Data Pre-processing

1. Taking Average Approach

Taking Average Approach, the one we are currently adopting. Conclusion: under this approach, we would expect to be off by around $19 on average.

2. Linear Regression modeling

Implemented python GridSearchcv capability. Cross-validation (with 70/30 split between train/test data) was utilized to compare various models with different combinations of some or all of features. An optimal linear model was found to be with eight best predictors, namely, a resort's vertical drop, areas covered by snow making equipment, total number of chairlifts, total number of four-person chairlifts, length of the longest run, the number of trams and the area of skiable terrain, listed in a descending order of feature's influence on ticket price. Their estimated coefficients are listed in the middle column of Table 1. The larger the absolute value of a coefficient is, the more power the feature has to affect the ticket price. A resort's vertical drop is found to be the biggest positive feature to bring up its price, Also, the area covered by snow making equipment is a strong positive as well. It seems that resorts with bigger vertical drops, larger area covered by snow making equipment, more chairlifts (especially fast four person ones), longer and/or more runs, less skiable terrain area and fewer number of trams have more leverage to set a higher price; Under this modeling approach, on average we would expect to estimate a ticket price about $10 or so off the real price (see Table 2). This is much, much better than the $19 from just guessing using the average.

Table 1. Regression Results

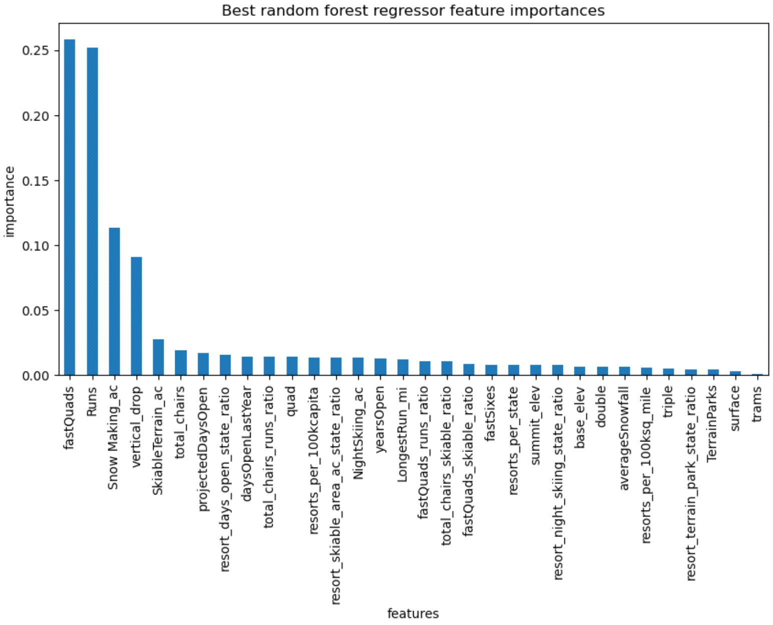
Features (standardized)	Linear Regression	Random Forest Regression
	Coefficients	Rank(top 8)
vertical_drop	10.767857	4
Snow Making_ac	6.290074	3
total_chairs	5.794156	6
fastQuads	5.745626	1
Runs	5.370555	2
LongestRun_mi	0.181814	NA*
trams	-4.142024	NA*
SkiableTerrain_ac	-5.249780	5

NA*: the feature did not make to the top 8.

3. Random Forest Regression modeling

Implemented python GridSearchcv. Cross-validation (with 70/30 split between train/test data) was utilized to compare models/forests of different sizes. A random forest with 69 decision trees were found to be optimal. Importance of all features were ranked in Figure 1. Although the order of importance is different (shown in the rightmost column in Table 1), this model does confirm the four most important features we identified in linear regression, namely the number of fast four person chairlifts, the number of runs, area covered by snow making equipment and vertical drops. Under Random Forest modeling, the importance of all features has a big drop/decline after the fourth one (vertical drop) as shown in Figure 2. So we suspect that a random forest model including only top four features might be sufficient enough, if a simpler version is preferred.

Figure 1. Feature Importance Rank Plot



4. Model Performance Comparison

Compared to the linear model, this random forest model provided better and more reliable/consistent predictions with less variability. The stats are shown in Table 2 given below. Besides, it does not assume linearity between the ticket price with other resort features. The linearity assumption is rarely true in reality.

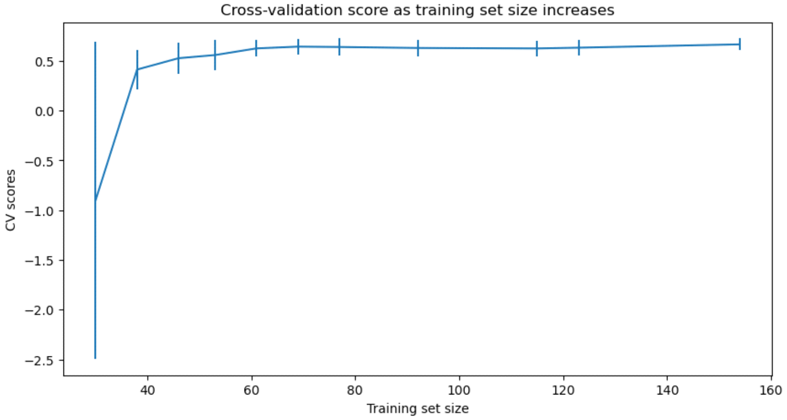
Table 2. Model Comparison

Model	Train Data		Test Data
	Mean Absolute Error	Standard Deviation of Error	Mean Absolute Error
Linear Regression	10.50	1.62	14.11
Random Forest Regression	9.58	1.37	9.54

5. Data Quantity Assessment

Do we have enough data to support our finding? The answer is yes, as shown in Figure 2, where we plotted Python learning curve, training data size vs performance score. The plot shows an initial rapid improvement in model scores as training data size increases, but it's essentially levelled off by around a sample size of 40-50. With our 70/30 train-test data split, we ended up with maximum of 193 data records in the training dataset, far more than enough.

Figure 3. Data Quantity Assessment


Cross-validation score as training set size increases
CV scores
Training set size