Data Analytics Assignment Documentation

Abhinna Manandhar

BSc. (Hons) Computing, Softwarica College of IT and E-Commerce,

Coventry University

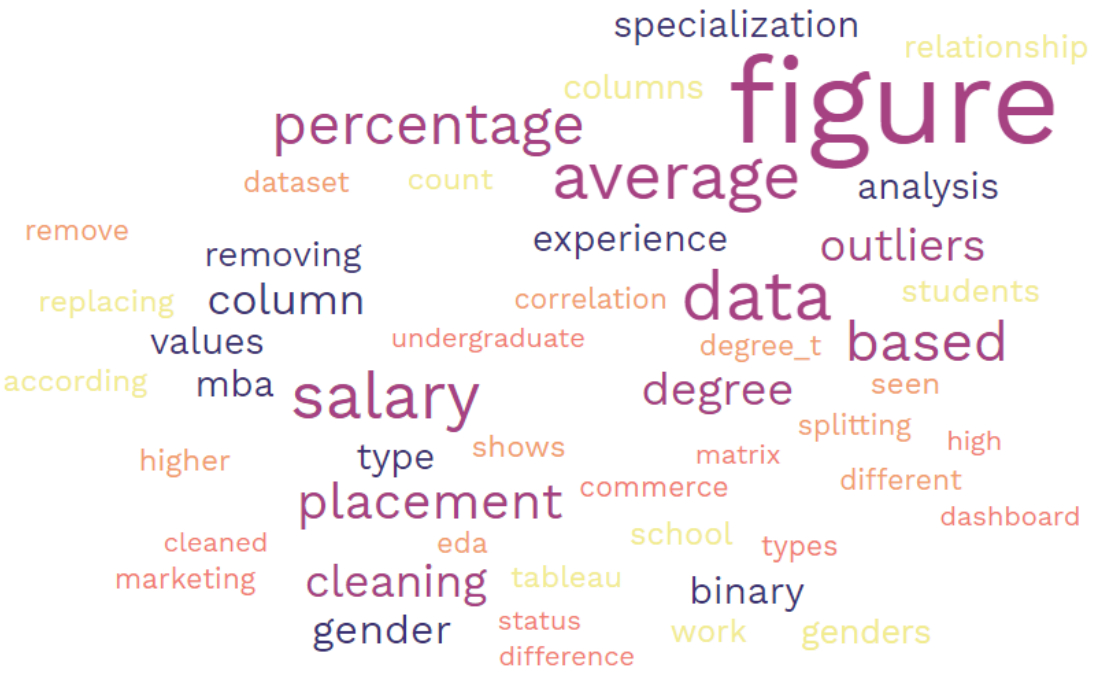
SP6000COM Data Analytics

Nahakul Prasad Jaishi

March-01-2023

Keywords

Figure 1: Keywords



Contents

Table of figures

Introduction

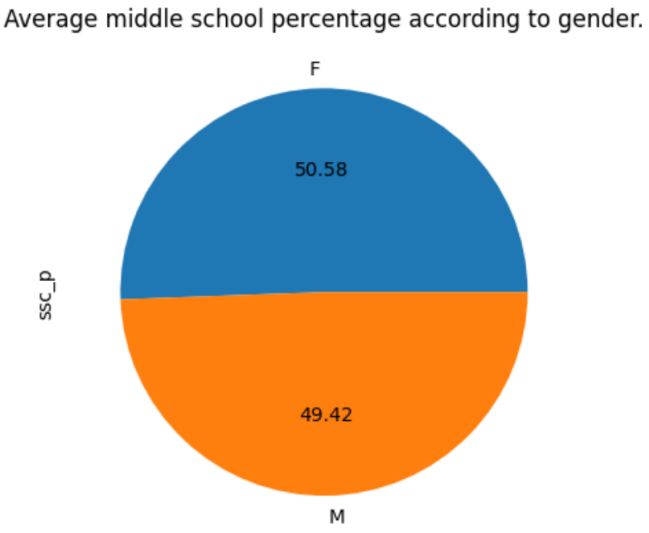
In this documentation we will analyze the provided dataset about the student's education history, and their job placement status, salary and other factors. This will allow us to analyze the scope of different specialization fields and their benefits to make better decisions. We will start by analyzing the data and then continue to clean it. After cleaning the data, we will again perform EDA to compare the changes in the data and generate our desired results and perform visualization using different tools like Tableau.

EDA

Exploratory data analysis helps data analysts to gain more detail about the data and perform further analysis on the data itself.

Performing EDA on our dataset helps in discovering the relationship between different entities and create appropriate matrices accordingly.

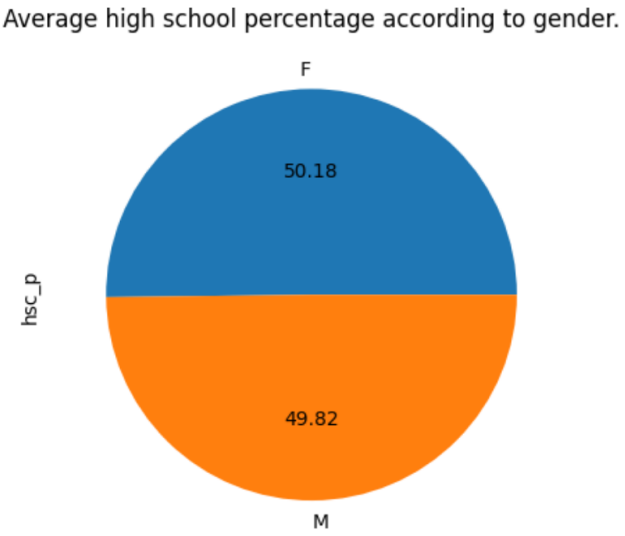
Figure 2: Average middle school percent according to gender.



The above analysis displays the average percentage obtained by each genders in middle school. We can conclude that that the differences in percentage based on genders is significantly low.

Analysis 2

Figure 3: Average high school percentage according to gender.



The analysis of percentage scored in high school is also similar.

Figure 4: Average undergraduate percentage according to gender.

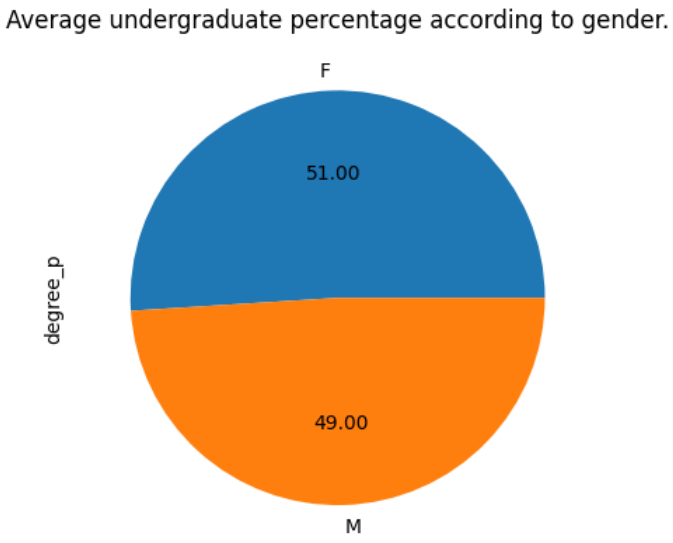
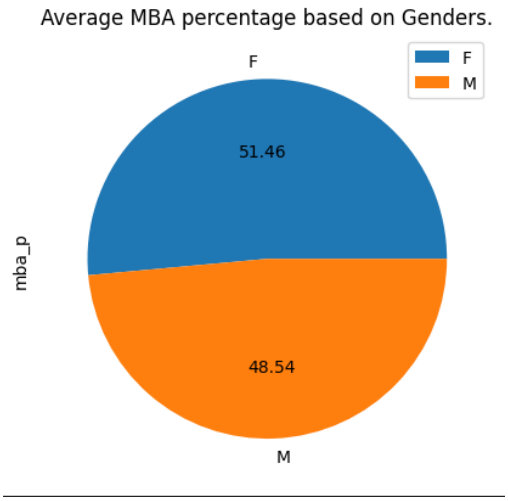


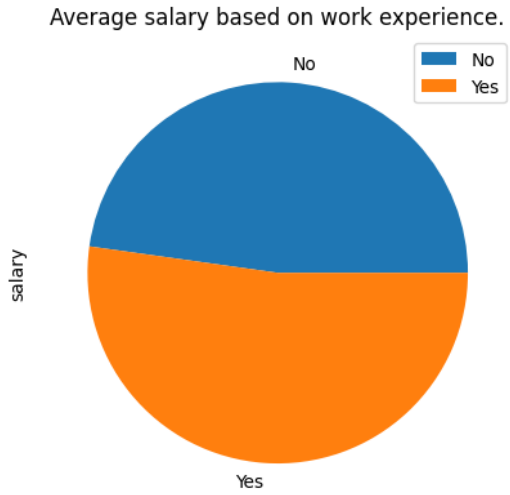
Figure 5: Average MBA percentage based on genders.



The average percentage difference between the genders in undergraduate is only of 1%. And 2.92 in MBA.

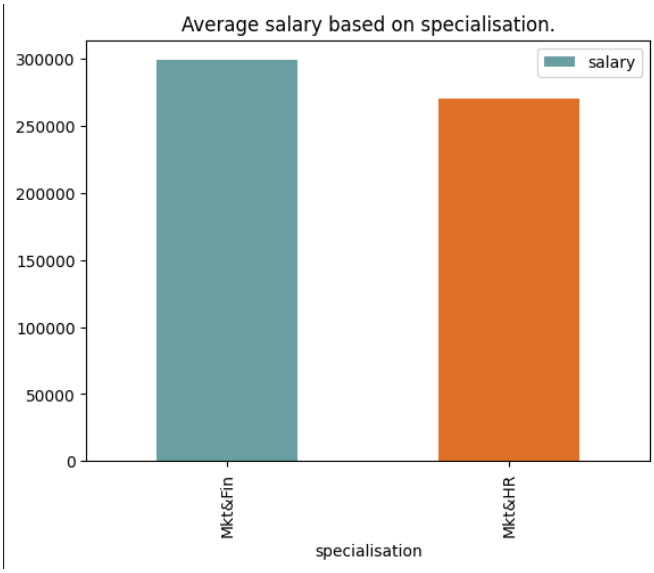
Therefore, we can conclude that gender has very less impact on the average percentage scored in all levels.

Figure 6: Average salary based on work experience.



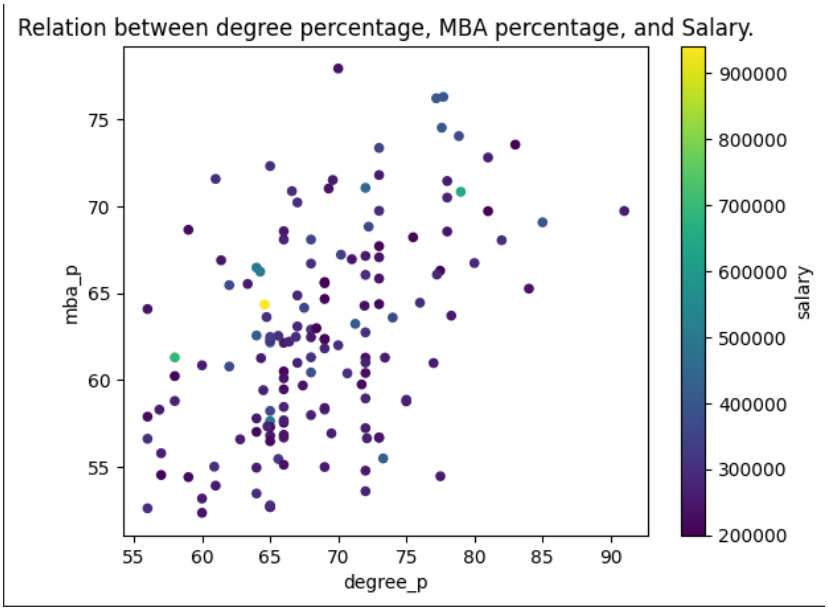
The average salary of an individual based on their work experience has shown a fairly good amount of difference with individuals with previous work experience having more chances of getting higher salary jobs.

Figure 7: Average salary based on specialisation.



Individuals with specialization in marketing and finance have shown to get paid higher salary in jobs than individuals with specialization in Marketing and HR.

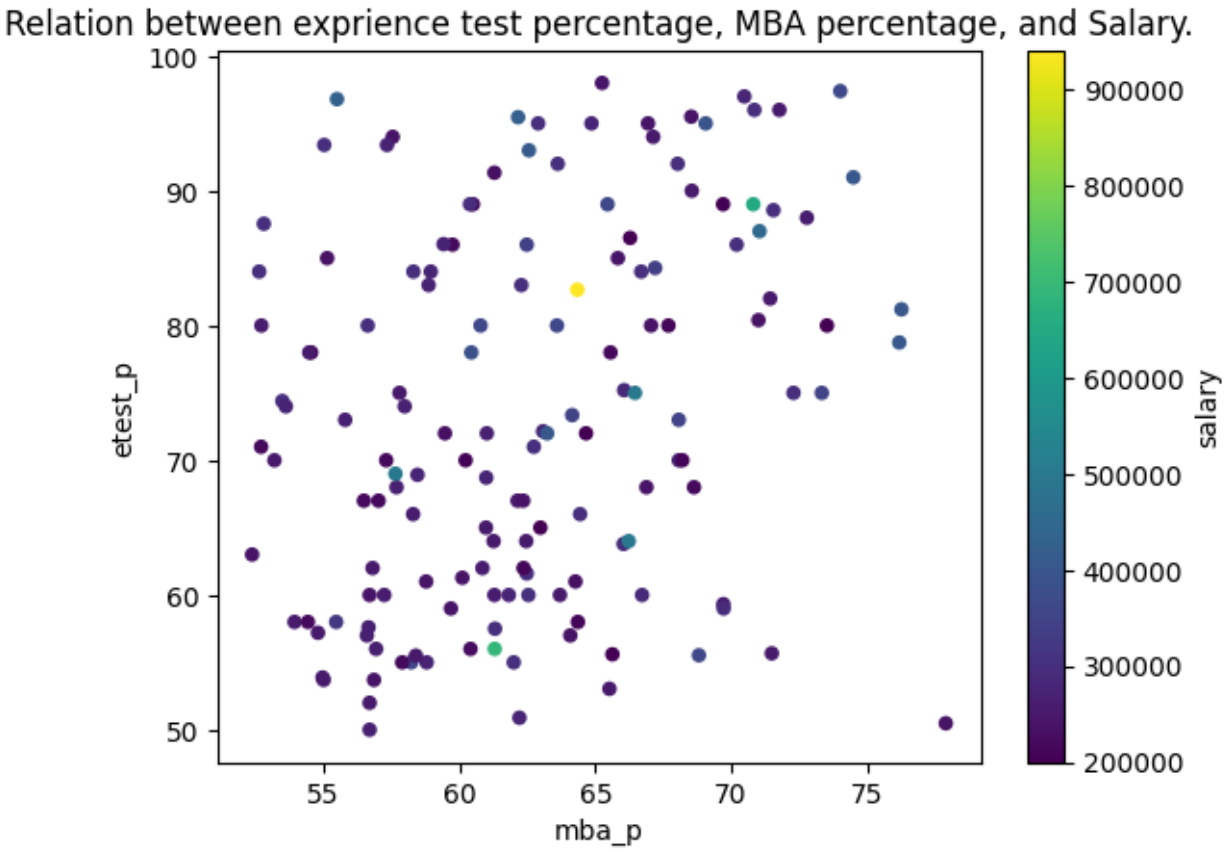
Figure 8: Relationship between degree percentage, MBA percentage and salary.



The percentage obtained by individuals in ther undergraduate degree and MBA have also shown a good relationship with each other with salaries identified by the color range.

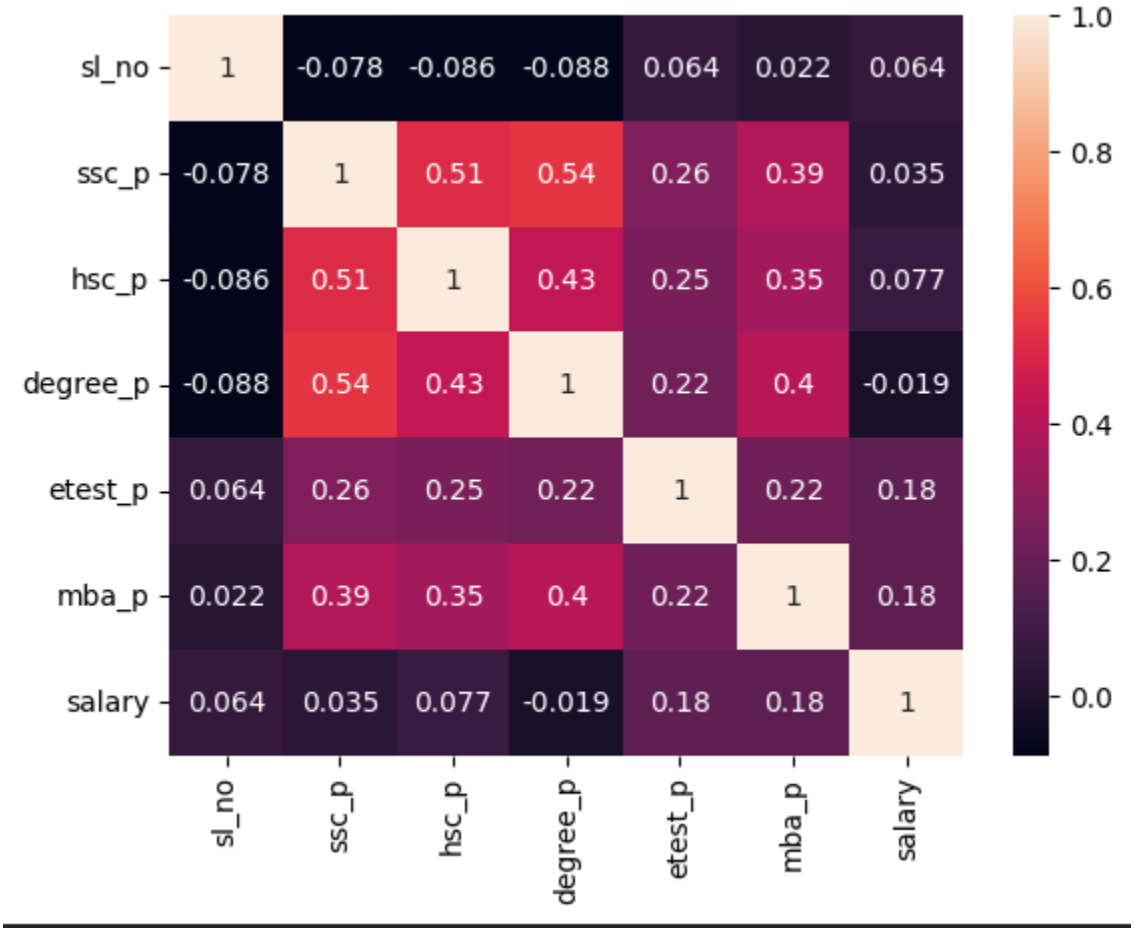
Because the outliers have not been removed currently and the analysis is performed in the raw dataset without any cleaning or manipulation, the salary ranges display a noticeable outlier with color yellow between 60-65 percentage in MBA in the above scatter plot.

Figure 9: Relationship between experience test percentage, MBA percentage, and salary.



The scatterplot displaying the relationship between the experience test percentage, MBA percentage and salary also has a somewhat similar output.

Figure 10: Correlation matrix before data cleaning.



Likewise, the above heatmap visualizes the correlation between all the numeric variables.

Data Cleaning

Data cleaning is the process of filtering our data to further support our analysis. The cleaning process may involve multiple steps. Such as replacing or removing the null values, converting categorical data into binary standards either by simply replacing the values if possible or by splitting the column into multiple columns and then providing binary values.

We start cleaning our data by first removing the “sl_no” column as it is unnecessary in our case.

Figure 11: Removing sl_no column.

```
In [41]: # drop 'sl_no' column as it is unnecessary.
         df.drop('sl_no', axis=1, inplace=True)
```

We then continue to replace the null values in the salary column with 0 as they represent that the students are not placed for any job or are not working.

Figure 12: Replacing NaN values in salary column.

```
In [42]: # replace missing values in 'salary' column with 0
         df['salary'].fillna(0, inplace=True)
```

Since the 'ssc_b' and 'hsc_b' columns simply have two unique values, we replace the categorical data types with binary inficators.

Figure 13: Replacing ssc_b and hsc_b with binary indecators.

```
In [44]: # convert 'ssc_b' and 'hsc_b' columns to binary indicators
         df['ssc_b'] = df['ssc_b'].apply(lambda x: 1 if x == 'Central' else 0)
         df['hsc_b'] = df['hsc_b'].apply(lambda x: 1 if x == 'Central' else 0)
```

For the columns 'hsc_s', and 'degree_t', we added a new binary indicator column. This is because the column consists of more than two unique values. So we cannot directly convert them into binary. Thus, we split the columns into multiple columns named after the unique values and provide them with binary values accordingly.

Figure 14: Splitting hsc_s column.

```
In [45]: # Create three new columns for the three unique categories in 'hsc_s'
         df['hsc_s_Commerce'] = df['hsc_s'].apply(lambda x: 1 if x == 'Commerce' else 0)
         df['hsc_s_Science'] = df['hsc_s'].apply(lambda x: 1 if x == 'Science' else 0)
         df['hsc_s_Arts'] = df['hsc_s'].apply(lambda x: 1 if x == 'Arts' else 0)

In [46]: # Drop the original 'hsc_s' column
         df.drop('hsc_s', axis=1, inplace=True)
```

Figure 15: Splitting degree_t column.

```
In [48]: # Create three new columns for the three unique categories in 'degree_t'
         df['degree_t_Sci&Tech'] = df['degree_t'].apply(lambda x: 1 if x == 'Sci&Tech' else 0)
         df['degree_t_Comm&Mgmt'] = df['degree_t'].apply(lambda x: 1 if x == 'Comm&Mgmt' else 0)
         df['degree_t_Others'] = df['degree_t'].apply(lambda x: 1 if x == 'Others' else 0)

In [49]: # Drop the original 'degree_t' column
         df.drop('degree_t', axis=1, inplace=True)
```

We then continue to implement binary indicators in as many columns as possible.

Figure 16: Implementing Binary Indicators.

```
In [52]: # convert 'workex' column to binary indicator
         df['workex'] = df['workex'].apply(lambda x: 1 if x == 'Yes' else 0)

In [55]: # convert 'specialisation' column to binary indicator
         df['specialisation'] = df['specialisation'].apply(lambda x: 1 if x == 'Mkt&Fin' else 0)

In [56]: # convert 'status' column to binary indicator
         df['status'] = df['status'].apply(lambda x: 1 if x == 'Placed' else 0)
```

Removing Outliers

Outliers are special types of data that fall outside the range of the variable. Outliers have a significant impact on our analysis as they are capable of changing our visualizations and machine learning models completely. Detecting and removing outliers are a vital part of data cleaning as they enable us to produce more accurate results from our analysis.

The columns containing quantitative data are “salary”, “ssc_p”, “ssc_p”, “degree_p”, “etest_p” and “mba_p”. Boxplot is one of the best visualization techniques to identify outliers in a dataset. Therefore, we move forward by producing boxplots for these quantitative data.

In order to remove the outliers, we have first created a function that finds and remove the outliers for any given quantitative column. This is done by calculating the first and the third quartile and then calculating the interquartile range. Then by further calculating the upper and lower bounds, we can remove the outliers that fall outside the bounds.

Figure 17: Function to remove outliers.

```
In [76]: # Function to detect and remove outliers
         def remove_outliers(df, column):
             Q1 = df[column].quantile(0.25)
             Q3 = df[column].quantile(0.75)
             IQR = Q3 - Q1
             lower = Q1 - 1.5 * IQR
             upper = Q3 + 1.5 * IQR
             df = df[(df[column] >= lower) & (df[column] <= upper)]
             return df
```

Figure 18: Removing outliers

```
In [77]: # apply the function to each numerical column in the dataset
         num_cols = ['ssc_p', 'hsc_p', 'degree_p', 'etest_p', 'mba_p', 'salary']
         for col in num_cols:
             df = remove_outliers(df, col)
```

Figure 19: Boxplots before data cleaning.

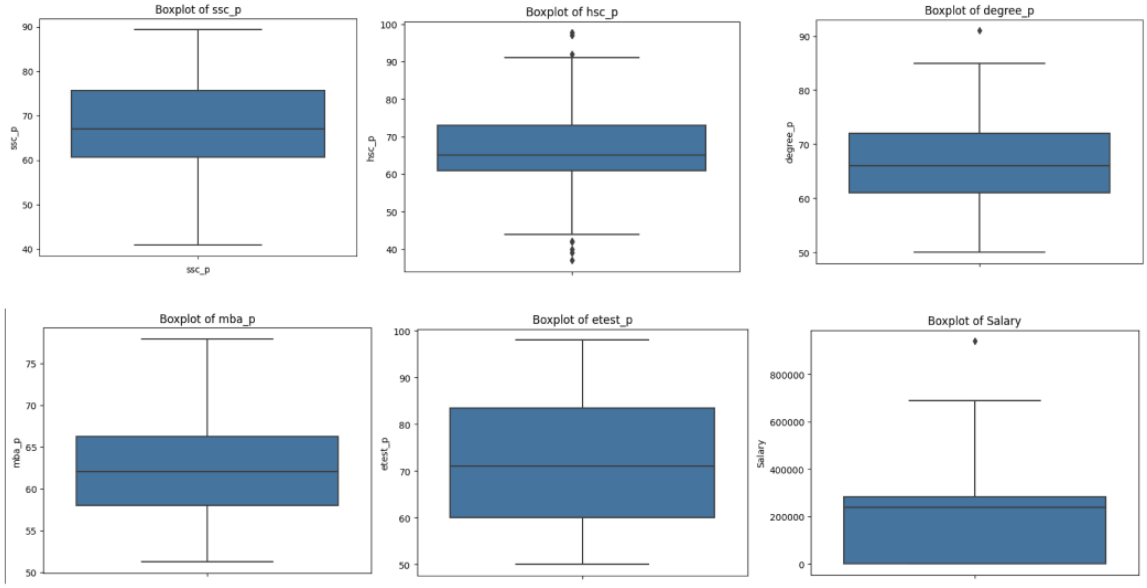
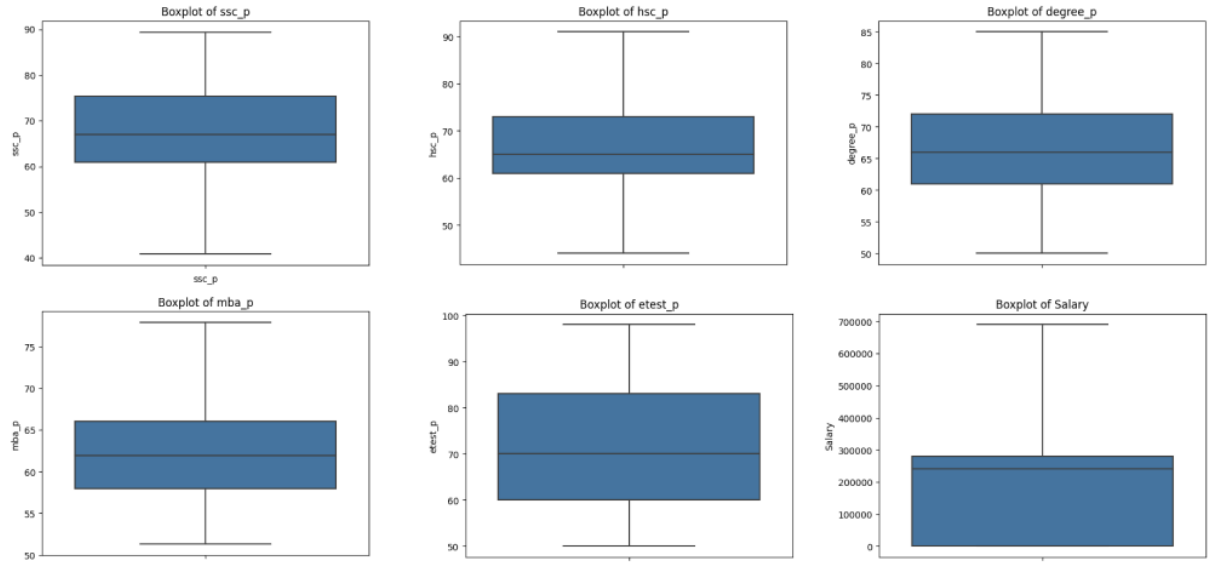


Figure 20: Boxplot after data cleaning.



We can now see that all the columns don't have any outliers in them. And with that we can now say that our data has been cleaned.

EDA on cleaned data.

By performing EDA on our cleaned data, we can now analyze the differences in the dataset before and after cleaning. We can also look for any visible patterns in the dataset by performing EDA.

Figure 21: Correlation matrix

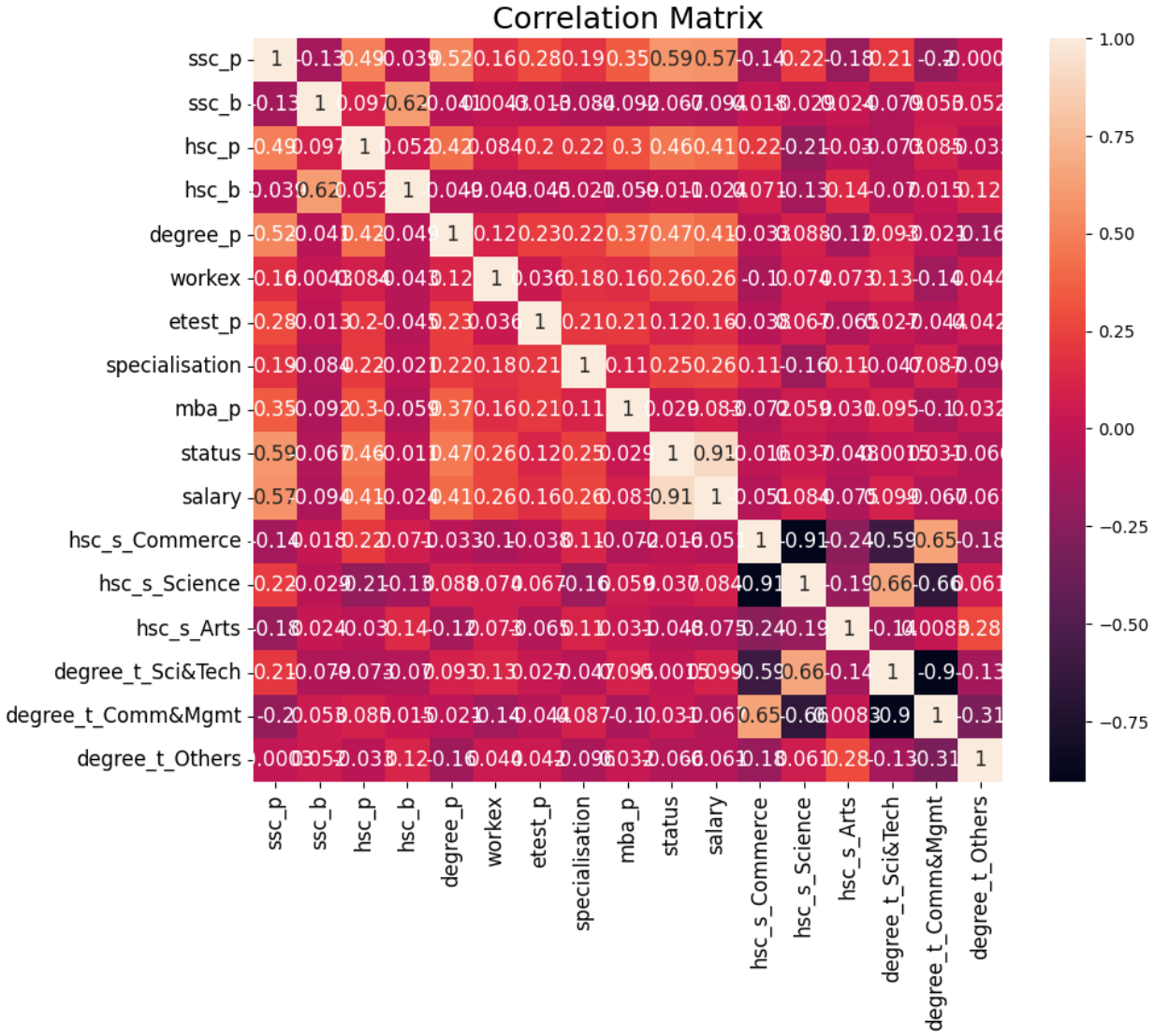
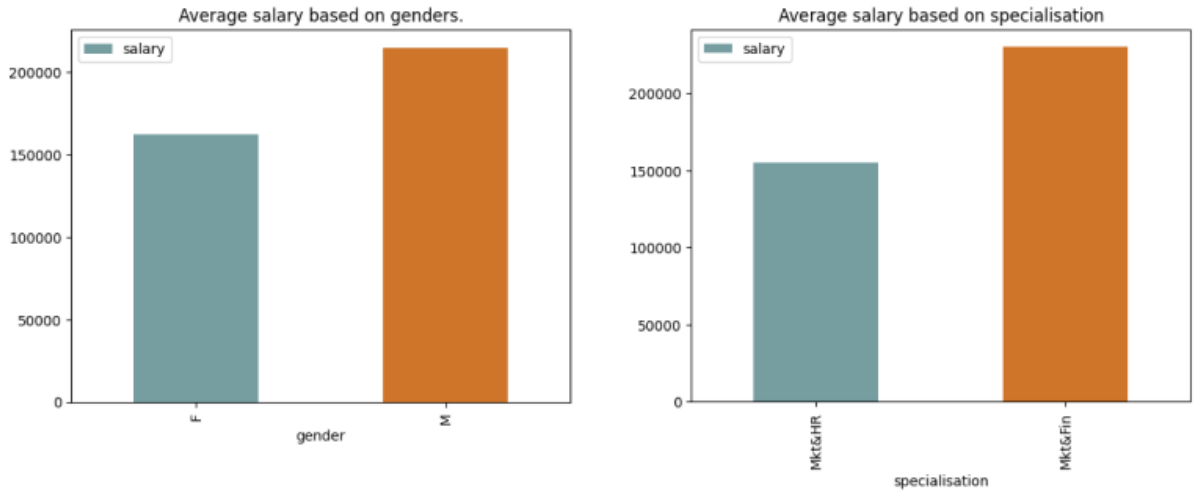
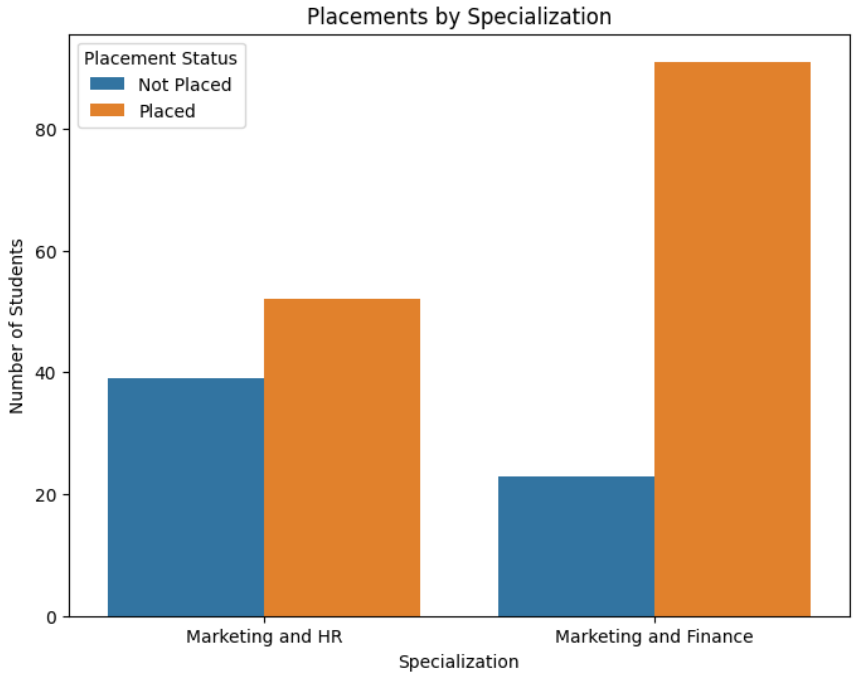


Figure 22: Average salary based on different factors.



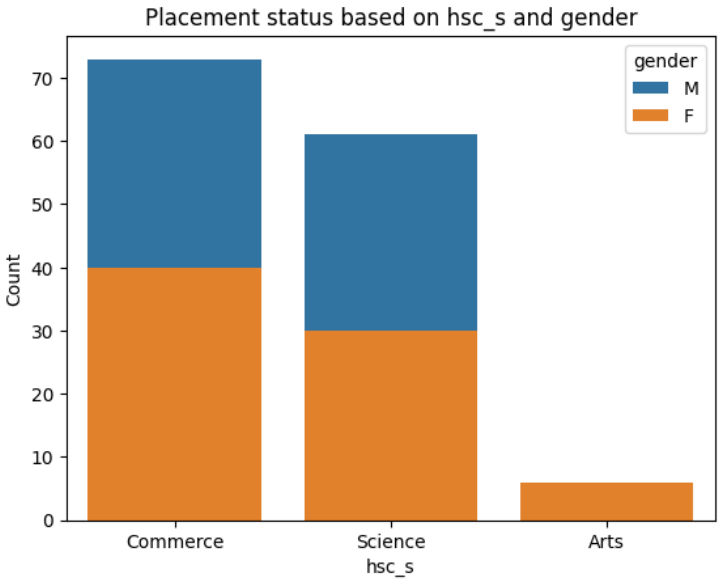
The difference between the average salary based on genders and specializations have seen a significant difference after the data cleaning with an estimate of 50000 between male and female and between Mkt&HR and Mkt&Fin.

Figure 23:Placement based on specialization



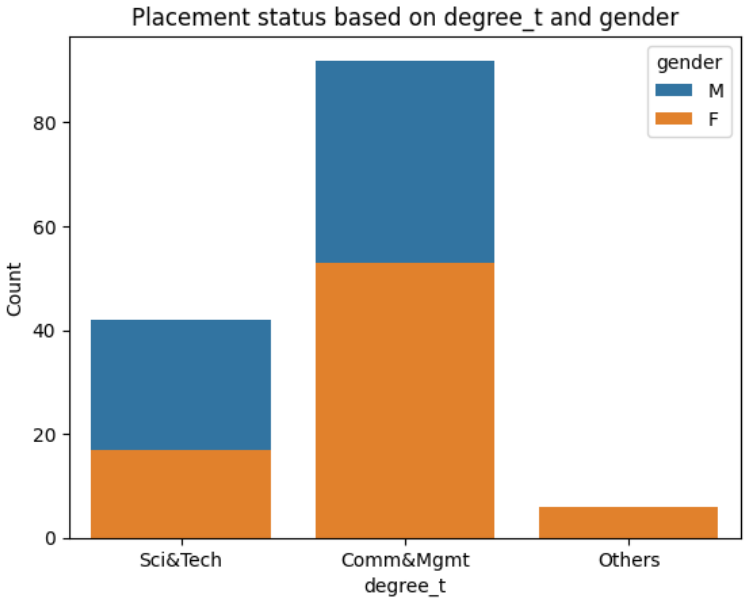
The plot shows that the students with specialization in Marketing and HR have a comparatively lower placement rate compared to those in Marketing and finance

Figure 24: Placement based on degree type and gender.



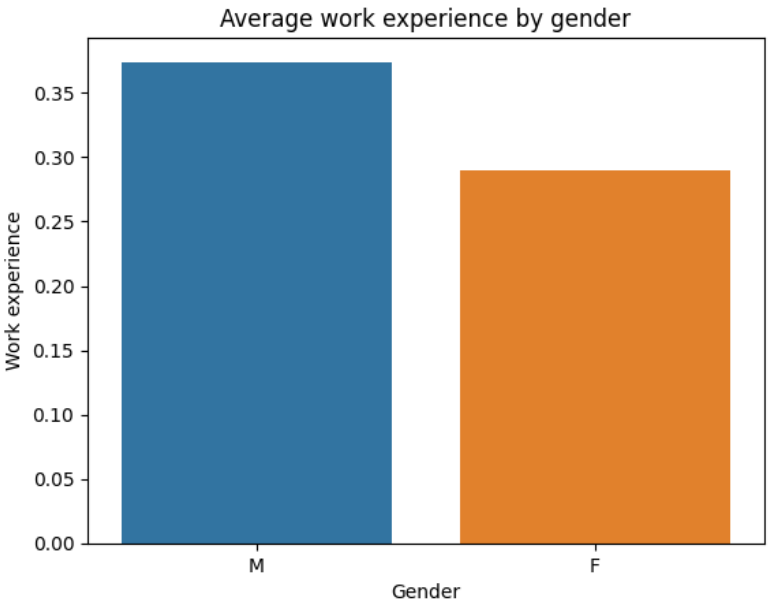
The above visualization shows the placement count of different high school specialization and their genders. It is seen that commerce has the placement ratio between the genders in commerce and science are somewhat similar. However, Arts has a 100% female population with the lowest placement count.

Figure 25:Placement status based on degree_t and gender



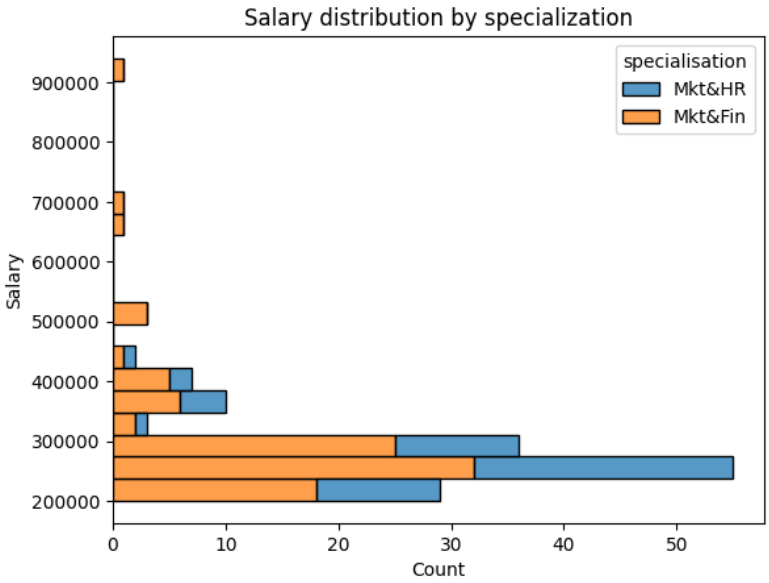
The degree type Commerce and management is seen to have the highest placement count with slightly higher females being placed. Then the science and technology stream is seen to have a higher male placement rate and the others having the lowest placement with 100% female population.

Figure 26: Average work experience by gender.



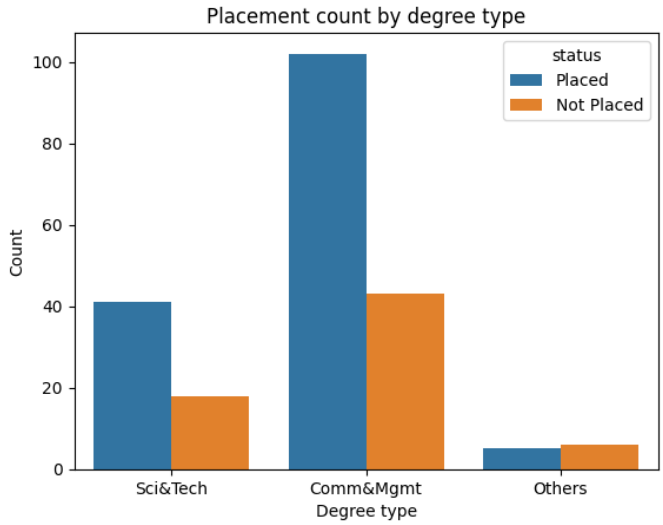
The male population is seen to have a higher average work experience right above 0.35 than the female population i.e around 0.275.

Figure 27: Salary distribution by specialization.



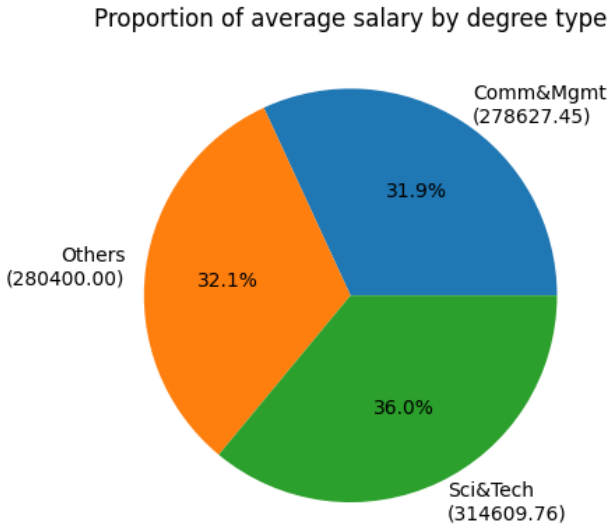
The above visualization summarizes that most students have a salary between 200000 and 300000. It is also seen that most students ini the higher salary range are from Marketing and Finance.

Figure 28: Placement count by degree type.



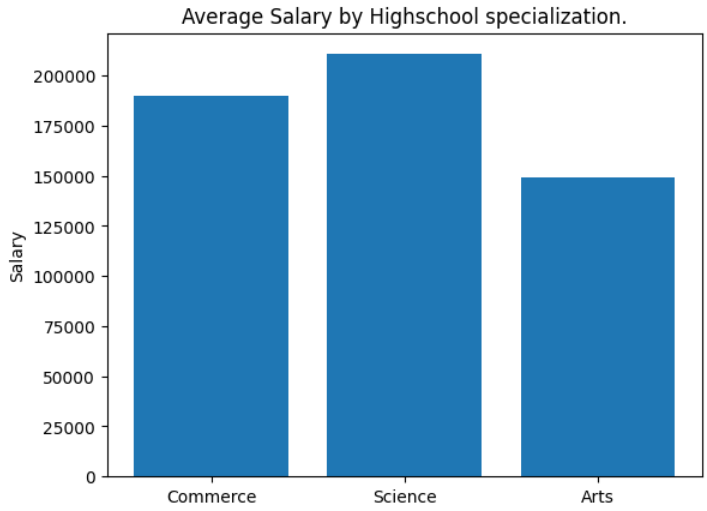
The above plot shows the placement rate of students based on their degree type.

Figure 29: Average salary by degree type



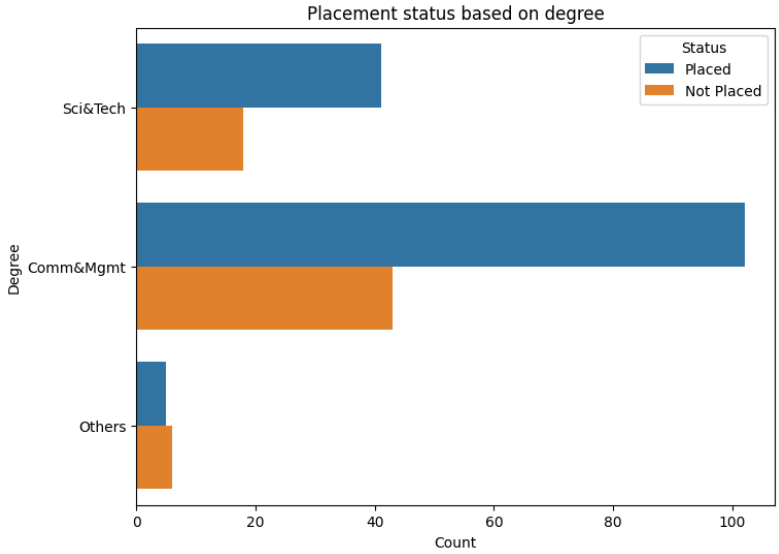
Te above pie chart shows the average salar of different degree types.

Figure 30: Average salayr by highschool specialization.



The average salary based on highschool specializations shows interesting numbers with Science students with highest average salary followed by commerce and arts respectively.

Figure 31: Placement status based on degree type.



The placement count based on degree types shows that most students in commerce and management and science and technology stream are placed while it is quite the opposite for other degree types.

Tableau Dashboard

Tableau is a data visualization tool that allows us to appropriately visualize our data to extract useful information. Using tableau, we can now further analyze our data to gain more insights that can allow us to make better business decisions.

Figure 32: Tableau dashboard.