Pavel Gabriel-Ioan 313CC GitHub Repo

Explicatii pe cerinte:

Cerinta 1

Citesc informatiile din `train.csv` si aplic functia `listData(df)` din `examine.py`. Aceasta construieste o lista cu toate informatiile cerute: numarul de coloane, numarul de linii, o lista cu tipurile de date, numarul de valori lipsa si daca exista duplicate.

Numar de coloane: 12

Tipurile datelor din coloana: ['int64', 'int64', 'int64', 'object', 'object', 'float64', 'int64', 'int64', 'object', 'float64', 'object', 'object']

Numar de valori lipsa pentru fiecare coloana: [0, 0, 0, 0, 0, 177, 0, 0, 0, 0, 687, 2]

Numar de linii: 891

Exista duplicate: Nu

Cerinta 2

Se folosesc functiile `prcS(df)`, `prcPclass(df)` si `prcSex(df)` pentru a determina procentul peroanelor care au supravietuit si care nu au supravietuit, procentul pasagerilor pentru fiecare tip de clasa, respectiv procentul barbatilor si al femeilor. Aceste functii se afla in `examine.py`. Functia `pies(df)` din `plots.py` realizeaza graficele pentru a reprezenta datele calculate mai devreme.

Procent persoane care au supravietuit: 38.38%

Procent persoane care nu au supravietuit: 61.62%

Procentul pasagerilor pentru fiecare clasa:

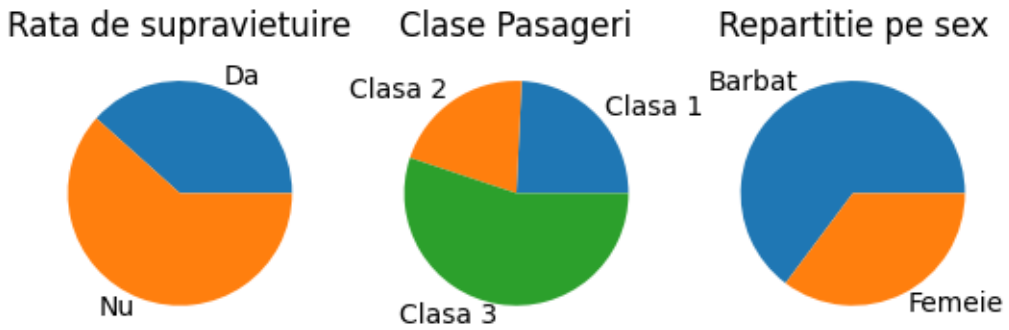
C1: 24.24%

C2: 20.65%

C3: 55.11%

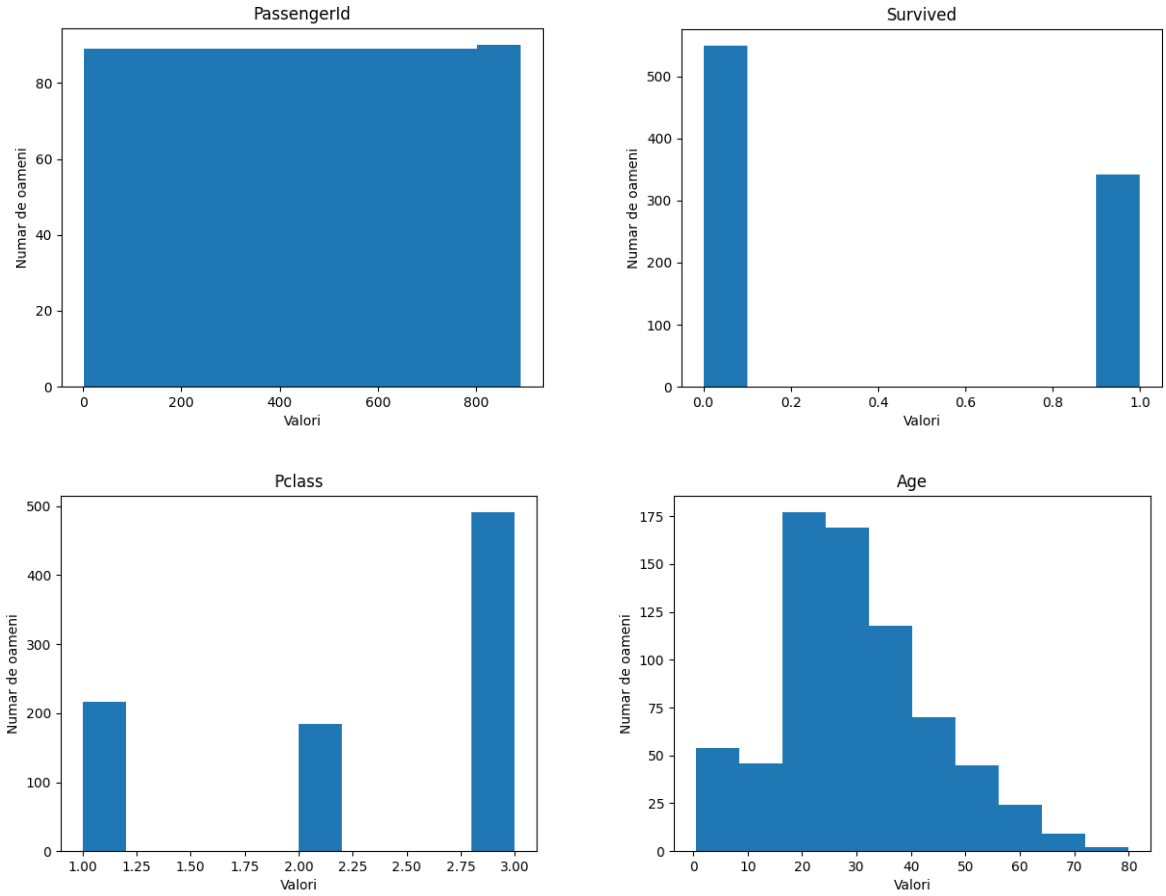
Procent barbati: 64.76%

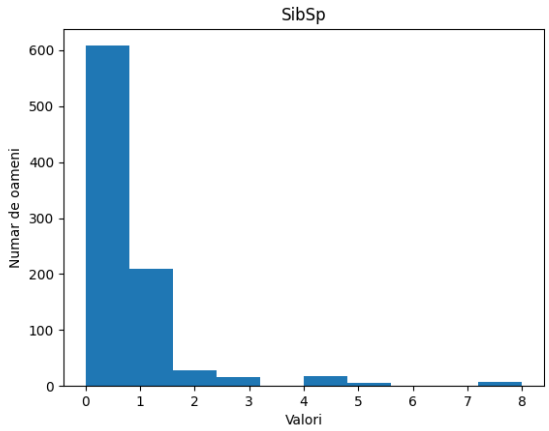
Procent femei: 35.24%

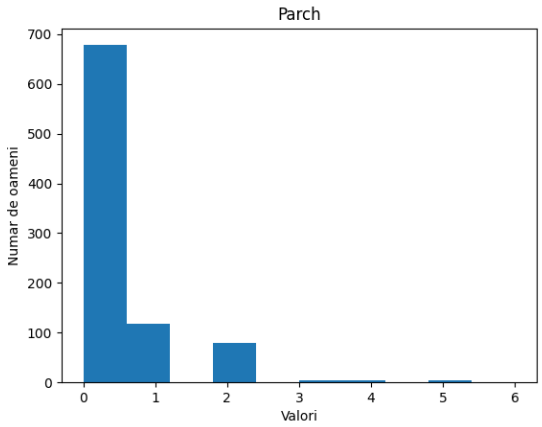


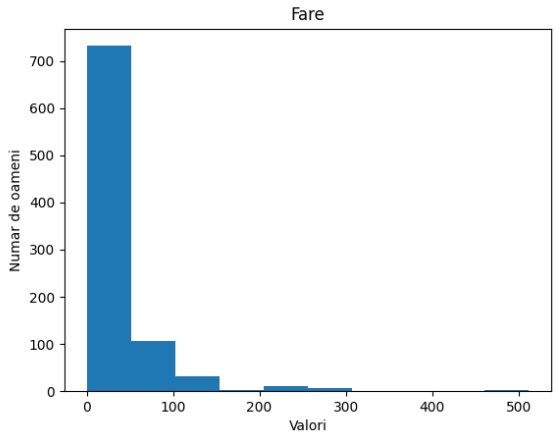
Cerinta 3

Functia `histograms(df)` parcurge toate coloanele numerice din dataframe si construieste o histograma pentru fiecare in parte.









Cerinta 4

Functia `nullColID(df)` identifica coloanele ce contin valori lipsa. Pentru numarul si proportia valorilor lipsa din coloane se foloseste functia `nullCols(df)`. Procentul acestora pentru fiecare dintre cele doua lase se determina cu ajutorul functiei `prcNullCols(df)`.

Coloane cu valori lipsa: ‘Age’, ‘Cabin’, ‘Embarked’.

Numarul si proportia valorilor lipsa:

Age: 177, 24.79%

Cabin: 687, 336.76%

Embarked: 2, 0.22%

Procentul acestora pentru fiecare dintre cele doua clase (0/1 – Starea de supravietuire):

Age: 22.77% / 15.20%

Cabin: 87.61% / 60.23%

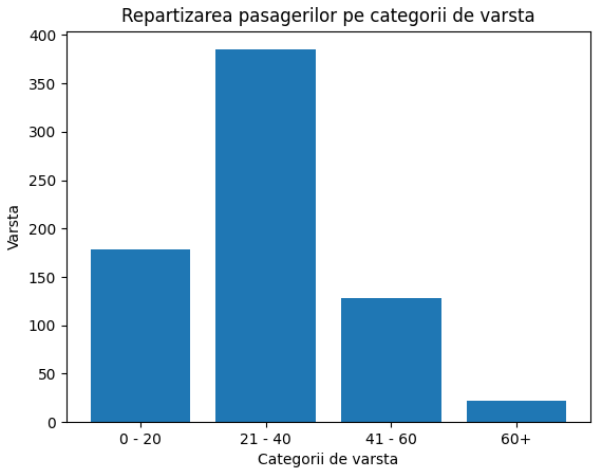
Embarked: 0.0% / 0.58%

Cerinta 5

Numarul de pasageri din fiecare categorie de varsta este determinat cu ajutorul functiei `detAges(df)`. Coloana suplimentara se introduce in urma executarii functiei anterior mentionate. Graficul pentru aceste date se realizeaza cu ajutorul functiei `agesPlot(df)`.

Numar pasageri in functie de varsta:

- [0 – 20]: 179
- [21 – 40]: 385
- [41 – 60]: 128
- [61 – max]: 22

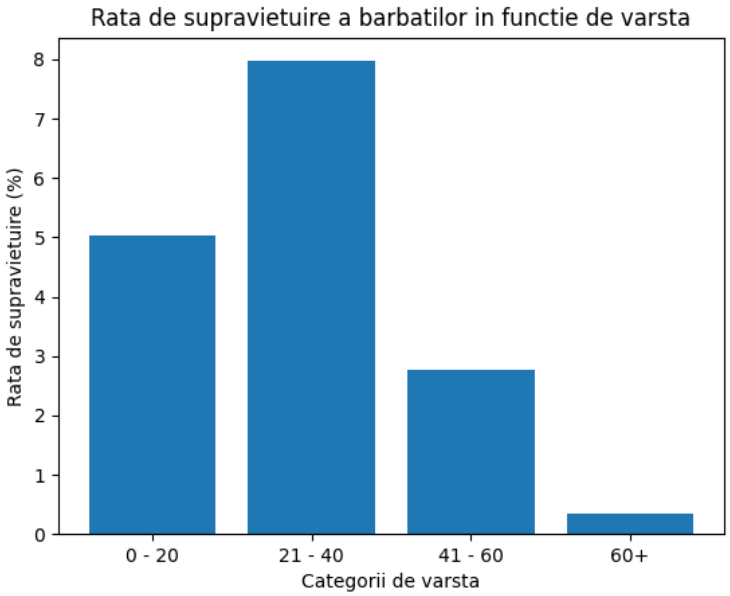


Cerinta 6

Numarul de barbati care au supravietuit pentru fiecare dintre cele 4 categorii este calculat cu ajutorul functiei `detMaleSurv(df)`. Graficul ce evidentiaza modul in care varsta influenteaza rata de supravietuire a barbatilor este construit de functie `maleSurvivalRate(df)`.

Numar pasageri in functie de varsta:

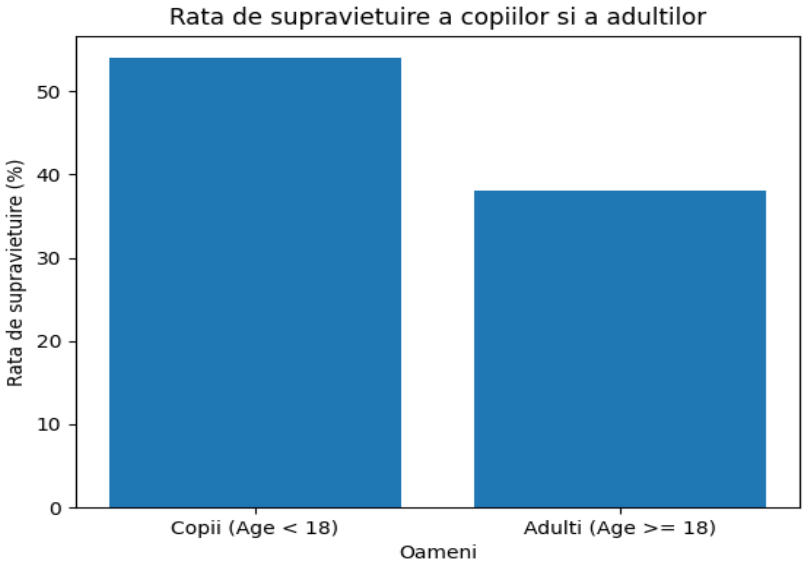
- [0 – 20]: 29
- [21 – 40]: 46
- [41 – 60]: 16
- [61 – max]: 2



Cerinta 7

Procentul copiilor aflati la bord este calculat de functia `prcChildren(df)`. Graficul care evidentiaza rata de supravietuire pentru copii si pentru adulti este realizat de functia `caSurvivalRate(df)`.

Procentul de copii aflati la bord este de 12.68%



Cerinta 8

Valorile lipsa din dataframe sunt completate cu ajutorul functiei `fillEmpty(df)` din `main.py`.

Cerinta 9

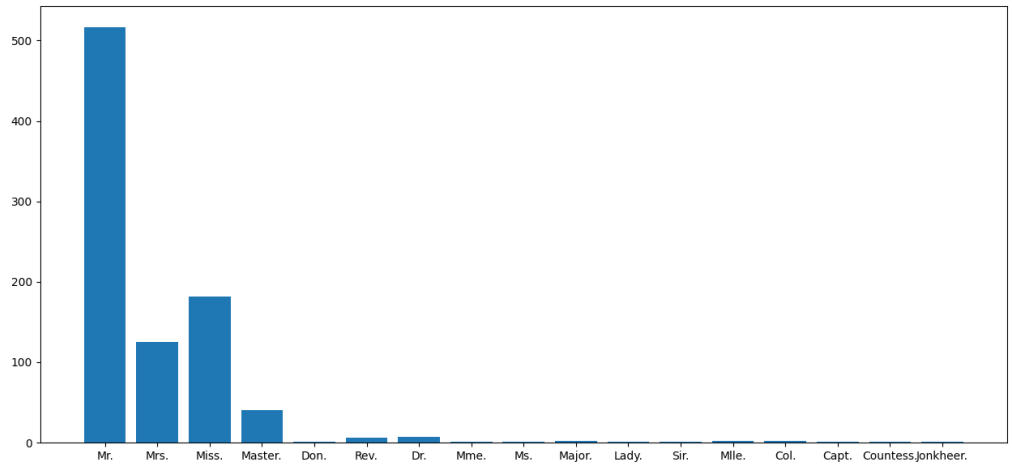
Am folosit urmatorul cod pentru a gasi toate titlurile prezente in coloana 'Names':

```
t = []
for index, row in df.iterrows():
    p = row['Name'].split(',')
    p = p[1].strip().split(' ')
    for i in p:
        if i.endswith('.'):
            tit = i
            break
    try:
        i = t.index(tit)
    except ValueError:
```

```
        i = -1
    if i == -1:
        t.append(tit)
```

Am construit manual un dictionar cu toate titlurile si am verificat pentru fiecare persoana daca sexul atribuit corespunde cu titlul. Apoi, calculez cate persoane corespund fiecarui titlu si afisez graficul corespunzator. Am folosit functiile `titles(df)`, `countTitles(df)` si `titleGenderPlot(df)`.

Toate titlurile corespund cu sexul persoanei respective.



Cerinta 10

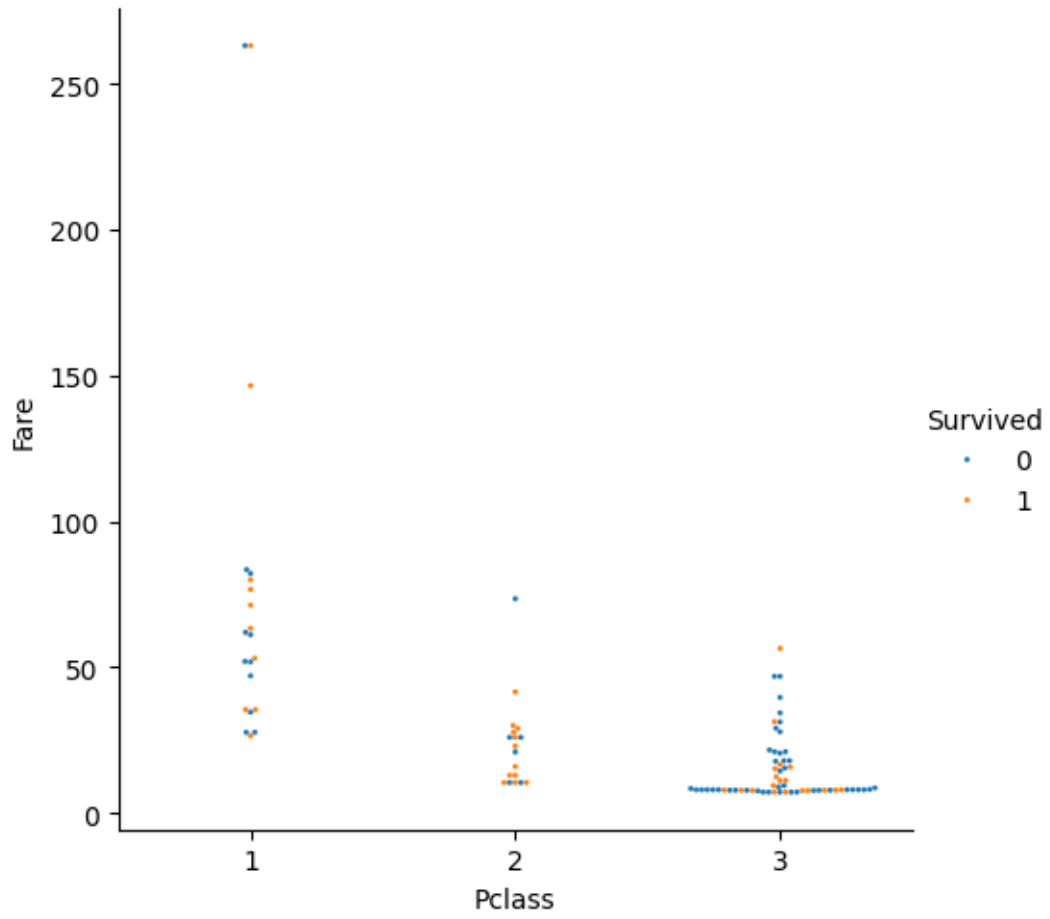
Pentru a verifica daca starea de a fi singur pe Titanic a influentat sansele de supravietuire am folosit functia `familySurvival(df)` din `examine.py`, iar pentru a investiga relatia dintre tarif, clasa si starea de supravietuire pentru primele 100 de inregistrari am creat un grafic folosind `catplot()` din `seaborn` in functia `tcsPlot(df)` din `plots.py`.

34.54% din oamenii care au fost singuri au supravietuit

46.64% din oamenii care au fost cu familia au supravietuit

Concluzie: oamenii care au fost cu familia au avut sanse putin mai mari de supravietuire

Analizand graficul, putem observa ca, pentru primii 100 de pasageri, tariful a crescut cu clasa, iar cei mai multi supravietuitori sunt in clasa a doua.



Descriere functii

main.py

fillEmpty(df) - Completeaza valorile lipsa din dataframe cu cele obtinute pentru media pasagerilor care fac parte din aceeasi clasa.

examine.py

listData(df) - Construieste o lista ce contine informatii despre dataframe:

- Numarul de coloane

- Numarul de linii

- O lista cu tipurile de date

- Numarul de valori lipsa

- Numarul de duplicate

nullColID(df) - Identifica coloanele care contin valori lipsa

prcS(df) - Calculeaza procentul de oameni care au supravietuit si oameni care nu au supravietuit.

prcPclass(df) - Calculeaza procentul de pasageri pentru fiecare clasa

prcSex(df) - Calculeaza procentul de barbati si de femei.

prcNullCols(df) - Calculeaza procentele valorilor lipsa ale fiecare coloanal care contine astfel de elemente pentru fiecare dintre cele doua clase (coloana Survived).

prcChildren(df) - Calculeaza procentul de copii aflati la bord

nullCols(df) - Calculeaza numarul si proportia valorilor lipsa din coloanele care contin astfel de elemente.

detAges(df) - Determina numarul de oameni incadrati in fiecare categorie de varsa si construieste o lista pentru a fi folosita in adaugarea noii coloane in dataframe.

detMaleSurv(df) - Determina numarul de barbati supravietuitori pentru fiecare categorie de varsta.

detChildAdultSurvivalRate(df) - Determina rata de supraviatuire a copiilor si a barbatilor.

titles(df) - Numara cati oameni au titlul corespunzator cu sexul si cati nu.

familySurvival(df) - Calculeaza procentul de oameni care au fost singuri si au supravietuit, respectiv de oameni care au fost cu familia si au suravietuit.

plots.py

pies(df) - Folosind array-uri `numpy`, plot-uri `matplotlib` construieste si functii din `examine` construieste pie chart-uri pentru procentajele pentru numarul de oameni care au supravietuit/murit, pentru numarul de pasageri din fiecare clasa si pentru numarul de barbati/femei.

histograms(df) - Parcurge pe rand fiecare coloana din dataframe, verifica daca este numerica si, daca da, ii construieste o histograma.

agesPlot(df, ageList) - Construieste graficul ce reprezinta repartitia pe categorii de varsta a pasagerilor.

maleSurvivalRate(df) - Construieste un grafic ce reprezinta cum influenteaza varsta procentul de supravietuire al barbatilor.

caSurvivalRate(df) - Construieste un grafic ce reprezinta ratele de supravietuire pentru copii si adulti.

titleGenderPlot(df) - Construieste un garfic asociat numarului de oameni carora le corespunde titlul cu sexul si carora nu.

tcsPlot(df) - Construieste un grafic in functie de tarif, clasa si starea de supravietuire pentru primele 100 de inregistrari din df.