**Problem Overview**

Big Mountain Resort, a ski resort in Montana, recently installed an additional chair lift to help increase the distribution of visitors across the mountain. This new chair lift brings in an added operational cost of about $1.5 million this season. Thus, they have decided to re-examine their pricing strategy to determine if the strategy of charging a premium over the average price of resorts in its market segment is the best approach. The business wants some guidance on how to select a better value for their ticket price. They are also considering a number of changes that they hope will either cut costs without undermining the ticket price, or will support an even higher price.

**Data Wrangling**

Initial glance of the dataset shows 330 rows and 27 columns, where each row aligns to a resort (including Big Mountain) and contains info about that particular resort. Next, we checked for missing values across each of the columns. The ticket price (our desired target quantity) is missing 15-16% of values - AdultWeekday is missing in a few more records than AdultWeekend. About 14% of the rows have no price data at all (both Weekend/Weekday prices are missing). Since price is our target, we dropped those rows.

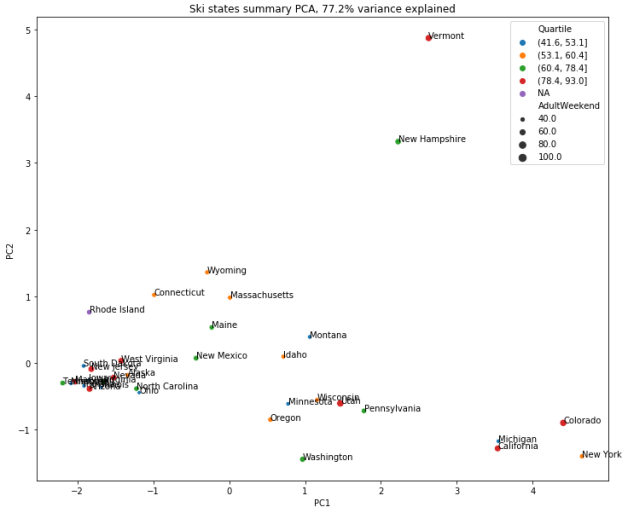
We plotted histograms of each numeric feature to look at their distributions and spot any outliers to investigate. SkiableTerrain_ac and SnowMaking_ac values are clustered down the low end. fastEight has very little variance (mostly 0) and half the values are missing. FastSixes and trams have more variability, but still mostly 0. yearsOpen has a max value of 2019, suggesting someone recorded the calendar year instead. We took remedial steps which included dropping the fastEight column entirely because half the values are missing and there is essentially no information in this column. We dropped rows that had yearsOpen > 1000.

Next, we wanted to include more state specific information to the data so we imported a table containing this data from wikipedia and extracted the state name, population, and area in square miles. We merged the new information with the ski resort data using a left join on ‘state’.

Finally, we wanted to determine what our target will be when modeling ticket price - Weekday or Weekend? Weekend prices being higher than weekday prices seem restricted to sub $100 resorts. Furthermore, the distribution for weekday and weekend prices in Montana seemed equal. Weekend prices have fewer missing values so we decided to drop weekday prices and just keep rows that have weekend prices. Our dataset is in a good spot and the data science problem we subsequently identified is to predict the adult weekend ticket price for ski resorts.

**Exploratory Data Analysis**

After wrangling the data, our resort dataset now has 277 observations and 36 features, two of which are categorical features: Region and State. After performing PCA, The resultant plots showed that in the representation of the ski summaries for each state, which accounts for some 77% of the variance, we simply do not see a clear pattern or correlation with price. This suggests a model which considers all of the states together. I merged state summary features into the ski resort data to add “state resort competition” features in order to understand what



share of a state's skiing assets is accounted for by each resort.

Next we used a feature correlation heatmap to gain a high level view of relationships amongst features. We also created scatterplots of the numeric features against ticket price. In the scatterplots you see what some of the high correlations were clearly picking up on. There's a strong positive correlation with vertical_drop. fastQuads seems very useful as well. Runs and total_chairs appear quite similar and also useful.

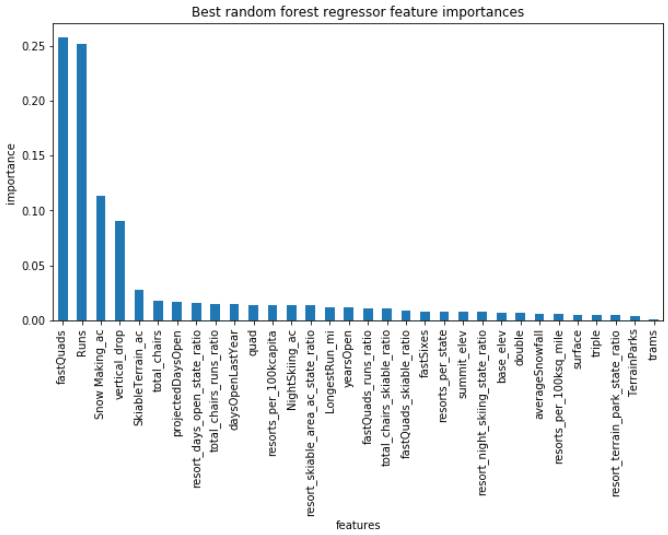
Lastly we added some features relating to how easily a resort can transport people account, for example the ratio of chairs to runs.

**Model Preprocessing**

We started by looking at the simplest model, using the mean as a predictor. We split the data into training and test using a 70/30 split to minimize bias or overfitting within the model. We calculated y_train.mean() and compared it to sklearn's dummyregressor. To evaluate, we used the R^2 metric, or coefficient of determination which measures the proportion of variance in the dependent variable that is predicted by our model. As expected, we got an R^2 of 0 when using the mean. And performance on the test set was slightly worse.

Next we built a linear model. We imputed missing values in the training and test splits with the median and then scaled the data. This gave an R^2 of 0.82 and 0.72 for train and test respectively. We suspected that the model was overfitting given the number of features we blindly used so we used sklearn's SelectKBest feature selection function to select the best k features, using f_regression as the score function. We assessed performance using cross validation, meaning we partition the training set into k folds, train our model on k-1 of those folds, and calculate performance on the fold not used in training. According to the GridSearchCV function in sklearn, k = 8 is best, including features like vertical_drop, Snow Making_ac, total_chairs, and fastQuads. Results suggest that vertical drop is the biggest positive feature which makes intuitive sense and is consistent with our findings from EDA.

We then built a Random Forest Model. Random forest has a number of hyperparameters, however here we'll limit to exploring some different values for the number of trees - with and without feature scaling, and try both the mean and median as strategies for imputing missing values. We found that imputing with the median helps, but scaling the features doesn't. After plotting the random forest's feature importances, we see that the dominant top 4 are fastQuads, Runs, Snow Making_ac, and vertical_drop, which are in common with the earlier linear model.



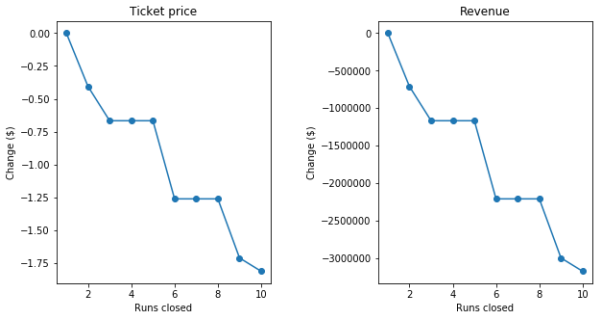

Ultimately we chose to go ahead with the random forest model. Looking at both models' performance showed that the RF model has a lower cross-validation MAE by almost $1. It also exhibits less variability. And verifying performance on the test set produces performance consistent with the cross-validation results

**Modeling**

The business shortlisted 4 cost-cutting / revenue increasing options:

1. Permanently closing down up to 10 of the least used runs. This doesn't impact any other resort statistics.
2. Increase the vertical drop by adding a run to a point 150 feet lower down but requiring the installation of an additional chair lift to bring skiers back up, without additional snow making coverage
3. Same as number 2, but adding 2 acres of snow making cover
4. Increase the longest run by 0.2 mile to boast 3.5 miles length, requiring an additional snow making coverage of 4 acres

We modeled each scenario to get a sense for how facilities support a given ticket price. We assumed average visitors ski for 5 days and there are 350K visitors in the season. The model for scenario1 showed closing one run makes no difference. Closing 2 or 3 successively reduces support for ticket price and so revenue. If Big Mountain closes down 3 runs, it seems they may as well close down 4 or 5 as there's no further loss in ticket price. Increasing the closures down to 6 or more leads to a large drop.




Modeling scenario 2 showed this scenario increases support for ticket price by $1.99 and over the season expected revenue increase is 3.47 million. This model does not account for the additional capital expenditure however. Modeling scenario 3 showed that such a small increase makes no difference to ticket price/revenue. And modeling scenario 4 showed no difference whatsoever.

**Recommendations and Next Steps**

Based on these results, it appears that Scenario 1, closing up to 5 of the least used runs, would be a strong option forward. While you may be charging less for ticket prices and reduce revenue by ~$500K, you would also lower operating / maintenance costs significantly. The business could test the scenario by ranking the runs based on usage and then eliminating the runs one at a time to determine effect on price and costs.

Scenario 2 is another strong option for consideration as it has a projected revenue of $3.5 million. However further analysis would be required to see how the installation of an additional chair affects costs and overall profit. Would it be possible to use the new chair lift that Big Mountain has already paid for to support this scenario? Further discussion with the client team is required.

The only price data in our dataset was ticket price. A next step to improve the model is to incorporate operating cost data for specific features into the dataset and perhaps visitor volume across the U.S. regarding willingness to pay a premium by amenities and features.

To make the model more accessible, the model could be deployed through an interactive dashboard so that business executives can easily see the effect each data point has on the price and could easily visualize the effects of various business scenarios (changes to current feature values) on revenues and profits.