Big Mountain Resort Project Report

Problem Statement:

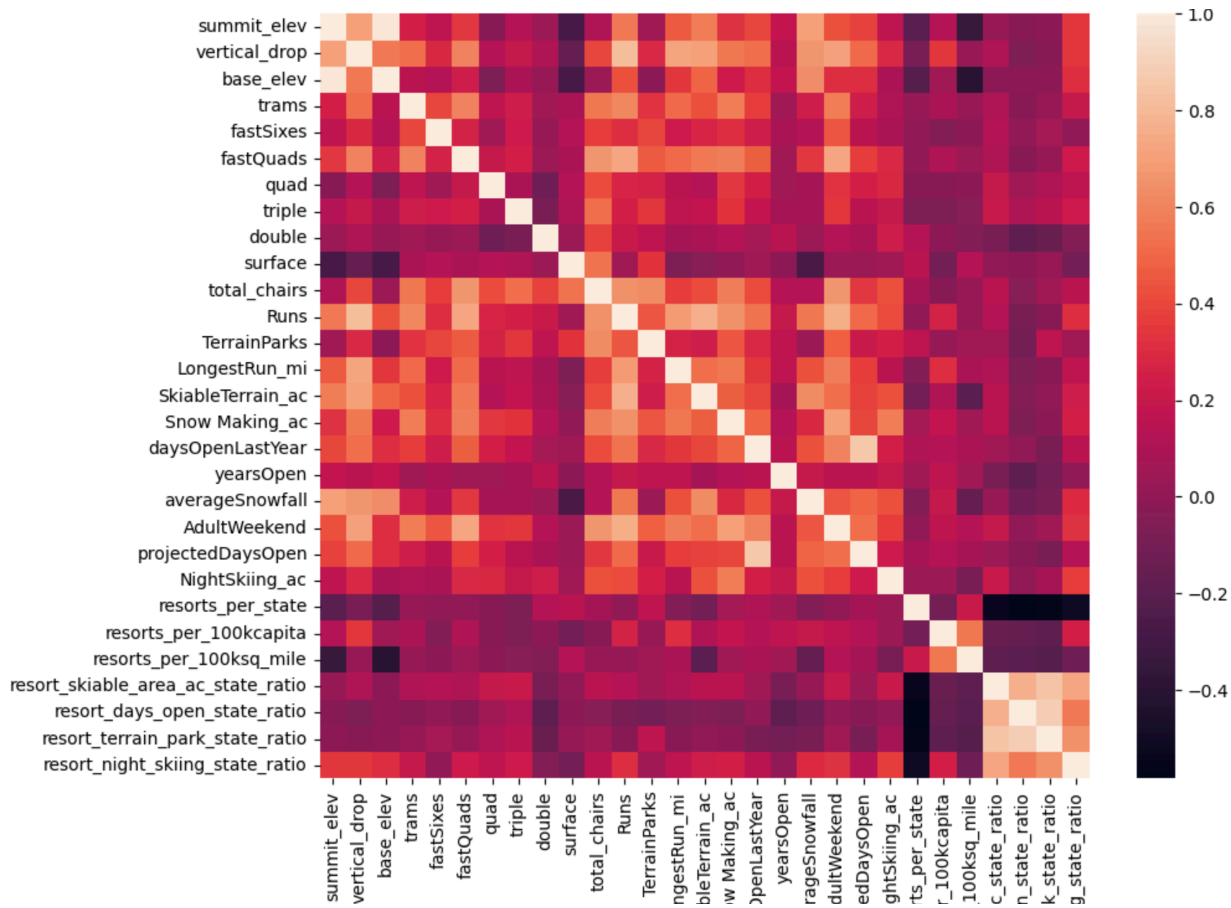
How can Big Mountain Resort optimize its revenue through a combination of increasing, upgrading, or removing lifts and changing its ticket costs? The goal is to increase revenue by 10% during the next ski season. Optimization will be determined from data driven analysis.

Data Wrangling:

To begin we observed that Big Mountain Resort was present in the data, and that there were 330 resorts to be used as data. Each row of data which corresponds to a resort had 27 features. We then did some basic data cleaning. We removed the 'AdultWeekday' feature column that contains weekday ticket prices because it is similar to and has less data than the 'AdultWeekend' feature column. We also removed the 'fastEight' feature column because it only had one row with a non zero or non null value. We then removed the rows with no value for the 'AdultWeekend' feature because we determined that ticket prices are the feature of interest that we are conducting our analysis on. The other row we removed was the resort with 2019 as the year's open value because this was a missed input, and the resort is too new to be of use in our analysis. These removals left us with 277 rows of data. Something odd we found is that the percentage of missing values for rows appear in multiples of four. There was nothing we could do about this, but it is important to note. After displaying histograms for the values of the numeric features and the quantity of missing inputs for each feature we took note of what features to investigate further. These were the features corresponding to total lifts, fast quads, skiable terrain, vertical drop, and day open out of the year. We are looking for how these features correspond to ticket prices.

Exploratory Data Analysis

The only categorical features in the data were the state and the region. No analysis was done on the region category since its values are largely the same as the state column. After performing a pca analysis with the state summary data we checked to see if there were any patterns that suggested a relationship between state summary data and ticket price, but we found none. This led us to decide to use the numeric features in the original data set for subsequent modeling. We wanted to see if any of these numeric features were correlated to a higher or lower ticket price. We created a feature correlation heatmap to gain a high level view of the relationships among features.



Based on the exploratory analysis we will not use the state labels in the data because the state features did not suggest any relationship between states and ticket price. The target features that look good for modeling because of their positive correlation with ticket price are fast quads, total chairs, runs, vertical drop. Snowmaking is also positively correlated, but because snow making is expensive it might not be something that actually improves profitability. Still it is probably worth targeting.

Model Preprocessing with feature engineering

The first baseline model compared the average weekend of adult weekend ticket prices to each actual value for adult weekend ticket prices. This was a good baseline because all future models examined should outperform this baseline model. The results for this baseline model was a value of around -0.003 for R-squared, around 19.136 for mean absolute error, and around 581.437 for mean squared error (these results are from the testing data).

Before trying the linear and random forest models we had to do some feature engineering. This included filling the missing values in each column with the median of the values in that column. The values were also all put on the same scale using the scale function. No new features were created using the data

Next I built a linear model, and this model identified features that were most important in determining ticket price. vertical_drop, Snow Making_ac, total_chairs, fastQuads, and Runs were all positively correlated with ticket prices whereas trams and SkiableTerrain_ac were negatively correlated with ticket prices. Using cross validation with 5 folds the estimate on the linear models performance was between 0.44 and 0.82 for R squared (two standard deviations away from mean), and its performance on the test set was consistent with this.

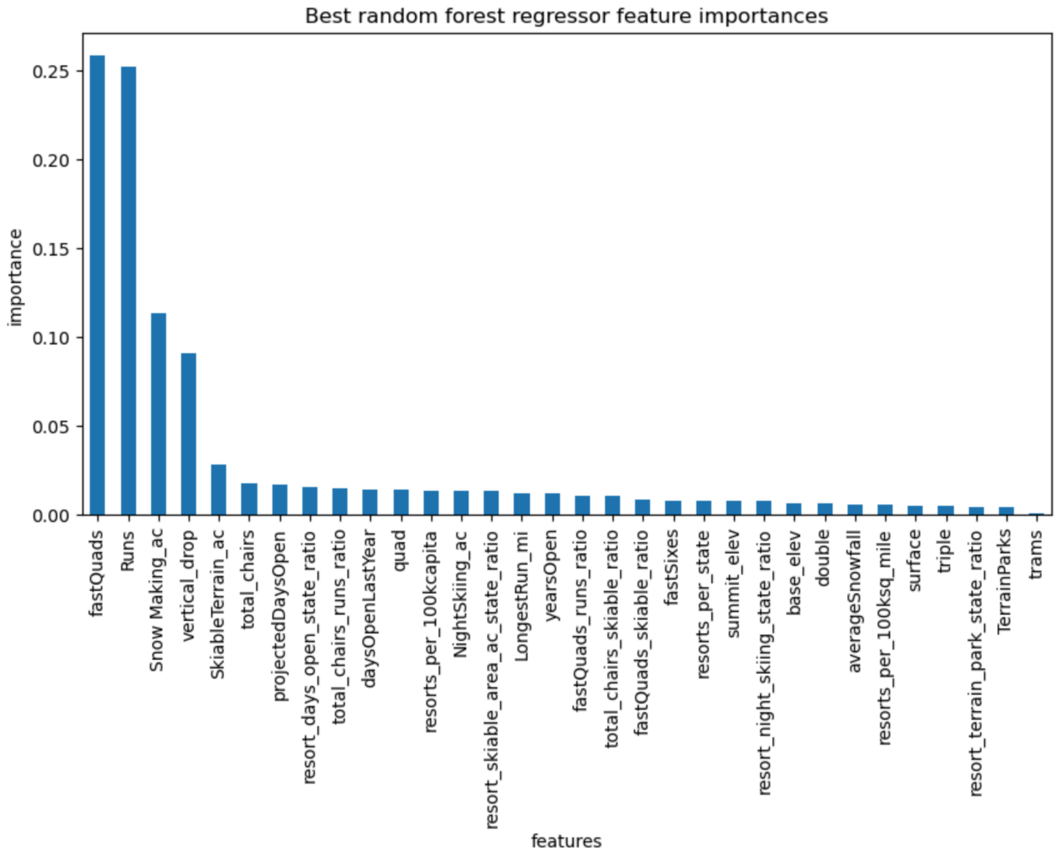
Lastly I tried a random forest regressor. The preprocessing steps that were determined best for model performance were 69 trees, no scaling, and using the median to impute missing values. Its estimated performance via cross-validation using R squared was a mean of 0.710 and a standard deviation of 0.065 for 5 fold cross validation, and its performance on the test set was consistent with this.

Winning Model

I have decided to go forward with the random forest regressor model because its mean of mean absolute errors and standard deviation of mean absolute errors 9.645 and 1.353 are lower than the values for linear regression which were 10.499 and 1.622.

I also did a data quantity assessment and determined that more data would not be useful as the cross validation score leveled off by around a sample size of 40-50.

The following graph shows the most important features in our model for determining ticket price:



Pricing Recommendation, Future Improvements, and Conclusion

Big Mountain currently charges 81 dollars for an adult weekend ticket. Our modeling suggests that they could charge a ticket price of 95.87 dollars based on their facilities with a standard deviation of 10.39 dollars.

I suggest to Business leadership that their quantity of fast quads and total chairs, which are above average for resorts, support a higher ticket price. Moreover, their ski resort's above average vertical drop and large number of acres covered with snow making also support a higher price point. If they wanted to make more improvements to their facilities to justify a higher price point I would recommend adding a chair lift that extends the vertical drop by 150ft. These changes increase support for the ticket price by 8.61 dollars. This increase in ticket price would amount to 15,065,471 dollars gained over the season if we assume each visitor buys 5 day tickets. This investment could be worth it especially given the cost of operating the new chairlift was only 1,540,000 dollars.

For future improvements I would look into adding more fast quads and increasing the vertical drop even more if possible. I would not suggest adding more snow making or increasing the longest run as the model shows these having limited to no effect on ticket price. If there are runs closed the business can still maintain a high ticket price based on the model. Closing 2 or 3 runs corresponds to a small decrease in ticket price but closing 4 or 5 runs has the same effect as closing 3 runs.

Future Scope of Work

A big deficiency in the data was no feature for the number of visitors to each resort. This information could help to explain ticket prices. It would have been nice to see estimated total revenue from each resort as well. While the additional operating cost of the new chair lift is useful cost information relating to snow making, cost of adding or removing trails, and the cost of different chairlift types. An explanation for why Big Mountain's modeled price was so much higher than its current price is that it is already by far the most expensive ski resort in Montana. Perhaps this is why business executives haven't already increased the price of tickets, and thus it might come as a surprise that the model supports charging more for tickets. Perhaps we should look to see if all Montana resorts are underpriced based on the model.

If the business leaders felt this model was useful they could experiment with changing the number of chairlifts, trails, vertical drop features and seeing what the model outputs. They then could decide what upgrades to make based on these results. They wouldn't need to come to me every time they want a new combination of parameters. The model would be made available so that business analysts could try different combinations of parameters, but they would not be able to change the underlying model.