**EXPLORATORY DATA ANALYSIS (EDA)**

Preliminary step in data analysis to:

* Summarize main characteristics of the data
* Gain better understanding of the data set
* Uncover relationships between variables
* Extract important variables

**DESCRIPTIVE STATISTICS**

* Describe basic features of data
* Giving short summaries about the sample and measure of the data

**Table

Description automatically generatedSummarize statistics using pandas describe() method: df.describe()**

The default setting of "describe" skips variables of type object. We can apply the method "describe" on the variables of type 'object' as follows: df.describe(include=['object'])

**Table

Description automatically generatedSummarize the categorical data is by using the value\_counts() method:**

For example:

Drive\_wheels\_counts = df[“drive-wheels”].value\_counts().to\_frame() (since the data after extracted is series not data frame)

Drive\_wheels\_counts.rename(columns= {‘drive-wheels’:’value\_counts’}, inplace= True)

Drive\_wheels\_counts.index.name = ‘drive-wheels’

Chart, box and whisker chart

Description automatically generatedBoxplot: show upper/lower extreme, upper/lower quartile, median, whisker, outlier

For example: sns.boxplot(x= “drive-wheels”, y= “price”, data= df)

Scatterplot:

Each observation represented as a point

Scatter plots show the relationship between 2 variables:

Chart, scatter chart

Description automatically generatedPredictor/independent variables on x-axis

Target/dependent variables on y-axis

For example:

y = df[“price”]

x= df[“engine-size”]

plt.scatter(x,y)

plt.title(“Scatterplot of engine size vs price”)

plt.xlabel(“Engine size”)

plt.ylabel(“Price”)

Table

Description automatically generated**GROUPBY IN PYTHON**

For example, let's group by the variable "drive-wheels". We see that there are 3 different categories of drive wheels. We use:

df['drive-wheels'].unique()

**Use pandas dataframe.groupby() method:**

Can be applied on categorical variables

Group data into categories

Single of multiple variables

For example:

df\_test = df[[‘drive-wheels’, ‘body-style’, ‘price’]]

df\_grp = df\_test.groupby([‘drive-wheels’, ‘body-style’], as\_index= False).mean()

**Pandas method – Pivot()**

One variable displayed along the columns and the other variable displayed along the rows

df.pivot = df\_grp.pivot(index= ‘drive-wheels’, columns= ‘body-style’)

Table

Description automatically generated

Chart

Description automatically generated

Heatmap: Plot target variable over multiple variables

plt.pcolor(df\_pivot, cmap= ‘RdBu’)

plt.colorbar()

plt.show()

**CORRELATION:** measure to what extent different variables is interdependent

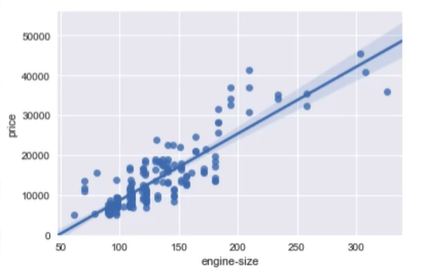
For example:

Lung cancer 🡪 smoking

Rain 🡪 umbrella

Correlation does not imply causation

**CASUSATION:** the relationship between cause and effect between 2 variables

Correlation – positive linear relationship

Correlation between 2 features (engine-sized and price)

For example:

sns.regplot(x= “engine-size”, y= “price”, data= df)

plt.ylim(0,)

Chart, scatter chart

Description automatically generatedCorrelation – negative linear relationship

Correlation between 2 features (highway-mpg and price)

sns.regplot(x= “highway-mpg”, y= “price”, data= df)

plt.ylim(0,)

Chart, scatter chart

Description automatically generatedCorrelation – Weak relationship

**CORRELATION – STATISTICS**

**Pearson correlation:** measure the strength of the correlation between 2 features

* Correlation coefficient
* P-value

Correlation coefficient:

Close to +1: large positive relationship

Close to -1: large negative relationship

Close to 0: no relationship

P-value:

P-value < 0.001: strong certainty in the result

P-value < 0.05: moderate certainty in the result

P\_value < 0.1: weak certainty in the result

P-value > 0.1: no certainty in the result

Strong correlation:

Correlation coefficient close to 1 or -1

Diagram

Description automatically generatedp-value less than 0.001

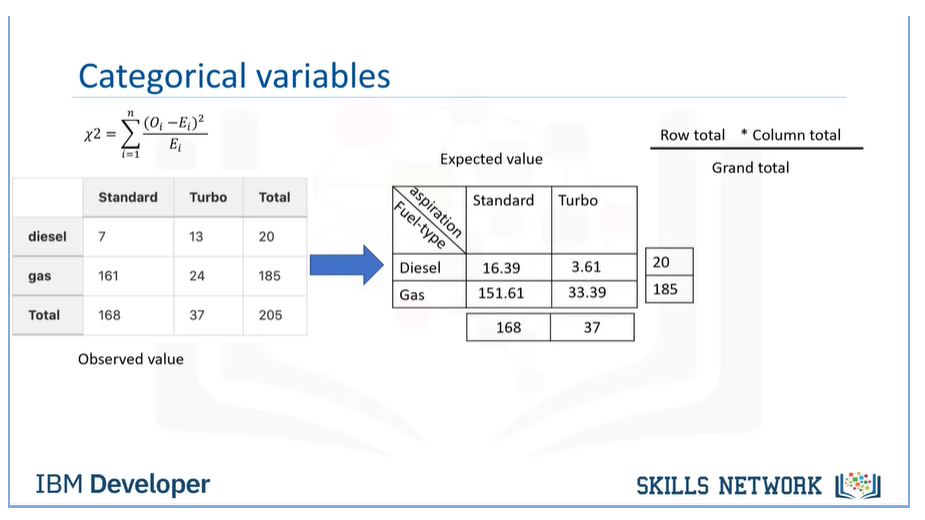
For example:

pearson\_coef, p\_value = stats.pearsonr(df[‘horsepower’], df[‘price’])

pearson correlation: 0.81

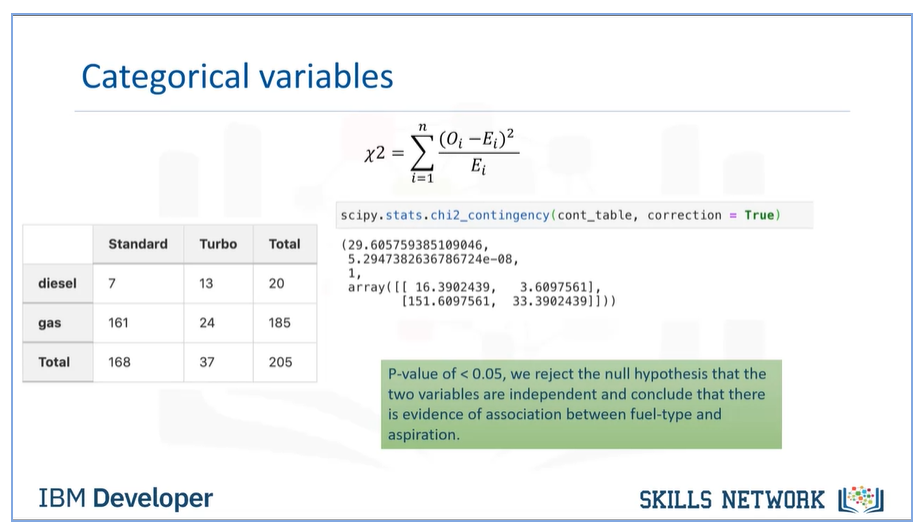
P-value: 9.35e-48

**ASSOCIATE BETWEEN 2 CATEGORICAL VARIABLES: CHI-SQUARE**

* Categorical variables
* We use the Chi-square test for association (denoted as χ2)
* The test is intended to test how likely it is that an observed distribution is due to chance
* The Chi-square tests a null hypothesis that the variables are independent
* The Chi-square does not tell you the type of relationship that exists between both variables; but only that a relationship exists

Table

Description automatically generated



ANOVA: Analysis of Variance

The Analysis of Variance (ANOVA) is a statistical method used to test whether there are significant differences between the means of two or more groups. ANOVA returns two parameters:

F-test score: ANOVA assumes the means of all groups are the same, calculates how much the actual means deviate from the assumption, and reports it as the F-test score. A larger score means there is a larger difference between the means.

P-value: P-value tells how statistically significant our calculated score value is

If our price variable is strongly correlated with the variable we are analyzing, we expect ANOVA to return a sizeable F-test score and a small p-value.

Since ANOVA analyzes the difference between different groups of the same variable, the “groupby” function will come in handy. Because the ANOVA algorithm averages the data automatically, we do not need to take the average before hand. To see if different types of 'drive-wheels' impact ‘price’, we group the data.

For example:

f\_val, p\_val = stats.f\_oneway(grouped\_test2.get\_group('fwd')['price'], grouped\_test2.get\_group('rwd')['price'], grouped\_test2.get\_group('4wd')['price'])

This is a great result with a large F-test score showing a strong correlation and a P-value of almost 0 implying almost certain statistical significance. But does this mean all three tested groups are all this highly correlated?

Let's examine them separately.

Fwd and rwd:

f\_val, p\_val = stats.f\_oneway(grouped\_test2.get\_group('fwd')['price'], grouped\_test2.get\_group('rwd')['price'])