the ticket price distributions per state.

I looked for outliers to determine if there were any irregularities (which would help me determine invalid entries), and I found a few suspicious entries which I dealt with by searching for the correct information and manually replacing them in my data. I examined each column's missing value distribution individually, and I made informed decisions to drop columns that had a lot of missing entries (and did not contain particularly useful information). I noticed that there were two price columns - AdultWeekend and AdultWeekday, and I dropped all rows that had both price values missing.

In order to understand the state-wide market segment for ski resorts, I decided to create a state summary statistics dataframe containing the number of resorts per state, the total skiable area in a state, etc. I wanted to obtain non-ski resort summary information for states as well, like its area and population, and used Wikipedia for this purpose.

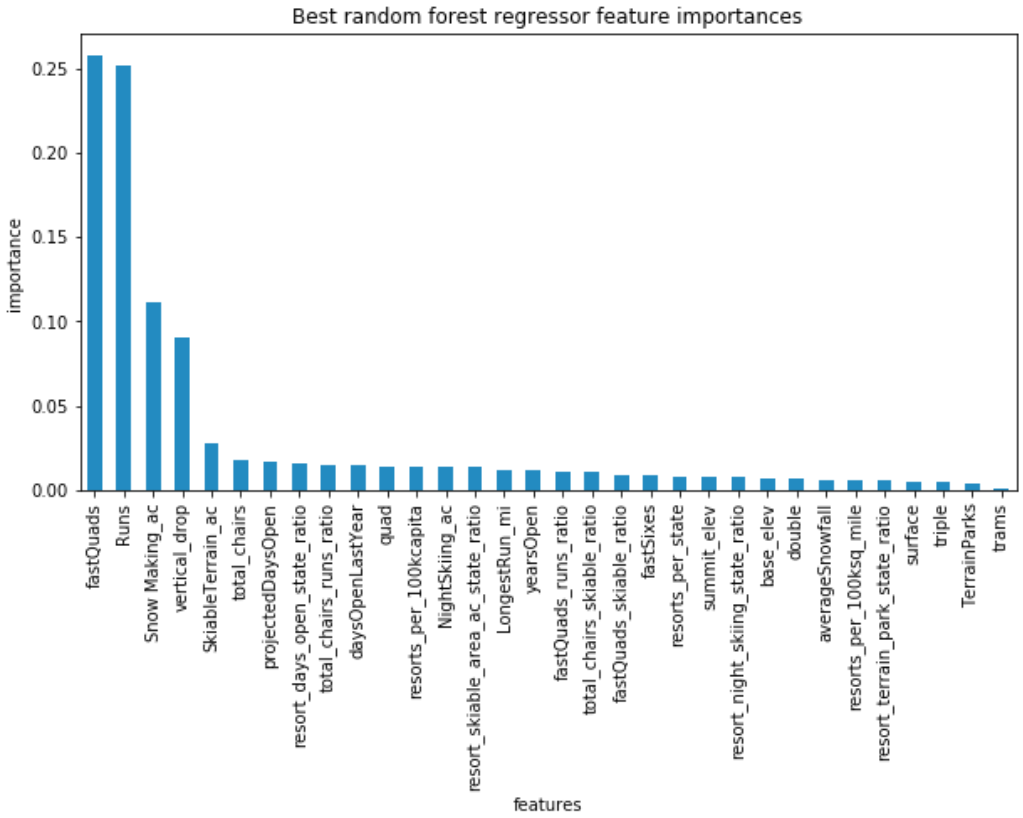
Checking for missing values helped me decide which feature I should use as my target variable between AdultWeekday and AdultWeekend, as I picked the feature that had a smaller number of missing values. I dropped the AdultWeekday column and all the rows that had missing values in the AdultWeekend column from my dataset.

I decided to examine my state summary dataset some more to see what I could learn about how the state influenced the pricing model of its ski resorts. I calculated the resort population density and resort area density per state and included these columns in my dataframe as well. At this point, my state summary dataset had 7 numerical columns (excluding the name of the state), and so I decided to use principal components analysis (PCA) to reduce the dataset to two primary variance-producing components in order to help visualize it. I scaled my dataset using scale() and used PCA().fit() on my scaled data to fit the PCA transformation. Upon plotting the cumulative sum of the explained_variance_ratio_ attribute of my fit function, I observed that the first two components/features accounted for around 75% of the variance and so visualizing the dataset just using these two components made sense. I then transformed my fitted data and plotted the first two principal components on the x and y scale of a scatter plot respectively.

I created a dataframe using the names of states as an index with the columns including the first two principal components of each state and the average AdultWeekend ticket price per state. I plotted the two principal components on the x and y axis respectively of a scatter plot, again, but this time I marked the size and color of the scatter points with the quartile of ticket prices that the state belonged to. This helped me visualize similarities between states on the basis of their ski summary statistics and also helped me see how their pricing model could be affected by these similarities. Examining ski statistics for each resort on a heatmap allowed me to see which features were more correlated than others.

Then it was time to create a machine learning model to train and test! I split my dataset using a 70 (train) - 30 (test) split where x = all of the numeric columns except ticket price and y = ticket price. I used DummyRegressor to determine how good of a predictor the mean of y is for y. I examined various metrics like r-squared, mean absolute error, and mean squared error to determine how good my model is at predicted actual y values. I then built two pipelines, one with LinearRegression() and one with RandomForestRegressor() to determine which model had the best performance and could be used on my dataset. Each pipeline also contained a SimpleImputer(strategy='median') which filled in the missing values in each column with the median value of the column and a StandardScaler() which standardized the data (by converting values to their z-scores) since the features were not homogenous. I decided not to use SelectKBest() to select the k best features from my dataset using a score function in my pipeline, since it did not contribute to a better performing model. I assessed my models' performances using cross-validate() with cv = 5 in order to avoid attuning the model to my dataset and overfitting. To find the best parameters for each of my functions in the pipeline, I used GridSearchCV() which showed me that imputing missing values with the median was better than with the mean and that scaling features did not help.

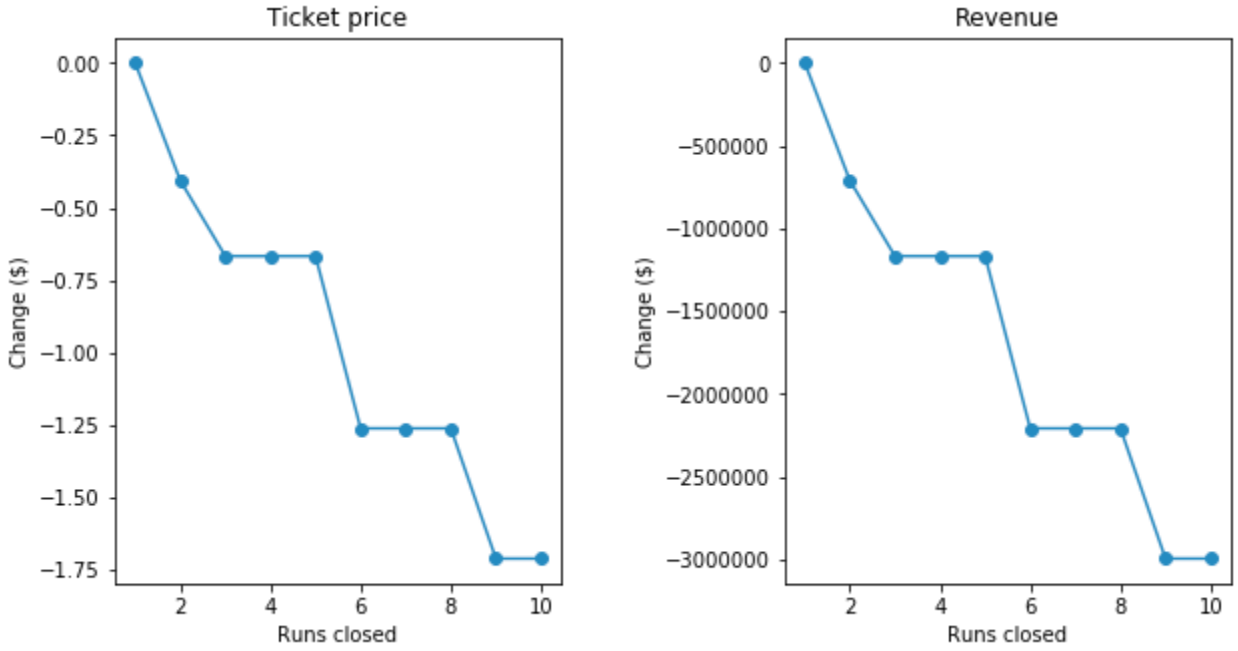
Upon cross validating my linear regression and random forest regression models, I determined that the random forest model performed better (had a lower mean absolute error) and so I decided to use that on my dataset.



I fit this model with the entire dataset excluding the Big Mountain Resort data since that is the value I want to modify. I predicted that the price that Big Mountain could charge according to the

model is $94.22 while its current ticket price is $81. However, this value is calculated under the assumption that the pricing model of all the resorts used to calculate it is perfect and no resort is either overcharging or undercharging, which may not be a true assumption.

There are two possible options for increasing profits - increasing revenue (through charging higher ticket prices) and reducing costs (by making changes to the resort's facilities). Even if Big Mountain increases ticket prices, the addition or removal of certain facilities incentivizes visitors to come, and so thinking of the pricing model as a two-step ladder is more efficient than a simple increase or decrease of ticket price. The first scenario I tested was closing down up to 10 of the least used runs and examining the change in the modelled price due to this. I discovered that closing any number of runs does not increase the modelled price and so abandoned this idea.



Next, I tried adding a run, increasing the vertical drop by 150 ft, and installing an additional chair lift, and this supported a revenue increase of $3,474,638. I then tried two more scenarios which did not provide any support for increased revenue.

The additional chair lift that the resort has recently installed increases its operating costs by $1540000. Before the data analysis, the resort's pricing strategy has been to charge a premium above the average price of resorts in its market segment.

I recommend an increase in the number of runs by 1 and an increase in the vertical drop by 150 feet. Scenario 2 also called for an additional chair lift to be installed, but I am assuming that the new chair lift we have recently installed takes care of that. Modelling this scenario shows us that there is additional support for increasing the ticket price by $1.99, which increases total revenue (assuming 350000 visitors and 5 days of skiing per visitor) by $3474638. Hence, I would also recommend that ticket price be increased from $81 to $82.99. This definitely counters the new operating costs caused by the installation of the chair lift.