Covaliu Stefan Rebis 313 CB Partea 1

Proiect Titanic Partea 1

Link github:

https://github.com/CovaliuStefan/titanic-analysis.git

Cerinta 1

Numarul de coloane: 12

Tipurile de date pentru fiecare colaona:

PassengerId int64

Survived int64

Pclass int64

Name object

Sex object

Age float64

SibSp int64

Parch int64

Ticket object

Fare float64

Cabin object

Embarked object

dtype: object

Numarul de valori lipsa pentru fiecare coloana:

PassengerId 0

Survived 0

Pclass 0

Name 0

Sex 0

Age 177

SibSp 0

Parch 0

Ticket 0

Fare 0

Cabin 687

Embarked 2

dtype: int64

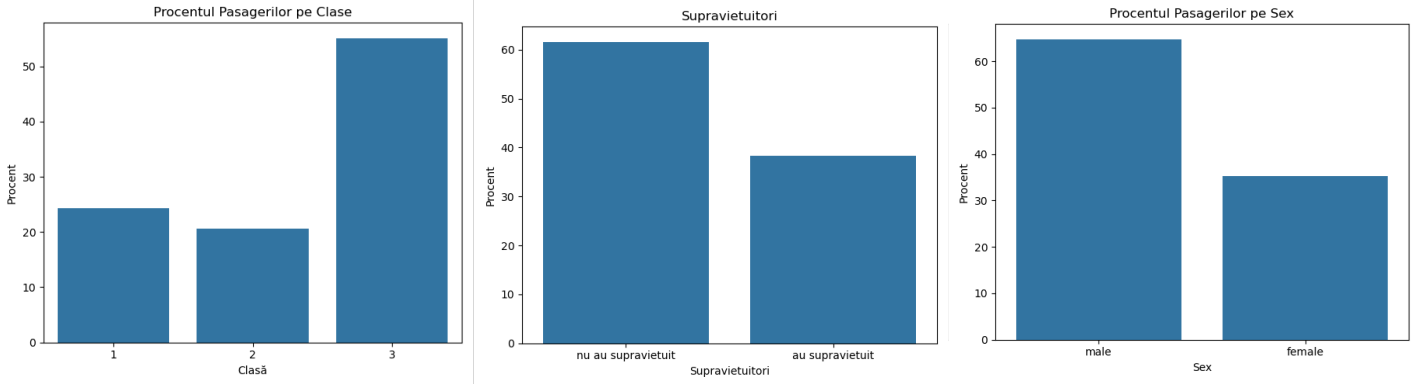
Numarul de linii: 891

Numarul de linii duplicate: 0

Numarul de coloane este egal cu dimensiunea len(df.columns). Extrag tipurile de date folosind df.dtypes. Determinam numarul de valori lipsa pentru fiecare coloanal folosind df.isnull().sum(). Aplic filtrul isnull() pentru a selecta valorile neintroduse si le numar folosind functia sum. Numarul de linii este len(df).

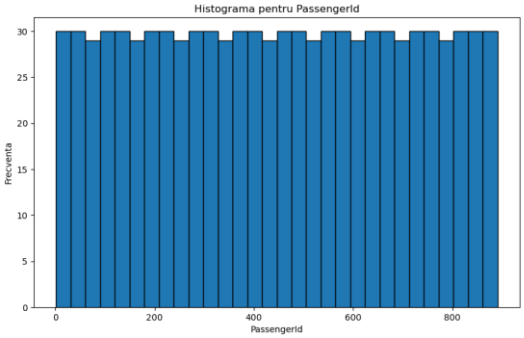
Verific daca exista linii duplicate cu df.duplicated().sum(). Procedeul este similar cu numararea valorile lipsa. La final afisez toate aceste date cu functia print.

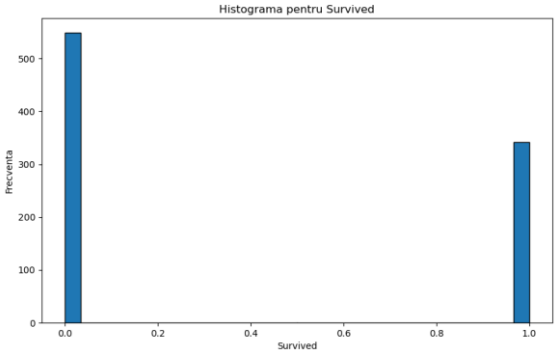
Cerinta 2

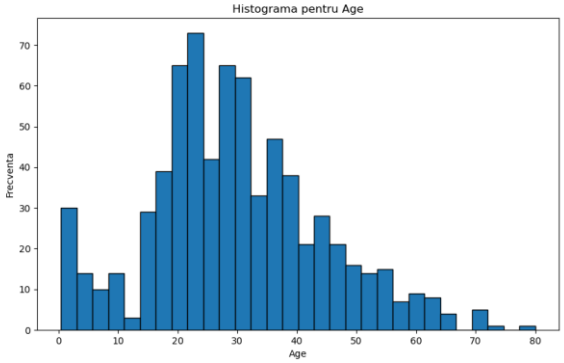


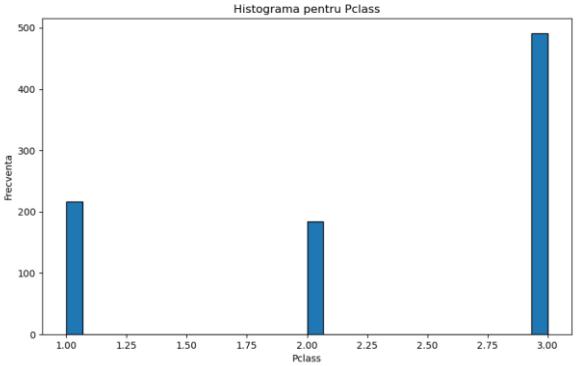
Determine procentul persoanelor care au supravietuit si care nu au supravietuit, procentul pasagerilor pentru fiecare tip de clasa si procentul barbatilor si femeilor. Folosesc functia value_counts. Voi creea o singura fereastra care continue cele trei grafice.

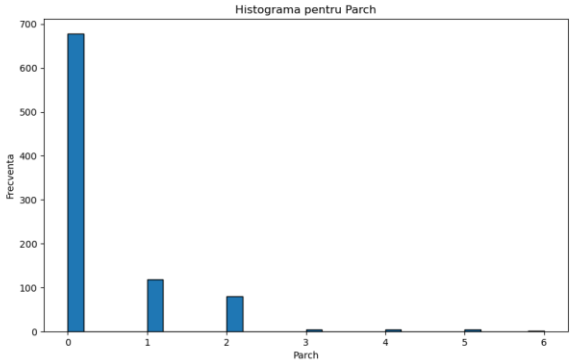
Cerinta 3

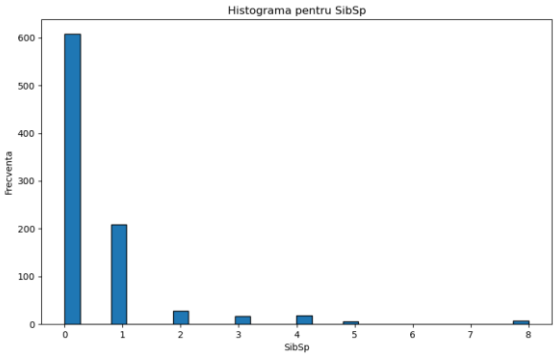

Histograma pentru PassengerId
Frecventa
PassengerId

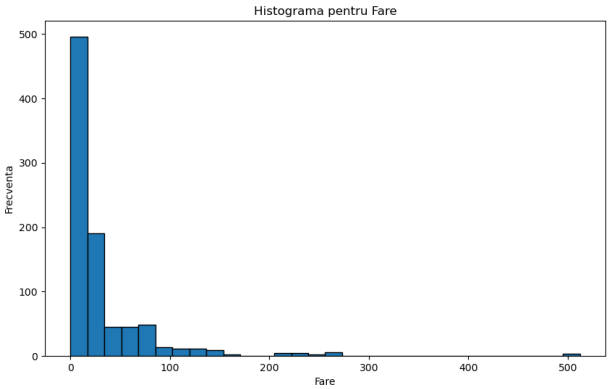

Histograma pentru Survived
Frecventa
Survived


Histograma pentru Age
Frecventa
Age


Histograma pentru Pclass
Frecventa
Pclass


Histograma pentru Parch
Frecventa
Parch


Histograma pentru SibSp
Frecventa
SibSp


Histograma pentru Fare
Frecventa
Fare

Se poate observa ca sunt mai multe personae care nu au supravietuit decat cele care au supravietuit.

In general, au fost mai multi tineri si adulti cu varsta intre 20 si 30 de ani. Majoritatea persoanelor au avut clasa a 3-a.

Cele mai multe personae au avut maxim un frate/sora sot/sotie la bord. Siblings/Spouses Aboard

Cele mai multe bilete vandute au costat sub 100 de lire.

Pentru a realiza graficele am extras coloanele care contin valori numerice

df.select_dtypes(include=['int64', 'float64']).columns

Apoi am realizat cate un grafic pentru fiecare coloana.

Cerinta 4

Coloanele cu valori lipsa și numarul acestora:

Age 177

Cabin 687

Embarked 2

dtype: int64

Proportia valorilor lipsa pentru fiecare coloana:

Age 19.865320

Cabin 77.104377

Embarked 0.224467

dtype: float64

Procentul valorilor lipsa pentru supravietuitori:

Survived	Age	Cabin	Embarked
0	22.768670	87.613843	0.000000
1	15.204678	60.233918	0.584795

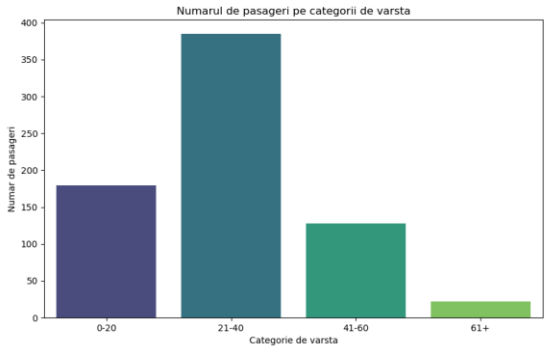
Pentru inceput pastrez doar coloanele care contin valori lipsa. Nr de valori lipsa > 0.

Proporitia valorilor lipsa = (coloane_valori_lipse / len(df)) * 100

Fac acelasi lucru doar pentru supravieturitori.

Pentru o afisare mai usor de inteles am eliminat coloanele care nu au valori lipsa. procent_lipsa_per_clasa.loc[:, procent_lipsa_per_clasa.any()]

Cerinta 5

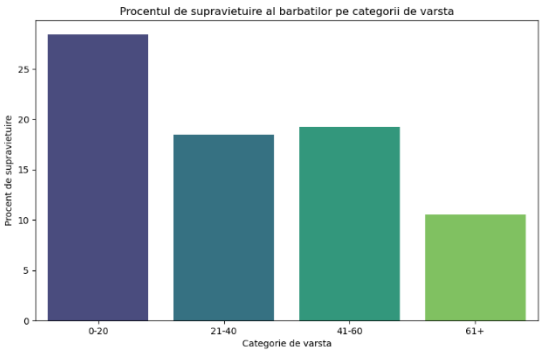


Majoritatea persoanelor imbarcare au intre 21 si 40 de ani.

Creez variabilele bins si labels pentru intervalele cerute. Adaug fiecarei inregistrari o coloanal in plus in care retin categoria de varsta din care face parte persoana respectiva pentru a putea numara cate persoane fac parte din fiecare categorie.

Pe axa x a graficului sunt categoriile de varsta, iar pe axa y este afisat numarul de persoane din fiecare categorie

Cerinta 6

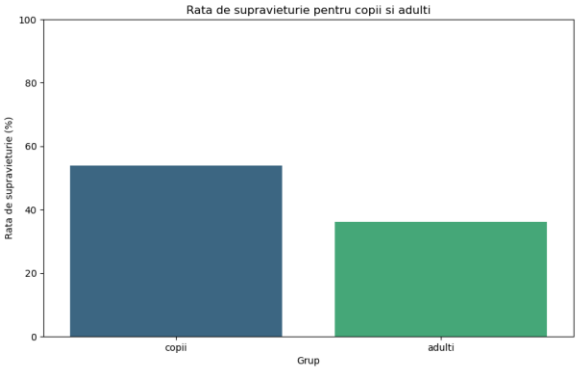


Dupa cum se observa, cei mai putini supravietuitori din numarul barbatilor erau mai in varsta de 61 de ani. Cei mai multi erau tineri de pana in 20 de ani.

Pentru aceasta cerinta doar am aplicat un filtru pentru a pastra doar barbatii.

df[df['Sex'] == 'male']

Cerinta 7



Au supravietuit mai multi copii decat adulti.

Am selectat din fisier copiii. df['IsChild'] = df['Age'] < 18

Am caluclat numarul lor aplicand filtrul si folosind functia sum. Apoi am calculate procenul lor. Am luat in considerare doar persoanele care au supravietuit.

Pentru rata supravietuirii la adulti am folosit calculul (totalPersoane - nuamrCopii) * 100.

Pentru crearea graficului am folosit un dataframe in care declar numele celor doua coloane si valorile acestora.

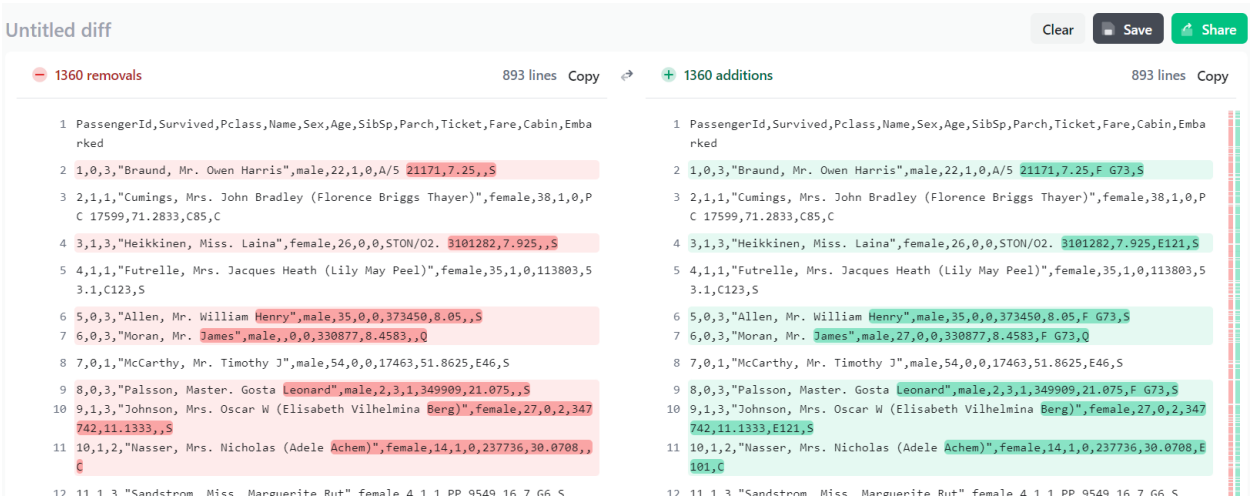
La final afisez datele si in format text.

Procentul copiilor aflati la bord: 12.68%

Rata de supravieturie pentru adulti: 36.12%

Rata de supravieturie pentru copii: 53.98%

Cerinta 8



Am adaugat valorile lipsa. Atat cele numerice cat si cele categoriale.

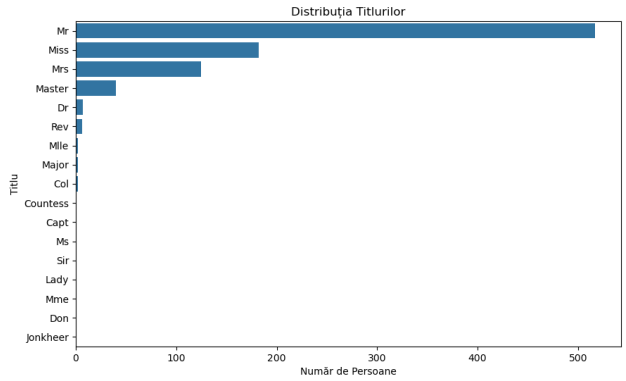
Am creeat 2 functii separate. fill_missing_numeric si fill_missing_categorical.

Pentru fiecare coloana functie corespunzatoare, odata cu filtrele pentru categoriile class si survived.

Pentru valorile numerice iterez prin fiecare grup si calculez media, apoi completez valoarea pentru fiecare inregistrare in care coloana respectiva nu este completata.

Pentru valorile categoriale iterez prin fiecare grup si gasesc cea mai frecventa valoare, apoi completez valoarea pentru fiecare intregistrare in care coloana respectiva nu este completata. In plus tin cont si de filtrele impuse in cerinta (class si survived).

Cerinta 9



Extrag titlurile din coloana Name

str.extract(r' ([A-Za-z]+)\.'

(sirurile de caractere alfabetice dinaintea punctului)

Creez o coloanal noua Title_Sex_Match in care pun true sau false, daca corespund titlurile.

Pentru a verifica daca titlul corespunde fiecarei persoane, am adaugata titlurile in cate o lista si le parcurg verificand totodata si sexul persoanei.

Creez graficul distributiei titlurilor.

Apoi verific pentru cate persoane nu corepunde titlul si afisez inregistrarea.

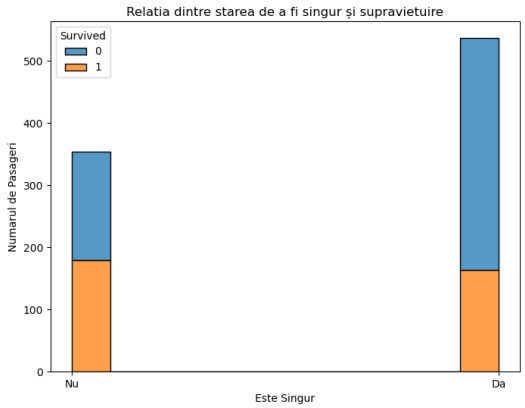
Titluri care nu corespund cu sexul persoanei:

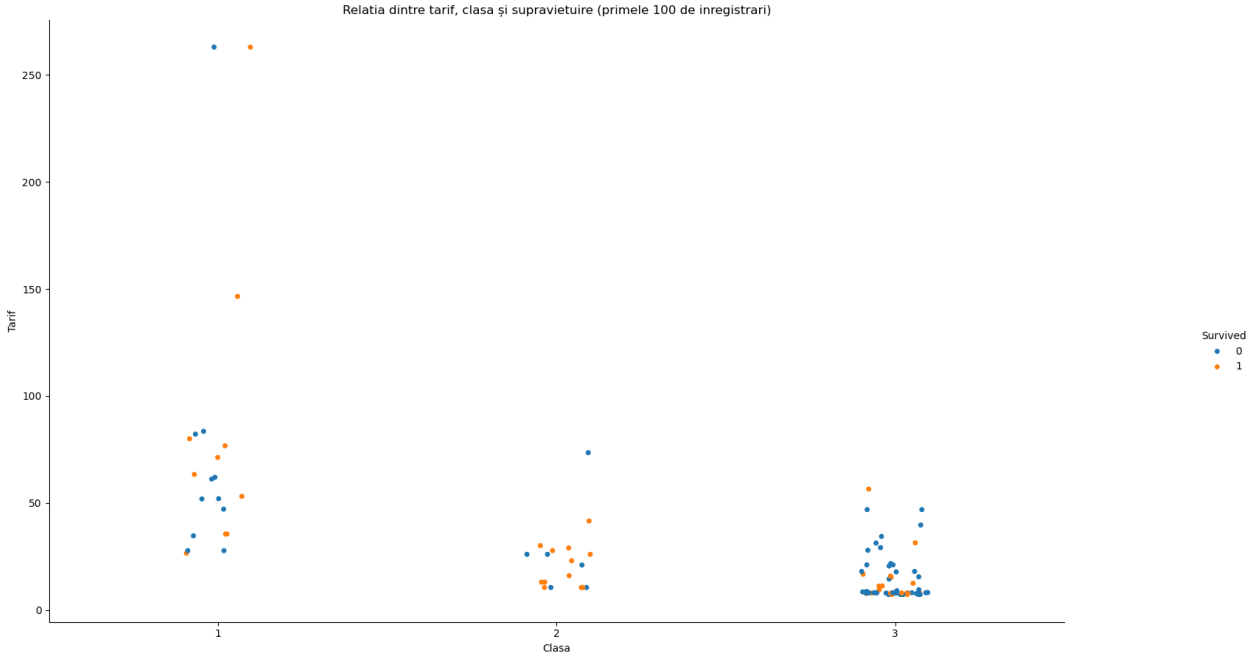
	Name	Sex	Title
796	Leader, Dr. Alice (Farnham)	female	Dr

Procentajul de barbati: 64.75869809203144 %

Am afisat si procentul de barbati pentru a verifica daca graficul este corect.

Cerinta 10





Au supravietuit mai multi pasageri care au calatori la clasa a 2-a. Cei mai multi pasageri care nu au supravietuit au calatori la clasa a 3-a.

verific daca un pasager este singur

df['IsAlone'] = (df['SibSp'] == 0) & (df['Parch'] == 0)

pentru primul grafic: x='IsAlone', hue='Survived'

pentru al doilea grafic am utilizat catplot: x='Pclass', y='Fare', hue='Survived' si kind='swarm' pentru a vedea detalii pe grafic.