"Guided Capstone Project Report" Roger Swartz

Add your project report to your GitHub repo Guided Capstone folder

Big Mountain Resort, a Spectacular ski resort located in Montana with access to 105 trails, 11 lifts, 2 T-bars, 1 magic carpet and 350,000 annual skiing and snowboarding customers at every ability level recently installed an additional chair lift that increases operating costs by $1,540,000 this season is seeking a strategy to increase profitability. The current pricing strategy is to charge a premium above the average price of resorts in its market. Although BMS wonders if they are not capitalizing on their facilities as well as it could. Further, there is not a clear benchmark on how on the relative value of their facilities compared to average price resorts and other resorts and BMS worries they are not profiting as much as they could and thus leads to the worry that their investment strategy is or will become misguided. BMS wants specific guidance to understand the price/value proposition of their lift tickets to increase sales and/or how to cut costs without undermining ticket price/value.

The file ski_resort_data.csv was evaluated. The csv file consists of data on 330 resorts each resorts data is stored in its own separate row and there is one header row. The file contains ski resort data in 35 states and 38 regions. There are 8 states with 15 or more ski resorts and 15 states with 10 or more ski resorts (**Fig. 1**) and thus ticket pricing trends vs amenities by state would give some insight for ticket pricing trends in the State of Montana that has 12 resorts / ski mountains.

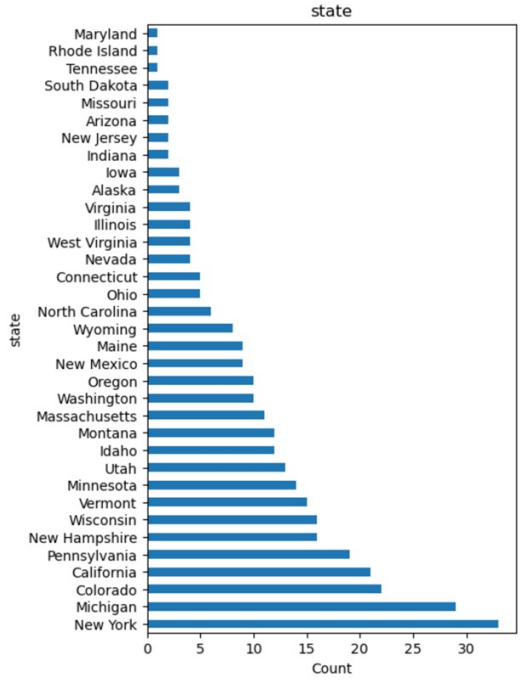

Fig 1. Number of Ski Resorts / Mountains By State

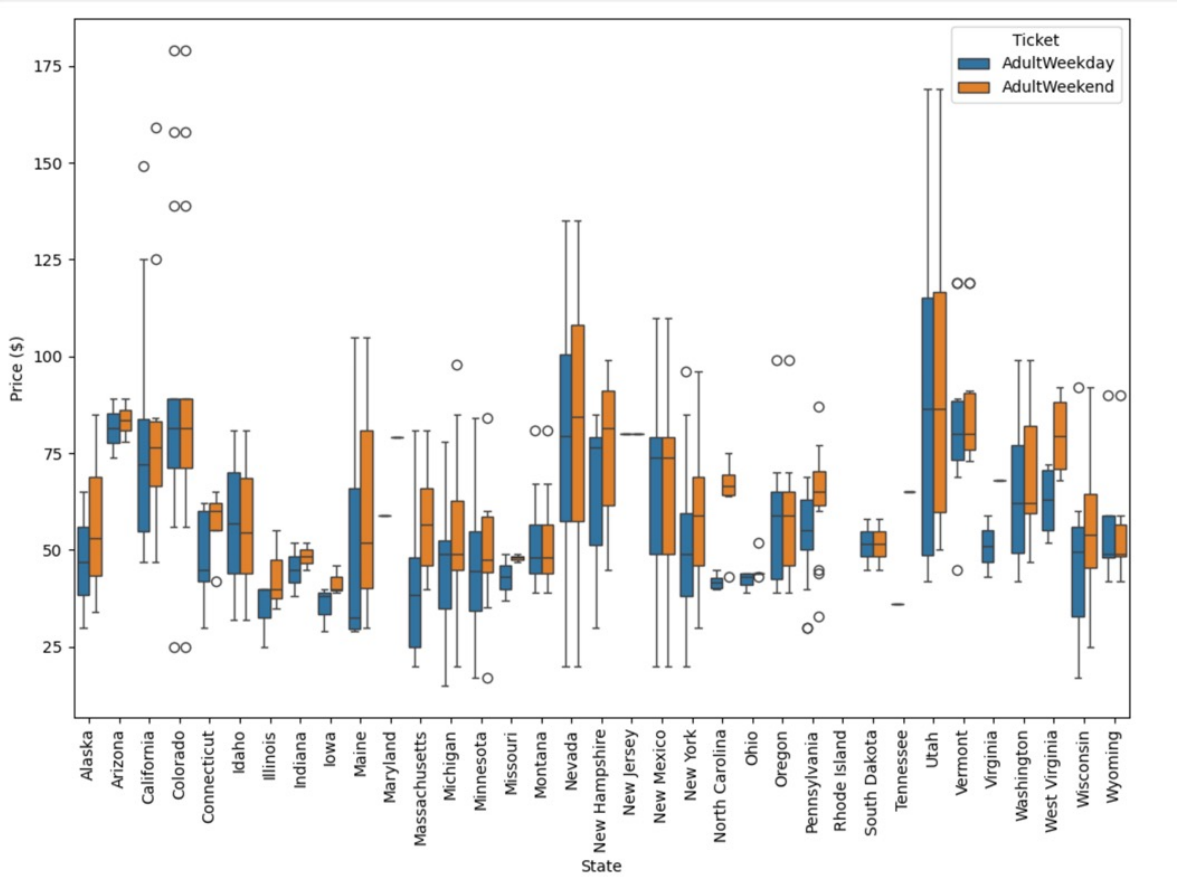

Fig 2. Boxplot of "AdultWeekend" and "AdultWeekday" skiing lift ticket prices by state

There are two ticket price columns: Adult Weekday and Adult Weekend. Later we decided that Adult Weekend was the more relevant value to consider. As some ski mountains / resorts have a large % difference in the price between AdultWeekend and AdultWeekday while other mountains have no difference at all. This could be explained in some cases by demand differentials. In other cases, a mountain may want to minimally discount an AdultWeekday ticket in order to not dilute AdultWeekend ticket sales – in those cases the mountain may very well have historically seen no difference in number of tickets sold on the weekend and weekday combined and figured all they would be doing with a lower cost AdultWeekday ticket is reduce AdultWeekend ticket sales and have an overall reduction in revenue.

A boxplot (**Fig. 2**) of ticket prices by state shows there are several states with a wide range of ticket prices. The price for an Adult Weekend lift ticket in the state of New Mexico for example can range from a price of $20 to a price of $110. The price for an Adult Weekend lift ticket in

the state of Montana can range from $39 to $81. Montana has a tighter boxplot than other states. Although, the boxplot itself can be a little misleading since the outliers are not included. And if you are working with an upper quartile that is made up of just a couple ski mountains it really should be included.

But also 14% of the resorts had no price data for neither the adult weekend nor the adult weekday and we dropped those resorts from that analysis.

There was an extensive amount of data to potentially use including:

1. summit_elev'
2. vertical_drop'
3. 'trams'
4. 'fastEight'
5. 'fastSixes'
6. 'fastQuads'
7. 'quad'
8. 'triple'
9. 'double'
10. 'surface'
11. 'total_chairs'
12. 'Runs'
13. 'TerrainParks'
14. 'LongestRun_mi'
15. 'SkiableTerrain_ac'
16. Snow Making_ac'
17. 'daysOpenLastYear'
18. 'yearsOpen'
19. 'averageSnowfall'
20. 'projectedDaysOpen'
21. 'NightSkiing_ac'

After exploring many Amenities / features we determined there was a need to find the correlations between different factors. We generated a correlation heatmap (**Fig. 3**) below with the following code:

```
ski_data['resort_skiable_area_ac_state_ratio'] =
ski_data.SkiableTerrain_ac / ski_data.state_total_skiable_area_ac
ski_data['resort_days_open_state_ratio'] =
ski_data.daysOpenLastYear / ski_data.state_total_days_open
ski_data['resort_terrain_park_state_ratio'] =
ski_data.TerrainParks / ski_data.state_total_terrain_parks
ski_data['resort_night_skiing_state_ratio'] =
ski_data.NightSkiing_ac / ski_data.state_total_nightskiing_ac

ski_data.drop(columns=['state_total_skiable_area_ac', 'state_total_days_open',
                       'state_total_terrain_parks', 'state_total_nightskiing_ac'], inplace=True)
int_or_float = ski_data.select_dtypes(include=['int', 'float'])
plt.subplots(figsize=(12,10))
sns.heatmap(int_or_float.corr())
```

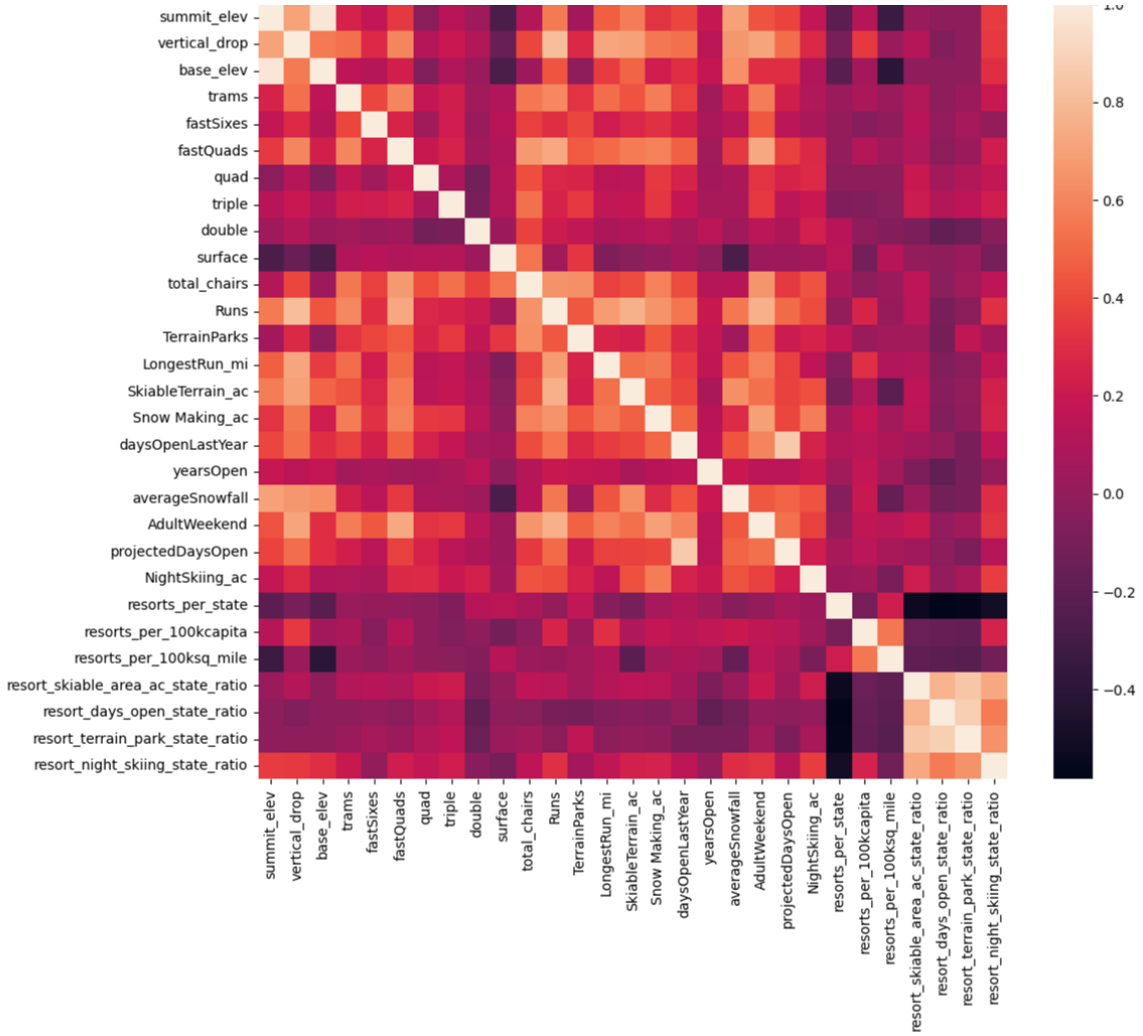

Fig. 3: Correlation Heatmap between all Columns from the file ski_data_cleaned.csv and also state level features.

Many relationships were uncovered that in some cases we used get a sense if a factor that drives ticket price is strongly correlated with other factors. Arguably, what is most important are the correlations with the AdultWeekend price. Such as:

1. Vertical Drop
2. Trams
3. Fast quads
4. Total chairs
5. Runs
6. Longest run
7. Skiable terrain
8. Snow making
9. Days Open per

From here we wanted to establish the relationships between different amenities and price in a more concrete way. And there were concerns about overfitting the data. We partitioned

the data into a 70/30 training/testing split for machine learning as a complete random split. We then calculated the coefficient of determination that is to say we calculated R^2, which is a measure of the proportion of variance that is predicted by our model vs. the total sum of squares(error). A value of 0 would mean there was no difference.

The total sum of squares (error), can be expressed as

$$SS_{tot} = \sum_i (y_i - \bar{y})^2$$

The above formula should be familiar as it's simply the variance without the denominator to scale (divide) by the sample size.

The residual sum of squares is similarly defined to be

$$SS_{res} = \sum_i (y_i - \hat{y})^2$$

where $\hat{y}$ are our predicted values for the depended variable.

The coefficient of determination, R^2, here is given by

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Equation 1: Coefficient of Determination Calculation

Interestingly, we did a completely random split. Although, what might have been interesting is would have been to distribute states to the training set vs. the test set using an approach analogous to “similar subject" methodology or "matched pair design" where each participant is paired with another participant who has similar characteristics, essentially acting as their own control. Although, this would have to be considered with a grain of salt because we also would not want to overfit the data by essentially making the training data more similar to the test data. On the other hand, what might be interesting would be to take states with a many resorts and then to essentially select some portion of the ski resorts in each quartile. Consistent with 70/30 test/train split we could essentially take 1 of every 3 resorts in each quartile. Although we might consider for example if we have 3 resorts per quartile we simply take the one with the mid price of the AdultWeekend lift ticket for that quartile for that state.

The following code was used.

```
def r_squared(y, ypred):
  """R-squared score.

  Calculate the R-squared, or coefficient of determination, of the input.
  Arguments:
  y -- the observed values
  ypred -- the predicted values
```

```
    """
    ybar = np.sum(y) / len(y) #yes, we could use np.mean(y)
    sum_sq_tot = np.sum((y - ybar)**2) #total sum of squares error
    sum_sq_res = np.sum((y - ypred)**2) #residual sum of squares error
    R2 = 1.0 - sum_sq_res / sum_sq_tot
    return R2
```

We used a standard scalar to scale each feature to zero mean a unit variance using the following code:

```
#scaler = StandardScaler()
#scaler.fit(X_tr)
#X_tr_scaled = scaler.transform(X_tr)
#X_te_scaled = scaler.transform(X_te)
```

We then generated a linear regression from the standard scalar.
From there a machine learning model was applied to make predictions :

```
#Call the `predict()` method of the model (`lm`) on both the (scaled) train and test data
#Assign the predictions to `y_tr_pred` and `y_te_pred`, respectively
#y_tr_pred = lm.predict(X_tr_scaled)
#y_te_pred = lm.predict(X_te_scaled)
```

The models performance was assessed according to the following code:

```
# r^2 - train, test
median_r2 = r2_score(y_train, y_tr_pred), r2_score(y_test, y_te_pred)
median_r2
```

This model essentially calculates the coefficient of determination from the training data against the predicted data and the test data and the predicted data. The regressor was able to explain more than 80% of the variance from the training data and more than 70% of the variance from the test data. Lower values for the test suggested overfitting was occurring.

We further assessed with the code below that using mean absolute error in place of the coefficient of determination.

```
# MAE - train, test
median_mae = mae(y_train, y_tr_pred), mae(y_test, y_te_pred)
median_mae
```

(8.547850301825427, 9.407020118581315)

Thus, we are able to estimate the ticket price within $8.60 and $9.40 of the actual price. Although there were concerns of overfitting the data. Thus, we choose to further define the linear model by adding a step in our pipeline called SelectKbest passing in the argument to use the f_regression score function. Our pipeline performs according to the following:

SimpleImputer → StandardScaler → SelectKBest → LinearRegression

Although, mean absolute error worsened. We wondered if we set the number of best features to select to 15 if that might help. While, it improved things it did not improve things over the pipeline that does not use SelectKBest. But also, we must take things with a grain of salt as the ski resorts themselves do not have an exact science in setting ticket prices so the model could be no better at predicting ticket prices than the science (Black Box = pricing strategy) that goes behind setting the price across all ski resorts.

Still, we want to do the best we can because the average of the ticket pricing strategy should indeed yield the best model. Arguably a model that would actually inform resorts on how to set their ticket price. There are obviously a few things we are missing such as what is the view like, is their a large investment in the surrounding grounds or is the surrounding nature impressive.

Generally, there is a concern that by continually modifying the k values that you might run into an overfitting problem. The solution to this is the following: We can use an approach called _cross-validation_ where the training set is partitioned into k folds and the model is trained on k-1 folds. We then want to see that if by excluding a fold does our performance improve. We evaluate this against every fold. Interestingly, it might also be worth taking the correlation analysis between the folds for those folds that are not correlated with each other but correlated with price it might perhaps be worth trying to have a preset weight.

One can image that if several amenities that are correlated with each other and each of those amenities are correlated with ticket price a ski resort might have a smaller increase in price by having the second of the two amenities in comparison to having just the one amenity. As an example, if longest run and vertical drop are both correlated with price. Perhaps they are each could be worth as much as an additional $10 above the mean. But having both would not necessarily increase the price above mean by $20. Perhaps having both might increase the mean by $14. Another amenities such as fastfour or number of chairs for example might not be correlated with longest run and in that case increasing the number of chairs to a high value might very well increase the average ticket price by $7. And so having a top longest run and top number of chairs should be worth more perhaps an additional $16. That is not to say that the model itself does not already do this.

Cross validation supported that setting k = 8 was the ideal value. Where the model's performance peaks and then worsens as you increase k to greater than 8. Below are the weighted parameters:

vertical_drop	10.767857
Snow Making_ac	6.290074

total_chairs	5.794156
fastQuads	5.745626
Runs	5.370555
LongestRun_mi	0.181814
trams	-4.142024
SkiableTerrain_ac	-5.249780

Vertical_drop is most influential and longest run is least influential. Interestingly, SkiableTerrain_as has a negative influence on ticket price. Surprisingly, trams have a negative influence. It almost makes no sense. What would be interesting would be to repeat the model removing the most influential factors such as vertical_drop, Snow_making etc. From the exploratory_data analysis worksheet number trams directly influences ticket price. This makes sense as trams are very expensive and something you see at the top results. What might be happening here is amenity overeffect where ticket price can no longer increase once you reach a certain number of amenities. What might happen instead then is we need to alleviate the effect of having too many amenities by taking a commonality and then subtracting from the total. What does this tell us? Well that there is a point where you can no longer increase ticket price by adding most but not necessarily all amenities. In that case it would generally be an effective waste of money to add amenities. Also, trams are not at most ski parks and the most we see are 3 trams. A ski park with 3 trams has a minimum AdultWeekend ticket price of $115 and if the ski park has 1 tram the minimum price is $62. There are only 17 ski parks with 1 trams and a total of 21 ski parks with 1 or more trams. Thus, because there is so little data on trams it may not sufficient to be correctly interpreted by the machine learning model.

SkiableTerrain suffers from the same problem as Trams. At most parks SkiableTerrain is concentrated to be below 1000 ac. There are only 15 parks with SkiableTerrain greater than 2000 ac. But also, once you exceed a specific amount of SkiableTerrain such as about 500_ac there is no longer any data that supports that increasing SkiablerTerrain equates to an increase in price because the price trend is effectively flat above 500 – 1000ac. SkiableTerrain is strongly positively correlated with vertical drop, number of runs and fast quads all of which influence price and so Skiable terrain while really should be 0 above 500ac has a negative value to negatively compensate for other factors that is correlated with that continually drive price but are also correlated with each other and are interpreted to have a greater collective influence than they have—their collective influence is less due to their correlation with each other—that is then corrected by a negative skiableTerrain wieghting. But agreeably correlation does not mean that having one makes that other worth. It just

means they tend to increase together. Although, in some cases having three makes the fouth less value add.

SkiableTerrain_ac	AdultWeekend
8464.0	88.00
5517.0	179.00
4318.0	99.00
3500.0	159.00
3000.0	115.00
2900.0	81.00
2614.0	116.00
2602.0	90.00
2600.0	99.00
2600.0	64.00
2527.0	158.00
2500.0	125.00
2400.0	49.00
2325.0	62.00
2026.0	169.00

Next we turned to a Random Forest Model using a Random Forest Regressor: This regressor is a series of methods for completing regression analysis using decision trees. By using a seed value (that could be any number) we ensure that the split between training and test data is always the same.

```
#Define a pipeline comprising the steps:
#SimpleImputer() with a strategy of 'median'
#StandardScaler(),
#and then RandomForestRegressor() with a random state of 47
RF_pipe = make_pipeline(
  SimpleImputer(strategy='median'),
  StandardScaler(),
  RandomForestRegressor(random_state=47)
)
```

Next we used the cross validation feature at 5-fold. This takes the training data and splits it into 5 parts or folds where 4 of the parts are used for training and 1 for testing. The process is repeated 5-fold and is considered a more robust approach for a machine model before it is finally applied to the test dataset. The results of the randomforestregressor are the following

The most influential amenities for pricing tickets are in order the highest importance first:

1. fastQuads - Highly Influencing

2. Runs - Highly Influencing
3. Snow Making – Moderately Influencing
4. Vertical Drop – Moderately Influencing
5. Skiable Terrain – Minorly influencing
6. Total Chairs – No more influencing than the subsequent several features

From here we proceeded to select the better of the two models:

We compared the performance of our Linear Regression Model and Random Forest Regression Model. The Random Forest Model outperformed the Linear Regression model.

We took a second look at all these amenities in a histogram for all resorts and placed a marker for where Big Mountain Fell. We also considered Big Mountain in the context of the state distribution for ticket price. Big Mountain Resort boasts a lot of impressive amenities and is by far the finest in Montana. They are at the upper echelon for every one of these amenities except trams.

This brought up the interesting question: Should Big Mountain invest in a tram?

No investing in a tram is not on Big Mountain's shortlist of options which includes:

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

Looking back on our earlier analysis without doing any further analysis it seems at face value that increasing the longest run, increasing vertical drop and or adding a little snow making cover is probably not worth the investment. Further, permanently closing down up to 10 of the least used runs seems like a great idea to increase margin.

Really the burning question is should Big Mountain install a tram? And what budget for a tram could they support. The tram budget is quite an important question because if the tram needs to have its cost recovered in 10 years factoring interest, then does the model justify that?

It is worth looking into the budgets for the mountains that have one tram since there are only 17 of them with 1 tram. We could then consider if the budget difference also speaks to the price of the AdultWeekend lift ticket. The cost of a tram can vary from $5 million to $50

million. Some basic research supported the Tram Costs provided below for each of the resorts with 1 tram. We also provde the AdultWeekend ticket prices for the 1st and 2nd most expensive resorts in the State referred to as "1st/2nd" and also the number of resorts in the state

Resort	AdultWeekend	State	1st /2nd	#Resorts	Tram Cost
Deer Valley Resort	169.00	Utah	$169/$125	12	$12.5M
Copper Mountain Resort	158.00	Colorado	$179/$158	14	$10M
Sugar Bowl Resort	125.00	California	$159/$125	14	$7M
Snowbird	125.00	Utah	$169/$125	14	$15M
Taos Ski Valley	110.00	New Mexico.	$110/$80	9	$10M
Sunday River	105.00	Maine	$105/$99	8	$8M
Crystal Mountain	99.00	Washington	$99/$95	8	$9M
Whiteface Mountain Resort	96.00	New York	$96/$95	33	$7M
Jay Peak	89.00	Vermont	$119/$119	10	$7.5M
Gore Mountain	88.00	New York	$96/$95	33	$8M
Alyeska Resort	85.00	Alaska	$85/$53	3	$11M
Lutsen Mountains	84.00	Minnesota	$84/$60	14	$6M
Big Mountain	81.00	Montana.	$81/$67	12	____
Mountain Creek Resort	79.99	New Jersey	$79.99	1	$6M
Cannon Mountain	79.00	New Hampshire	$99/$93	14	$5.5M
Ski Apache	74.00	New Mexico	$110/$80	9	$7M
Belleayre	72.00	New York	$96/$95	33	$5.5M
Silver Mountain	62.00	Idaho	$81/$70	10	$10M

Considering this data it seems that Big Mountain resort could charge upwards of $95 - $100 per ticket with installation of a $8 - $12 Million dollar tram. A tram appears to be the mark of a premier ski resort. There is that burning questions as to whether the second most expensive ski resort in the state has an influence on the first most. This analysis supports that there may be some maximum different that a state can tolerate. Perhaps if another resort in the state of Montana was to also install a tram such as Red Lodge Mountain that an even higher ticket price might be supported. In fact, a potential strategic acquisition would be for Big Mountain to acquire Red Lodge Mountain because it would support a greater increase in ticket price if both mountains installed a tram than if only Big Mountain installs a tram. This dual strategy could result a massive economic stimulus for the state of Montana and furthermore it seems possible that Montana could become known as among the premier skiing states from this strategy. I see no reason why Big Mountain could not see an increase of at least 100,000 additional lift tickets sold annually within 2 – 3 years from this strategy depending on the marketing budget and pricing the tickets at $110 per day.

It may be very difficult for the model to predict the value of a tram because of the challenge that only 7% of resorts have a tram to begin with. Comparing to resorts with 0 trans may not

be effective for a random forest model. This is the real issue with the model. Other Amenities are providing a stronger signal to the model because generally trams are only found in premier ski resorts that have so much to offer already and the tram feature which is critical is getting buried because the random forest model because the 0 value for 93% of resorts makes correlation very challenging. The decision tree will also be ineffective for trams due to suboptimal splits. Additionally, the 5-fold cross validation probably make the situation worse for trams. Because the excluded 20% splits may have only a single data point with a tram.

For the most part Trams are found almost exclusively in ski resorts in the top quartile of Adult Weekend price for each state where ski resorts have a tram. Because Big Mountain is a premier mountain for its state. Completing the same modeling from the very beginning on only ski resorts in the top quartile for each state would allow us to model the influence of the tram with greater precision.

From here we chose to repeat the random forest regressor on only the ski resorts falling in the top quartile of AdultWeekend lift ticket price with the hope that we could isolate the influence of trams and possibly skiableterrain_ac on price.

When AdultWeekend is set to the 75th percentile 17 of the 75 data points had trams. The randomforestregressor shed no further light on trams with respect to this subset of the data. And there were no new learnings from the randomforestregressor on the entire dataset. The Linear Regression Model on the other hand did prove to be enlightening:

https://github.com/rswartz2/DataScienceGuidedCapstone/blob/master/04_preprocessing_and_training_Roger_B_Swartz_Quantile75.ipynb

Original DataSet	
vertical_drop	10.767857
Snow Making_ac	6.290074
total_chairs	5.794156
fastQuads	5.745626
Runs	5.370555
LongestRun_mi	0.181814
trams	-4.142024
SkiableTerrain_ac	-5.249780

AdultWeekend_75thQuantile by State Dataset:	
Runs	10.181102
summit_elev	8.365763
fastQuads	8.010931
trams	4.447020
total_chairs	1.633524
daysOpenLastYear	-1.164149
vertical_drop	-11.014030

Now trams does shift to 4th position and has an overall change of +8.5. The change is quite important here because it is telling us what is relevant at the upper echelon ski mountains. It is also telling us that it is the most significant difference between the two data sets. That is it has the greatest positive change of all that factors. And the tram must have this influence when it is only found in 20% of the ski resorts at the 75th quantile for price. It must overcome

the effect of being buried and obscured by the effect of other amenities. It may very well be that fastQuads increase at the 75thQuantile borrows some of the influence of trams because there is an abundance of a correlation already. Trams are strongly correlated with both runs and fast quads.

But obviously since trams are a costly installation and are considered the mark of a great ski resort why do they appear to have a negative influence on the entire dataset. This has do to with other collective factors. As we discussed earlier one can image that if several amenities that are correlated with each other and each of those amenities are correlated with ticket price a ski resort might have a smaller increase in price by having the second of the two amenities in comparison to having just the one amenity.

Revisiting the entire Dataset: Vertical_drop is most influential and longest run is least influential. Interestingly, SkiableTerrain_as has a negative influence on ticket price. Surprisingly, trams have a negative influence. It makes no sense. What would be interesting would be to repeat the model removing the most influential factors such as vertical_drop, Snow_making etc. From the exploratory_data analysis worksheet number trams directly influences ticket price. This makes sense as trams are very expensive and something you see at the top results. What might be happening here is amenity overeffect where ticket price can no longer increase once you reach a certain number of amenities. What might happen instead then is we need to alleviate the effect of having too many amenities by taking a commonality and then subtracting from the total. What does this tell us? Well that there is a point where you can no longer increase ticket price by adding amenities. In that case it would be an effective waste of money to add them.

Also, trams are not at most ski parks and the most we see are 3 trams. A ski park with 3 trams has a minimum AdultWeekend ticket price of $115 and if the ski park has 1 tram the minimum price is $62. There are only 17 ski parks with 1 tram and a total of 21 ski parks with 1 or more trams. Thus, because there is so little data on trams it may not sufficient to be correctly interpreted by the machine learning model.

So lets try to explain what is happening at the upper echelon of ski resorts. When you get to the level of the best the rules for pricing change a little. Why, well what does someone want at the more expensive resorts. Well first they probably want an exceptional view that is generally achieved with a higher summit. They want the option for more runs than they can do because they are looking for the effect that the ski resort has so much terrain to explore that there is no end in sight and there is always something new to discover that they need to come back many times which a ski resort needs repeat customers to stay in business and when the prices are high repeat customers want to see at least some new runs virtually every time they ski. Once you have everything and you want to have the final wow effect you install

a tram. Because it offers an additional experience that goes beyond the skiing. If you were not skiing you would not pay to go up a chair lift in the cold air to the top of the mountain but you would pay for a tram. Once a ski mountain / resort reaches a certain level a tram offers additional value to customers when no other amenity can and can be viewed as a pay for additional feature included in every ticket. A Tram may even have a health benefit allowing skiers to warm up between some of their runs reducing susceptibility to getting a cold and maximize their amount of enjoyment per unit time. As if a skier decides to take a break and warm up inside that generally involves atleast a 30 - 45 minute detour. And while the first time going in to warm up is a novelty the second time is a drag. Skiers may even assign a dollar value to the lost time.