Exploratory data analysis

**Decimals**

import numpy as np

import pandas as pd

pd.set\_option("display.precision", 2)

**Read dataset**

df = pd.read\_csv("train.csv")

df.head()

**Printing dataset dimensions**

Print(df.shape)

**Printing name of colums**

Print(df.columns)

**Printing information about dataframe**

Print(df.info())

**Change column type with int64**

from matplotlib.pylab import int64

df["Fare"] = df["Fare"].astype(int64)

df.describe()

**Non-numerical features**

df.describe(include=["object", "float64"])

**Categorical and boolean features using value\_counts**

df["Sex"].value\_counts()

**Calculate fractions**

df["Sex"].value\_counts(normalize=True)

**Sort by Name**

df.sort\_values(by=”Name”, ascending=False).head()

**Sort multiple columns**

Df.sort\_values(by=[“Churn”, “Total day charge”], ascending=[True, False]).head()

**Indexing data**

df["Age"].mean()

**Average daytime**

df[df["Age"] == 1]["Age"].mean()

**Maximum length**

df[(df["Survived"] == 0) & (df["Sex"] == "male")]["Age"].max()

**Loc method is used for indexing by name**

df.loc[0:5, "Survived": "Sex"]

**Iloc method is used for indexing by number**

df.iloc[2:8, 3:5]

**Show last line of the dataframe**

df[-1:]

**Apply colums, rows and functions**

df.apply(np.count\_nonzero)

**Apply method is used to apply function in each row**

df[df["Ticket"].apply(lambda ticket: ticket[0] == "S")].head()

**Map method is used for replacing values in column**

d = {0: "False", 1: "True"}

df["Survived"] = df["Survived"].map(d)

df.head()

**Same just with replace method**

d = {"male": "M", "female": "F"}

df = df.replace({"Sex": d})

df.head()

**Building contingency table using crosstab method**

pd.crosstab(df["Survived"], df["Sex"])

**Defining pivot\_table**

df.pivot\_table(

    ["Survived", "Parch", "SibSp"],

    ["Age"],

    aggfunc="sum",

)

**Dataframe transformation**

total = (

    df["Name"]

    + df["Cabin"]

)

df.insert(loc=len(df.columns), column="Sum", value=total)

df.head()

Same in other way

df["Total"] = (

    df["Name"]

    + df["Cabin"]

)

df.head()

**Drop method is used to delete columns**

Df.drop([“Total charge”, “Total calls”], axis=1, inplace=True) --- koi koloni da ne gi prikazuva

Df.drop([1,2]).head() --- koi redovi da ne gi prikazuva

crossTab

pd.crosstab(df["Survived"], df["Sex"], margins=True)

**Show First Image**

import matplotlib.pyplot as plt

import seaborn as sns

sns.countplot(x="Sex", hue="Survived", data=df)

**Show Second Image**

pd.crosstab(df["Survived"], df["Sex"], margins=True)

sns.countplot(x="Sex", hue="Survived", data=df)

**Show First Table**

pd.crosstab(df["Sex"], df["Survived"], margins=True)

**Show Third Page**

sns.countplot(x="Sex", hue="Survived", data=df)

**Show Second Table**

pd.crosstab(df["Sex"], df["Survived"])