Guided Captsone Project Report

Sam Castillo

**The Problem**

This project began as an effort to help Big Mountain Ski Resort rethink its ticket pricing strategy and approach to facility renovations based on data-driven decisions rather than subjective perspective. Presently, BM’s ticket prices are derived from the average ticket price for resorts in the same market share (approximately $65) with a $16 premium to account for BM’s above-average facility, totaling $81. While this approach has been relatively sustainable up to now, BM is facing a $1.5M increase in operating cost due to a new chairlift and questions as to how to move forward with equipment and facility investment. Moreover, BM executives are worried that even with the added premium, current prices don’t reflect the value of facilities that the resort offers to customers.

**The Solution**

Raising ticket prices, cutting costs, and devising new strategies requires more than a keen eye for business – to complement and renew BM’s current strategies and investment campaigns, I’ve created a data-driven model that predicts ski resort ticket prices based on the resort features available (vertical drop, total runs, number of chairs, etc.). This model accounts for 32 different features based on intra-resort features, ratios of features across resorts, and state-based ratios. The goal of the model was to determine how much, if at all, Big Mountain can reasonably raise prices according to its resort features based on deep analysis of other resorts’ prices and their features. The model suggests that BM has room to raise ticket prices by $14 per ticket – with next season’s expected turnout and an average stay of five days, this increase would generate about $26M in additional revenue.

As exciting as this sounds, the model makes a few assumptions due to lack of available data, and it doesn’t account for the subjective approaches other resorts might employ in their pricing strategies. However, the model does provide detailed predictions of ticket value based on current and projected facility features, so it provides objective leverage to increase prices as well as a measure to consider future facility renovations. I will include my recommended course of action at the end of this report.

**The Process**

***Data Wrangling and Exploratory Data Analysis****.* I began with a large dataset of 330 ski resorts across the US in Big Mountain’s market share. The data included several features for each resort but was missing quite a bit of data due to collection issues. I removed the following data: several rows that didn’t contain any pricing data and were thus unhelpful for this analysis; one row for an ambiguous feature; one column for providing no info; and I made one outlier correction through web-scraping. I spotted many of these anomalies and inconsistencies by displaying distributions of various features (***Figure 1***). By the end of this, my dataset contained 277 rows and 25 columns.

In addition to the ski data, I found it useful to gather basic state statistics to inform population and area based ratios of the resort features. I gathered this summary of states from Wikipedia and wrangled it into a format suitable for merging into my larger dataset, though I waited to merge until further analysis on the state summary. First, I needed to decide whether I should use the state labels to partition features or contribute to the target feature analysis or simply treat resorts across all states equally.

To answer this question, I conducted a cursory check of the most prominent features by state (e.g., total runs, resorts per state, etc.). This revealed a few loose patterns, but the most accurate way of visualizing the high-dimensional data was principal components analysis. The results of PCA revealed a few vague clusters of states based on the first two components, but a seaborn plot showed that these distinctions had no relationship to ticket price (***Figure 2***). So, I decided to move forward treating all resorts equally regardless of the state label. I did, however, keep the state-based ratios that seemed to offer some valuable information. At this point, I merged the ski data and state summary datasets and conducted further heatmap and scatterplot analyses on how features related to the target feature, ticket price (***Figure 3***).

The most relevant resort features to ticket price were vertical drop, total runs, snowmaking acreage, and total chairs - particularly fast quads. I made a point to keep an eye on these features as I began developing my model.

***Preprocessing and Training.*** I performed a test-train split on the ski data, using 70% of the observations as training data and the remaining 30% as test data. Training and test features consisted of 32 numeric features from ski\_data, while the training and test targets consisted of one target feature: weekend ticket price.

Before I created/evaluated/compared model performance, I developed a baseline idea of how well I could predict test targets based on the mean of the training targets. This was, as expected, generally inaccurate with a mean absolute error of about $19.

Given the unsuitability of the mean as an estimator, I created two models: a linear regression model and a random forest regression model (random state = 47). GridSearchCV found that simple imputation using the median was more effective than the mean in both models, and selectKbest based on f-regression scores yielded 8 features in the linear regression model, all of which were consistent with EDA (***Figure 4***). Between the two models, random forest with 69 trees proved slightly more accurate in cross-validation and in the test set performance. The linear regression best model yielded an MAE of 11.79 with a standard deviation of 1.62, and the random forest regression best model yielded an MAE of 9.53 with a standard deviation of 1.35. Given the lower error and standard deviation, I decided to move forward with the best estimators of the random forest model.

***Modeling***. As mentioned in the solution, the random forest model accounted for 32 of Big Mountain’s numeric features, the features of 276 other resorts, and the ticket prices of other resorts in order to predict an appropriate ticket price for Big Mountain. Based on BM’s facilities, the model predicted a price of $95 – a $14 increase from current prices. However, BM also wanted a model that could help drive renovation and operation cost-cutting decisions. In order to model these scenarios, I created a function that returned the difference in price given an adjustment to any number of BM’s features. Among the scenarios given by BM, the most profitable was to add a run that increases vertical drop by 150 feet and install an additional chair lift. My suggestions for implementation and general ticket pricing are below.

**Suggested Course of Action**

Add 1 run that extends the vertical drop by 150 feet and add one chairlift. Per the model, this should increase the ticket price value by $1.99 per ticket. To compensate for the extra operating costs, cut 3 previously existing runs - this will reduce value by about 0.41 per ticket, resulting in an approximate value increase of 1.58 per ticket in tandem with the renovations. Since you will be adding major features (i.e., increasing vertical drop), this is a good time to increase prices per the initial suggestion of the modelled price. A safe and reasonable increase would line up with the lowest end of the error allowed by the modelled price - that is, 95.97 - 10.39, a total of 85.58. This is a 4.58 increase from the current ticket price.

To account for the renovation value, I would recommend raising current prices by 4.58 in addition to the 1.58 of projected value increase to account for both the current facilities as well expected value of the renovated facilities. This brings us to a total of 87.16, a 6.16 increase from current prices. With expected turnout of 350,000, this will generate a 10.7 million increase in revenue next year, which will cover our current and projected increases in operating costs while leaving some room for comfort.

If the executive team is worried about whether customers will complain or choose other resorts in response, keep in mind that according to our model, Big Mountain is among leaders in the most important features that drive ticket price in resorts across the country. We don't need to worry about raising prices beyond our true value - we just need to strategize as to how to do so while also cutting costs. For instance, regarding the course of action I suggested above, I would close down one run at a time. When you close the third one, open the new run adding vertical drop at the same time to avoid dramatic value effects (losing 2 runs total is much better than losing 3). If the new run generates an above-average popularity for an extended period of time, then I would consider implementing a new chair. Overall, this process will take several months - but it will be worth the slow implementation to see how these decisions affect business one at a time, particularly run closures.

**Figures**

A screenshot of a graph

Description automatically generated

**Figure 1. Distributions of Features to Detect Anomalies.** A close look reveals a severely skewed distribution of the Skiable Terrain feature. This indicates an anomalous figure somewhere in the data – as it turns out, there was an incorrect value for Silverton Mountain that required correction via web-scraping.

A screen shot of a graph

Description automatically generated

**Figure 2. Seaborn Plot of PCA Results, Colored by Price Quantile**. There are a few patterns present in the component weights, but none related directly to price.

A group of blue dots

Description automatically generated

**Figure 3. Scatter Plots of Each Feature Related to Ticket Price.** Notice the positive correlations of Snow making acreage, total runs, fast quads, total chairs, and vertical drop. These are likely to be the most significant determining features in a modeling analysis.

A graph of a graph

Description automatically generated

**Figure 4. Line Plot Demonstration of SelectKBest.** GridSearchCV determined that the threshold of gleaning meaningful data for the linear regression model reached a peak r-squared and minimal error at 8 features.