# Correlation concepts in Attrition analysis.

1. Correlation between **Attrition** and **Age**

**Hypothesis:**

H0 -> There is no significant correlation between Attrition and Age

H1 -> These is significant correlation between Attrition and Age

*import pandas as pd*

*import matplotlib.pyplot as plt*

*from sklearn.preprocessing import LabelEncoder*

*from scipy.stats import pearsonr*

*# loading dataset*

*dataset = pd.read\_csv("general\_data.csv")*

*# data treatment*

*dataset.isnull()*

*dataset.dropna()*

*# converting categorical string data into integers*

*number = LabelEncoder()*

*dataset["Attrition"] = number.fit\_transform(dataset["Attrition"].astype("str"))*

*# determining correlation*

*stats,p = pearsonr(dataset.Attrition, dataset.Age)*

*print(stats,p)*

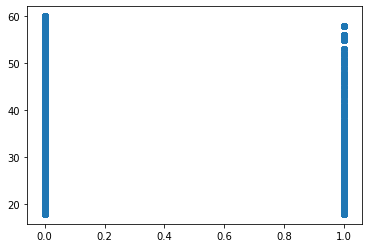
*-0.15920500686577507 1.996801615893171e-26*

*Analysis:*

|  |  |  |
| --- | --- | --- |
| *Parameter* | *Value* | *Inference* |
| *Correlation coefficient, r* | *-0.15920500686577507* | *Negative correlation* |
| *p value* | *1.996801615893171e-26* | *Less than 0.05, thus NULL HYPOTHESIS IS REJECTED* |

*# graphically displaying correlation*

*plt.scatter(dataset.Attrition, dataset.Age)*



1. Correlation between **Attrition** and **DistanceFromHome**

**Hypothesis:**

H0 -> There is no significant correlation between Attrition and DistanceFromHome

H1 -> These is significant correlation between Attrition and DistanceFromHome

*import pandas as pd*

*import matplotlib.pyplot as plt*

*from sklearn.preprocessing import LabelEncoder*

*from scipy.stats import pearsonr*

*# loading dataset*

*dataset = pd.read\_csv("general\_data.csv")*

*# data treatment*

*dataset.isnull()*

*dataset.dropna()*

*# converting categorical string data into integers*

*number = LabelEncoder()*

*dataset["Attrition"] = number.fit\_transform(dataset["Attrition"].astype("str"))*

*# determining correlation*

*stats,p = pearsonr(dataset.Attrition, dataset.DistanceFromHome)*

*print(stats,p)*

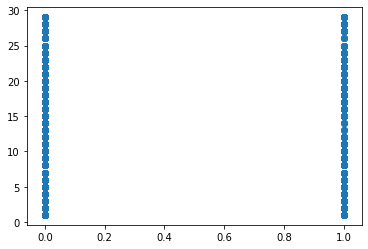
*-0.009730141010179438 0.5182860428049617*

*Analysis:*

|  |  |  |
| --- | --- | --- |
| *Parameter* | *Value* | *Inference* |
| *Correlation coefficient, r* | *-0.009730141010179438* | *Negative correlation* |
| *p value* | *0.5182860428049617* | *Greater than 0.05, thus NULL HYPOTHESIS IS ACCEPTED* |

*# graphically displaying correlation*

*plt.scatter(dataset.Attrition, dataset.DistanceFromHome)*



1. Correlation between **Attrition** and **Education**

**Hypothesis:**

H0 -> There is no significant correlation between Attrition and Education

H1 -> These is significant correlation between Attrition and Education

*import pandas as pd*

*import matplotlib.pyplot as plt*

*from sklearn.preprocessing import LabelEncoder*

*from scipy.stats import pearsonr*

*# loading dataset*

*dataset = pd.read\_csv("general\_data.csv")*

*# data treatment*

*dataset.isnull()*

*dataset.dropna()*

*# converting categorical string data into integers*

*number = LabelEncoder()*

*dataset["Attrition"] = number.fit\_transform(dataset["Attrition"].astype("str"))*

*# determining correlation*

*stats,p = pearsonr(dataset.Attrition, dataset.* *Education)*

*print(stats,p)*

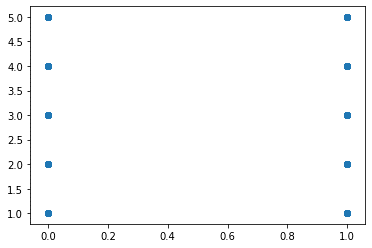
*-0.015111167710968734 0.3157293177118575*

*Analysis:*

|  |  |  |
| --- | --- | --- |
| *Parameter* | *Value* | *Inference* |
| *Correlation coefficient, r* | *-0.015111167710968734* | *Negative correlation* |
| *p value* | *0.3157293177118575* | *Greater than 0.05, thus NULL HYPOTHESIS IS ACCEPTED* |

*# graphically displaying correlation*

*plt.scatter(dataset.Attrition, dataset.Education)*



1. Correlation between **Attrition** and **JobLevel**

**Hypothesis:**

H0 -> There is no significant correlation between Attrition and JobLevel

H1 -> These is significant correlation between Attrition and JobLevel

*import pandas as pd*

*import matplotlib.pyplot as plt*

*from sklearn.preprocessing import LabelEncoder*

*from scipy.stats import pearsonr*

*# loading dataset*

*dataset = pd.read\_csv("general\_data.csv")*

*# data treatment*

*dataset.isnull()*

*dataset.dropna()*

*# converting categorical string data into integers*

*number = LabelEncoder()*

*dataset["Attrition"] = number.fit\_transform(dataset["Attrition"].astype("str"))*

*# determining correlation*

*stats,p = pearsonr(dataset.Attrition, dataset.* *JobLevel)*

*print(stats,p)*

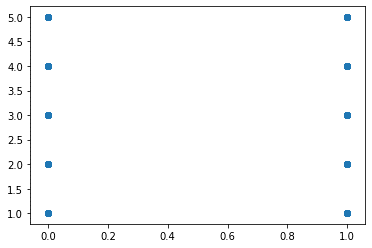
*-0.010289713287495119 0.49451717271817114*

*Analysis:*

|  |  |  |
| --- | --- | --- |
| *Parameter* | *Value* | *Inference* |
| *Correlation coefficient, r* | *-0.010289713287495119* | *Negative correlation* |
| *p value* | *0.49451717271817114* | *Greater than 0.05, thus NULL HYPOTHESIS IS ACCEPTED* |

*# graphically displaying correlation*

*plt.scatter(dataset.Attrition, dataset.JobLevel)*



1. Correlation between **Attrition** and **NumCompaniesWorked**

**Hypothesis:**

H0 -> There is no significant correlation between Attrition and NumCompaniesWorked

H1 -> These is significant correlation between Attrition and NumCompaniesWorked

*import pandas as pd*

*import matplotlib.pyplot as plt*

*from sklearn.preprocessing import LabelEncoder*

*from scipy.stats import pearsonr*

*# loading dataset*

*dataset = pd.read\_csv("general\_data.csv")*

*# data treatment*

*dataset.isnull()*

*dataset.dropna()*

*# converting categorical string data into integers*

*number = LabelEncoder()*

*dataset["Attrition"] = number.fit\_transform(dataset["Attrition"].astype("str"))*

*# determining correlation*

*stats,p = pearsonr(dataset.Attrition, dataset.* *NumCompaniesWorked)*

*print(stats,p)*

*0.04283056724471901 0.004572057121625154*

*Analysis:*

|  |  |  |
| --- | --- | --- |
| *Parameter* | *Value* | *Inference* |
| *Correlation coefficient, r* | *0.04283056724471901* | *Positive correlation* |
| *p value* | *0.004572057121625154* | *Lesser than 0.05, thus NULL HYPOTHESIS IS REJECTED* |

*# graphically displaying correlation*

*plt.scatter(dataset.Attrition, dataset.NumCompaniesWorked)*

