**Big Mountain Resort:**

Big Mountain Resort, a ski resort located in Montana. Big Mountain Resort offers spectacular views of Glacier National Park and Flathead National Forest, with access to 105 trails. Every year about 350,000 people ski or snowboard at Big Mountain. This resort has increased the chair lift which increases operating cost too much. In order to increase the profit, we need to fix the ticket price by comparing the features from the other 330 resorts in US.

By comparing the features from the other resorts and to analyze the profit based on the run time, ticket fare and some more features. With these data we need to analyse and will fix the best ticket price for Big Mountain Resort.

From the given CSV file by the stake holders, we need to go with the remaining process. Will be starting with the process of data wrangling initially.

**Data Wrangling Summary:**

With this dataset, started with the initial phase of importing the necessary packages and libraries. Once imported then loaded the dataset from the drive. To understand the dataset, listed the top 5 rows with 'head' method. We have 330 entries in the resort data. Then we need to check that is there any null values in the BigMountain Resort. After that we need to check the total missing values in the dataset. As the region and state columns has some similar values, so handled the duplicate values with the use of various methods like sum, value\_counts etc. Also checked the distribution of states and region through bar plot. As our main idea is to fix the average ticket price, so handling of Average Weekday and Weekend price is more important. With the help of melt function just checked the Ticket price for weekday and weekend with particular state. It made so easier in checking the ticket prices. Then handling the null values of ticket price for weekday and weekend column. After that handled the outliers for each column which will make a greater impact during the modeling. Found that fastEight column has more null values and dropped the same which has not much impact. Found an huge outlier in 'SkiableTerrain\_ac' column and 'yearsOpen'. Then checked with the other values and handled the same. Also taken the aggregate of 'daysOpenLastYear', 'TerrainParks', and 'NightSkiing\_ac' for checking out the features. Just loaded the population dataset of US, and to compare the ticket sold and various features. Then checked out the Adult weekday and weekend through scatter plot to see the difference of each state prices. After that weekday and weekend prices of Montana state to check the ticket price of Montana state which helps to fix the ticket price for Big Mountain Resort. Dropped the null values in Adultweekday and weekend column of Montana state. Later checked the non-null values of each row and found there is not much difference with that. Once done with the data wrangling part, pushed the cleaned data and summary data in two different csv files.

**Exploratory Data Analysis:**

We've seen that at the end of data wrangling session we have converted two dataframes into two different csv files. From that we have loaded 'data\_cleaned.csv' to process this EDA. Once loaded we have started with the initial process of checking the information of the dataset. By using head function will be seeing the overview of the dataset. Later we have loaded 'state\_summary.csv' as well. Just checked the information and overview of the same. Will explore this state information dataset. From the state dataset, we will take one the state values on one dataset and we will start processing with that variable further to avoid issues in original dataset. Initially we will be checking with total state area of the dataset to identity which are the states stand top in total state area. Afterthat we will be checking out total population, total resorts in the state, total skiable area, total night skiing area and total days open. From these data will be comparing where each state stands and which tops in all these. After seeing these data we have found that there is no need that big states need to be more populous and states with more resorts doesn't need to have larger total skiing area and state with most total days opened doesn't meant to have more number of resorts. Then scaling the data of state population and state area sq miles with 100\_000 scaling. With the scaled data finding the top states by resort density and found Vermont has high per capita income and New Hampshire top in more in area. New Hampshire and Vermont tops in both. Now we will be scaling the states data and keeping it in column. While checking the mean it is close to zero. While seeing Standard deviation its 1. After scaling then we will be fitting to transform using PCA. Now will be checking the average ticket price by state to see the top earnings. Creating a new dataframe with PCA values, Weekend price and Quartiles. From the histogram of the Weekend price we will not able to see the difference, so to see that we will have one more column by seeing its quartile.Now see a scatter plot where in we can able to see on which quartile which states stands. With the help of PCA transform we have kept all the state data in a small scale. Will be checking the state summary data of New Hampshire and Vermont data before and after scaled. Now we will merge the state\_summary data into ski\_data to find the best one. Seeing the heatmap of correlation of ski\_data, where ski\_data will have latest ratio taken for skiable area, days open, terrainparks and night\_skiing states. Then will see how the correlation looks, it looks like average price deals with many factors like night\_skiing, chairs count and even more. The scatterplot shows it even better and it shows that the more number of chairs, number of runs increases the revenue. It also says no fast quads limits the ticket price. With these findings will save these data in to another csv for further pre-processing.

**Data Preprocessing and Training:**

Loaded the csv file created from EDA process and checked the features of the dataset. Then will be starting the process of preprocessing, splitting the dataset with Train/Test split. This split will be done for training the model to learn the data. Once the data is trained and the testing can be done with the test split data. From these data, we will be checking the average price of the resorts through mean function initially. Also the metrics can be calculated using predefined methods like R-squared. Then we will be checking out Mean Absolute Error, Mean Squared Error. With the test and train data finding out the r2 score and with that we will be tuning the data with various models. Before training into some models will make the scaling has been done to keep all values in the same metrics. Will be checking the mean and median values through the fit and predict in Linear Regression. From the r2\_score of the Linear Regression it says 81% of accuracy from the train data and 72% accuracy from the test data. There is no much difference with mean and median test. To avoid repeating the steps, will be using cross validation method to check the scores. Once done with the linear regression, will be using grid search with hyperparameters. Then we will be using Random Forest to check out the accuracy. Fit and assess performance using cross validation techniques, Random forest has increased the cv\_scores from 0.63 to 0.70. From these data we will be choosing the RandomForest as the better one.

**Modeling:**

At present, Big Mountain Resort is at 124 and it charges ticket price fare as 81 dollars and our modeling price is 94.22 dollars and it has also suggested that we can increase some features where in it will increase fare even more. After increasing vertical drop to 150feet then we can increase the ticket price by 8.46 dollars and if we increasing snow making area by 2 acres as suggested then we can increase price by even 9.75 dollars more.

Also suggesting business with some 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.

The expected number of visitors over the season is 350,000 and, on average, visitors ski for five days. Assume the provided data includes the additional lift that Big Mountain recently installed.

The model says closing one run makes no difference. Closing 2 and 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.

Although the longest run feature was used in the linear model, the random forest model (the one we chose because of its better performance) only has longest run way down in the feature importance list.

The Modeling has been done and we are now having the modeling ticket price for Big Mountain resort. There is an increase in the price range from 10-15 dollar per ticket. From the data given for fixing the ticket price there is no dependencies.Ticket price is not determined by any set of parameters; the resort is free to set whatever price it likes. However, the resort operates within a market where people pay more for certain facilities, and less for others. Being able to sense how facilities support a given ticket price is valuable business intelligence. Yes from the analysis we can conclude that without any further changes in the facilities we can able to make increase in the price of upto 10-15 dollars from the current price. We can make the necessary changes and we can deploy this model to the cloud and make it available for the business users at anytime. Also if there any combination of parameters change then during the initial stages we will be helping them and will explain the process to perform. Later after deploying it in cloud will provide elevated access to the business users.