

# Data Analysis with Python

## Cheat Sheet: Exploratory Data Analysis

| Package/Method                  | Description   | Code Example   |
|---------------------------------|---|--|
| Complete dataframe correlation  | Correlation matrix created using all the attributes of the dataset.   | <pre>1. 1 1. df.corr()</pre> <div>Copied!</div>  |
| Specific Attribute correlation  | Correlation matrix created using specific attributes of the dataset.  | <pre>1. 1 1. df[['attribute1','attribute2',...]].corr()</pre> <div>Copied!</div>   |
| Scatter Plot                    | Create a scatter plot using the data points of the dependent variable along the x-axis and the independent variable along the y-axis.   | <pre>1. 1 2. 2 1. from matplotlib import pyplot as 2. plt plt.scatter(df[['attribute_1']],df[['attribute_2']])</pre> <div>Copied!</div>  |
| Regression Plot                 | Uses the dependent and independent variables in a Pandas data frame to create a scatter plot with a generated linear regression line for the data.  | <pre>1. 1 2. 2 1. import seaborn as sns 2. sns.regplot(x='attribute_1',y='attribute_2', data=df)</pre> <div>Copied!</div>  |
| Box plot                        | Create a box-and-whisker plot that uses the pandas dataframe, the dependent, and the independent variables.   | <pre>1. 1 2. 2 1. import seaborn as sns 2. sns.boxplot(x='attribute_1',y='attribute_2', data=df)</pre> <div>Copied!</div>  |
| Grouping by attributes          | Create a group of different attributes of a dataset to create a subset of the data.   | <pre>1. 1 1. df_group = df[['attribute_1','attribute_2',...]]</pre> <div>Copied!</div>   |
| GroupBy statements              | a. Group the data by different categories of an attribute, displaying the average value of numerical attributes with the same category.<br>b. Group the data by different categories of multiple attributes, displaying the average value of numerical attributes with the same category. | <pre>1. 1 2. 2 3. 3 4. 4 5. 5 6. 6 1. a. 2. df_group = 3. df_group.groupby(['attribute_1'],as_index=False).mean() 4. b. 5. df_group = df_group.groupby(['attribute_1', 6. 'attribute_2'],as_index=False).mean()</pre> <div>Copied!</div> |
| Pivot Tables                    | Create Pivot tables for better representation of data based on parameters   | <pre>1. 1 2. 2 1. grouped_pivot = 2. df_group.pivot(index='attribute_1',columns='attribute_2')</pre> <div>Copied!</div>  |
| Pseudocolor plot                | Create a heatmap image using a PsuedoColor plot (or pcolor) using the pivot table as data.  | <pre>1. 1 2. 2 1. from matplotlib import pyplot as plt 2. plt.pcolor(grouped_pivot, cmap='RdBu')</pre> <div>Copied!</div>  |
| Pearson Coefficient and p-value | Calculate the Pearson Coefficient and p-value of a pair of attributes   | <pre>1. 1 2. 2 3. 3 1. From scipy import stats 2. pearson_coef,p_value=stats.pearsonr(df['attribute_1'], 3. df['attribute_2'])</pre> <div>Copied!</div>  |



# Skills Network