Mandatory Assignment 1

Datasets & Data Processing

Markus Kinn

Classification:

The dataset:

The dataset I used for the classification task is: The Titanic Disaster Dataset (https://www.kaggle.com/competitions/titanic/data). The goal of this dataset is to predict whether a person, given some attributes, was going to survive the Titanic Disaster. The given attributes are: PassengerId, Pclass, Name, Age, SibSp, Parch, Ticket, Fare, Cabin and Embarked. In this notebook I process the data and look for correlations between the independent variables and the dependent variable.

Data processing:

The very first thing I did was getting a basic idea of how the dataset looks like with regards to column names, datatypes, basic statistics, null values, etc.

Then comes the cleaning. I first got rid of columns I know won't make a difference with regards to the EDA or a future ML model or had to many null values to fix. In this dataset, that meant removing PassengerId, Name, Ticket and Cabin.

After the feature selection I handled the reminding null values. This can be done in several ways. Some of the most normal ones are removing the row all together, but this is only recommended if the affected rows are only a small percent of the total dataset. In this case it would mean removing about 20% of the dataset and is therefor not a viable option. More commonly one would fill these null values with the mean or median of that specific column. I ended up with using the median since this handles outliers better.

I also chose to encode the column Sex to be represented by either 1 or 0. For most algorithms this step is needed since they don't accept strings as input.

Next, I split the Age column into 5 groups called AgeGroup. The idea behind this move is to make it easier to do EDA with regards to the Age column.

Last step of data processing was creating the column FamilyMemberCount. This column is a combination of the columns Parch and SibSp. This column later proved useful and made it possible to reduce the dataset dimensionality.

Exploratory Data Analysis:

The very first correlation I looked at was survival with regards to gender.

```
df[['Sex', 'Survived']].groupby(['Sex'], as_index=False).mean()
```

	Sex	Survived
0	0	0.742038
1	1	0.188908

Here we can clearly see that gender plays a massive role in one's chances of survival. 74% of all females survived, but only 18% of males. This is due to women and children being prioritized to get on the lifeboats, etc.

Next, I looked at the chance of survival with regards to ticket class, 1 being the best, and 3 being the worst.

```
df[['Pclass', 'Survived']].groupby(['Pclass'], as_index=False).mean()
```

	Pclass	Survived
0	1	0.629630
1	2	0.472826
2	3	0.242363

Here we can see a clear correlation between survival and ticket class. The higher your class, the more likely you were of surviving. This correlation most likely comes from two variables:

1. Higher classes were situated closes to the main deck and therefor had easier access to the few lifeboats. On the contrary, the lower class was most often situated at lower deck and therefor had a smaller chance of reaching the lifeboats.

2. Higher classes were also prioritized when it came to which was getting a lifeboat or not.

Then a looked for a correlation between FamilyMembersCount and survival.

```
df[['FamilyMembersCount', 'Survived']].groupby(['FamilyMembersCount'], as_index=False).mean().sort_values
 (by='Survived', ascending=False)
```

	FamilyMembersCount	Survived
3	4	0.724138
2	3	0.578431
1	2	0.552795
6	7	0.333333
0	1	0.303538
4	5	0.200000
5	6	0.136364
7	8	0.000000
8	11	0.000000

Here we can see that people traveling with a family size of 4, had the greatest chance of survival.

- This could be a result of richer families having fewer children

We can also see that families with a family size greater than 7, had a 0% survival rate.

- A low number of families were larger than 7 and by random they all died. Ex.: Only two families being bigger than 7 and by random they all died
- Bigger families = lower class = lower chances of survival
- Could also be due to collective suicide. Their mentality could have been; since all of us most likely won't survive, let's end it together.

With a family size of 1 (traveling alone), you also had a small chance of survival. I'm guessing this is due to most males traveling alone, and therefor this group has a low chance of survival.

```
df[['FamilyMembersCount', 'Sex']].groupby(['FamilyMembersCount'], as_index=False).mean().sort_values
 (by='FamilyMembersCount', ascending=True)
```

	FamilyMembersCount	Sex
0	1	0.765363
1	2	0.459627
2	3	0.519608
3	4	0.344828
4	5	0.200000
5	6	0.636364
6	7	0.333333
7	8	0.666667
8	11	0.571429

This tables shows that almost 77% of everyone traveling alone was male.

I then looked at the correlation between where a person embarked from and chances of survival.

```
df[['Embarked', 'Survived']].groupby(['Embarked'], as_index=False).mean()
```

	Embarked	Survived
0	C	0.553571
1	Q	0.389610
2	S	0.339009

With a quick google search we know that C = Charbourg, Q = Queenstown and S = Southampton. Q and S has basically the same survival rate, but for some reason C has a way higher chance of survival. I can only think of two reasons for this.

1. A higher percent of people from S knew how to swim.
2. A generally richer population.

Another interesting correlation is between AgeGroup and survival.

```
df[['AgeGroup', 'Survived']].groupby(['AgeGroup'], as_index=False).mean()
```

	AgeGroup	Survived
0	0	0.550000
1	1	0.344168
2	2	0.404255
3	3	0.434783
4	4	0.090909

People in the age group 0 had the best chances of survival, which is obvious why, but for some reason age group 1 has a worse chance of surviving than age group 2 and 3. I couldn't think of a reason to why this strange switch happened.

At last, I looked at a correlation heatmap of the dataset. This heatmap showed a strong correlation between survival and: sex, age and ticket class, with a lesser correlation with fare and close to zero correlation with family size.

Regression:

The dataset:

The dataset I used for the regression task is: Medical Cost Dataset. (https://www.kaggle.com/datasets/mirichoi0218/insurance). The goal of this dataset is to predict how much people would most likely spend on treatment at a hospital based on the independent variables: age, sex, bmi, children, smoker and region. In this notebook I process the data and look for correlations between the independent variables and the dependent variable.

Data processing:

As with the classification dataset, the first thing I did was to get to know the dataset, this includes looking for the same things as the previous data.

This dataset contained zero null values, which means I skipped over this part. Unlike the previous dataset, there were no columns which could be removed immediately before doing EDA.

I then had to encode several variables to make them easier to work with and make them ready for a future ML algorithm. This includes sex, smoker and region.

The last part of data processing was used to create two new columns: AgeGroup and BMIGroup. This makes it easier to do EDA on age and bmi columns, while only removing a little bit of details from these columns.

Exploratory Data Analysis:

The first correlation I looked at was between the columns charges and smoker.

```
df[['smoker', 'charges']].groupby(['smoker'], as_index=False).mean()
```

	smoker	charges
0	0	8434.268298
1	1	32050.231832

As one would guess, people which smokes pays on average about 400% that of those who doesn't. Before doing any more EDA, I can almost certainly say that smoking has the greatest correlation with charges

I then looked at the correlation between BMI group and charges.

```
df[['BMIGroup', 'charges']].groupby(['BMIGroup'], as_index=False).mean().sort_values(by='charges',
 ascending=False)
```

	BMIGroup	charges
4	4	17289.421583
2	2	15790.802305
3	3	15258.426910
1	1	11554.830480
0	0	9503.486692

We can see a pretty straight forward trend between BMI group and charges, but it doesn't seem like it makes a difference if you are in the 2nd or 3rd group. Group 1 has the lowest average charge and Group 4 has the highest charge.

Next, I looked at the correlation between AgeGroup and charges.

```
df[['AgeGroup', 'charges']].groupby(['AgeGroup'], as_index=False).mean().sort_values(by='charges',
 ascending=False)
```

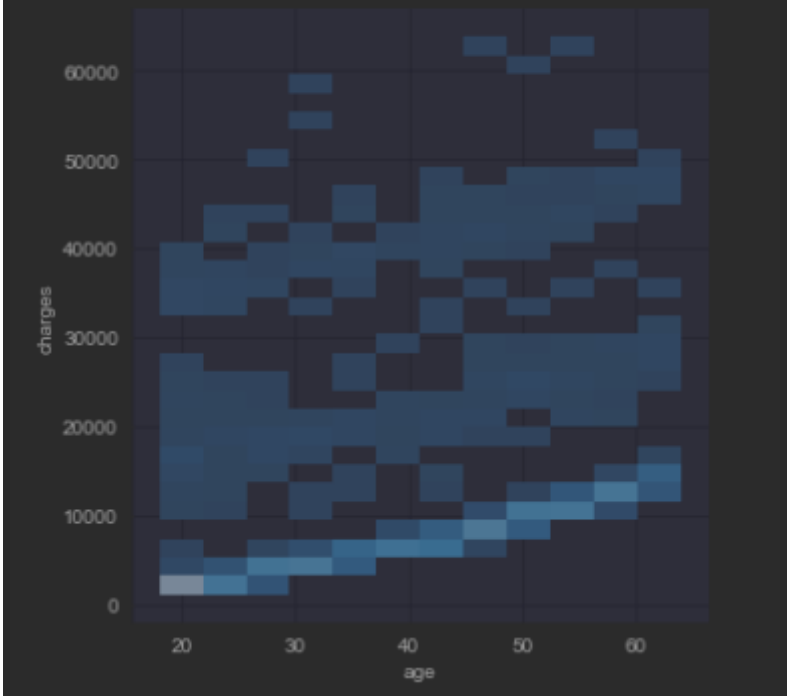
	AgeGroup	charges
4	4	18513.276227
3	3	15968.998082
2	2	13628.318836
1	1	10991.125921
0	0	9098.192248

Here there is a clear correlation between these variables. The higher the age group, the higher the average charges. By the looks of it, age plays a bigger role in the average charge

compared to bmi. Most likely this comes as a result of older people having more serious medical problems, and therefor needing more expensive treatment.

```
sns.displot(df, y=df['charges'], x=df['age'])
```

```
<seaborn.axisgrid.FacetGrid at 0x20fe475ad90>
```



This graph shows that the older one gets, the higher the lowest cost gets.

There is also a correlation between region and charges.

```
df[['region', 'charges']].groupby(['region'], as_index=False).mean().sort_values(by='charges', ascending=False)
```

	region	charges
2	2	14735.411438
0	0	13406.384516
1	1	12417.575374
3	3	12346.937377

Region 2 takes a good lead over especially region 1 and 3. Just by looking at this table it is not possible to explain where this correlation comes from, but from earlier analysis we know that bmi, smoking and age plays a major role in charges. I will therefor take a closer look at these variables in correlation with region.

First, I look at region and bmi group.

```
df[['region', 'BMIGroup']].groupby(['region'], as_index=False).mean().sort_values(by='BMIGroup',
 ascending=False)
```

	region	BMIGroup
2	2	1.832418
3	3	1.449231
1	1	1.289231
0	0	1.277778

Here we can see that region's average BMI group is 1.8, around 40% higher than the rest. This is probably one of the main factors explaining the difference between the regions.

Second, I look at region and Age Group.

```
df[['region', 'AgeGroup']].groupby(['region'], as_index=False).mean().sort_values(by='AgeGroup',
 ascending=False)
```

	region	AgeGroup
3	3	1.855385
0	0	1.848765
1	1	1.830769
2	2	1.813187

The age distribuiton is the excat same and this is therefor not a factor explaining the difference.

Third, I look at region and smoker.

```
df[['region', 'smoker']].groupby(['region'], as_index=False).mean().sort_values(by='smoker', ascending=False)
```

	region	smoker
2	2	0.250000
0	0	0.206790
1	1	0.178462
3	3	0.178462

Here we can see that the regions have the same order as the first region table. This means that this is probably the main factor explain the difference between the regions.

By looking at a correlation heatmap, we can see that the four most correlated variables are: age, bmi and smoker.

Clustering:

The dataset:

The dataset I'm using for this clustering task is: Airline Passenger Satisfaction (https://www.kaggle.com/datasets/teejmahal20/airline-passenger-satisfaction). This dataset is usually used for classification, with the goal of predicting whether a customer was satisfied with its flight experience. For this task I've removed the dependent variable/ the ground truth to convert it to a clustering problem. The rest of the dataset contains scores for different aspects of the flight experience.

Data processing:

As with the other datasets, the first thing I did was to get to know the data.

In this dataset there were only one column with missing dat. This column had 310 missing values out of a total of 104 000 rows. Because of such a small number of values missing I chose to handle these by filling them with the average value for this column.

Next step was to drop unwanted columns. These columns were: satisfaction, 'Unnamed: 0', id and Arrival Delay in Minutes. I chose to drop the Arrival delay column due to passengers most likely caring of the plane arrived late, if it didn't depart late. This was also done to remove dimensionality.

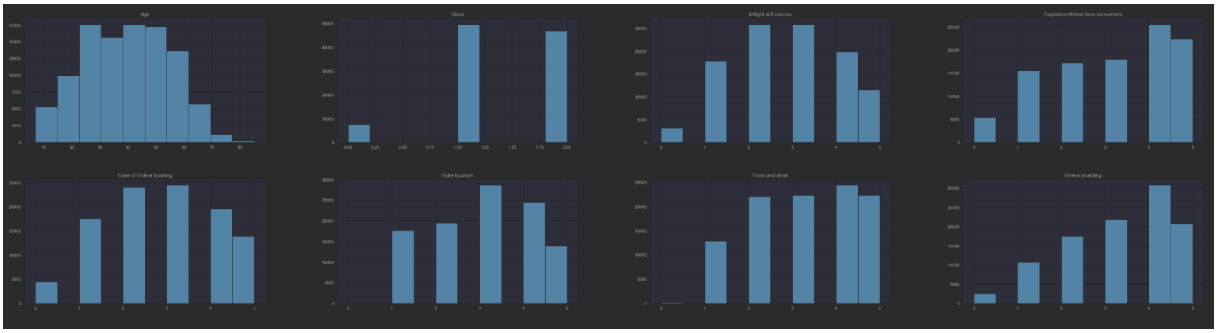
Then I did some Feature Engineering. I chose to create one new column: Late Departure. This column contains either a 0 or 1 and indicates whether a plain took off late or at time. This column might be removed depending on whether I need to reduce dimensionality.

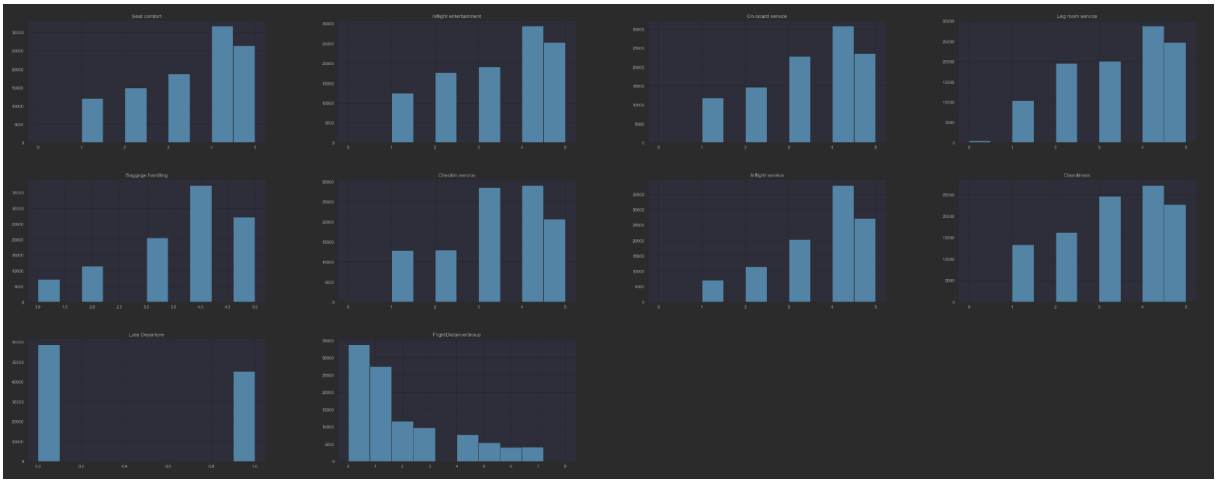
I also made Flight Distance Groups to be able to categorize the flight distance. This makes it easier to do EDA with regards to the flight distance. This column might also be removed after the EDA.

As the last step of the data processing, I encoded all the categorical variables.

Exploratory Data Analysis:

Since this is a clustering dataset and therefor it contains no ground truth, there is not much EDA to, since you can't look for correlation with a ground truth. However, I did look at the distribution of scores per column and the average scores for each column.





Here you can see that most columns are leaning right, indicating that most customers are more satisfied than unsatisfied.

I also used the function df.describe() and focused on the mean value for each column. Here I can see that most columns have an average of low 3 (out of 5). This means that most customers were more satisfied then unsatisfied.