Airline Passenger Satisfaction

Adatbányászat projekt

Tartalomjegyzék

1. Célok

A projekt során a Kaggle Airline Passenger Satisfaction adathalmazzal foglalkoztunk. Célunk két modell segítségével megjósolni az utasok elégedettségét.

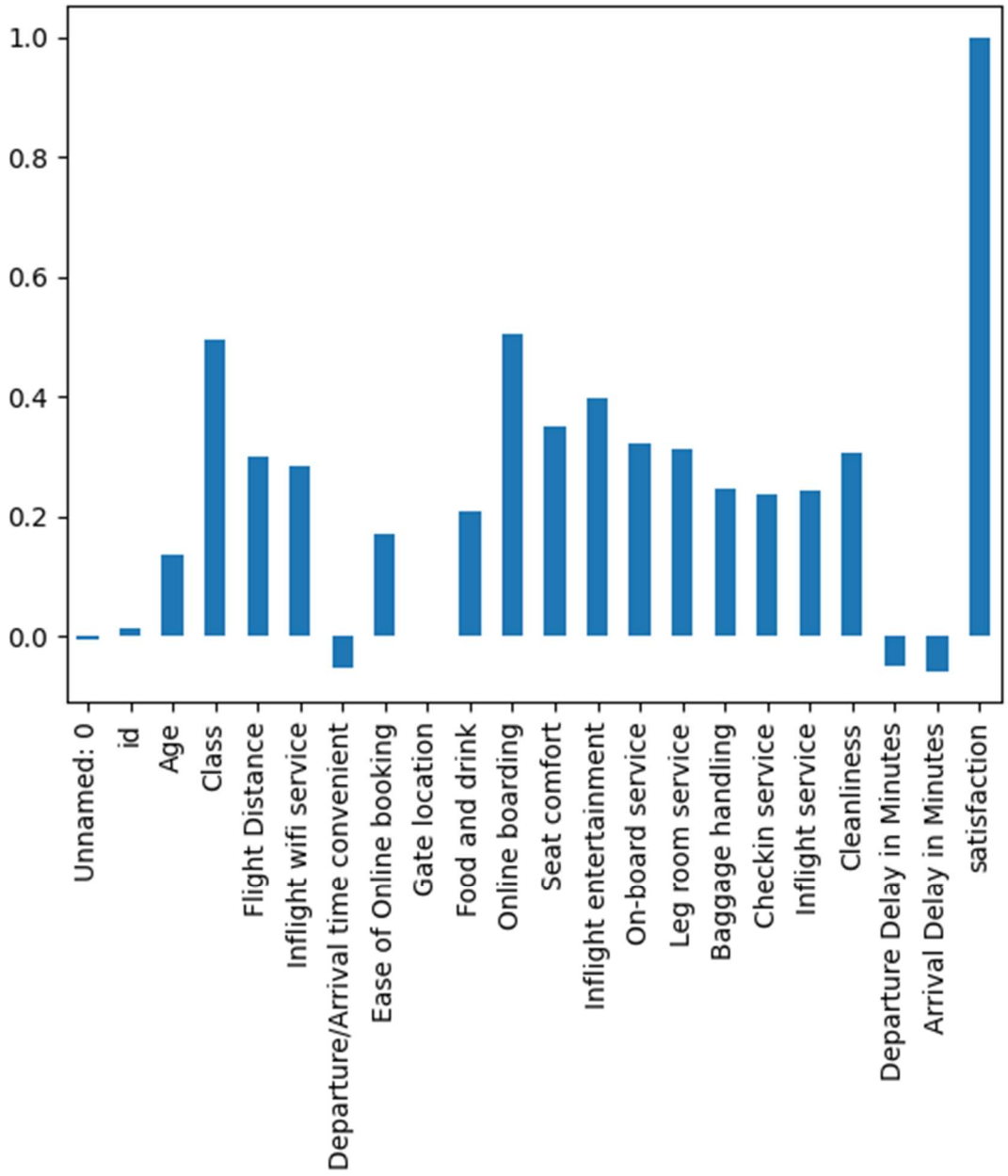
2. Az adat

Az adathalmaz szét van szedve train és test halmazra. Mind a kettő 25 oszloppal, a test 25976, míg a train 103904 sorral rendelkezik. A rekordok object, int és float típusú változókat is tartalmaznak.

3. Az adathalmaz leírása

Unnamed	Sorszám
id	Egyedi azonosító
Gender	Az utas neme
Customer Type	Hűséges-e az utas?
Age	Az utas kora
Type of Travel	Az utazás célja
Class	Utasosztály
Flight Distance	Utazási távolság
Inflight wifi service	Van-e wifi?
Departure/Arrival time conveniant	Indulási és érkezési idő megfelelősége
Ease of Online booking	Online foglalással való elégedettség
Gate location	Kapu elhelyezkedése
Food and drink	Étel és ital
Online boarding	Online checkin
Seat Comfort	Ülések kényelme
Inflight entertainment	Fedélzeti szórakoztatás
On-board service	Fedélzeti kiszolgálás
Leg room service	Lábhely
Baggage handling	Csomagkezelés
Checkin service	Chekin
Inflight service	Fedélzeti szolgáltatás
Cleanliness	Tisztaság
Departure Delay in Minutes	Indulási késés percben
Arrival Delay in Minutes	Érkezési késés percben
satisfaction	Elégedettség

4. Statisztikai jellemzés

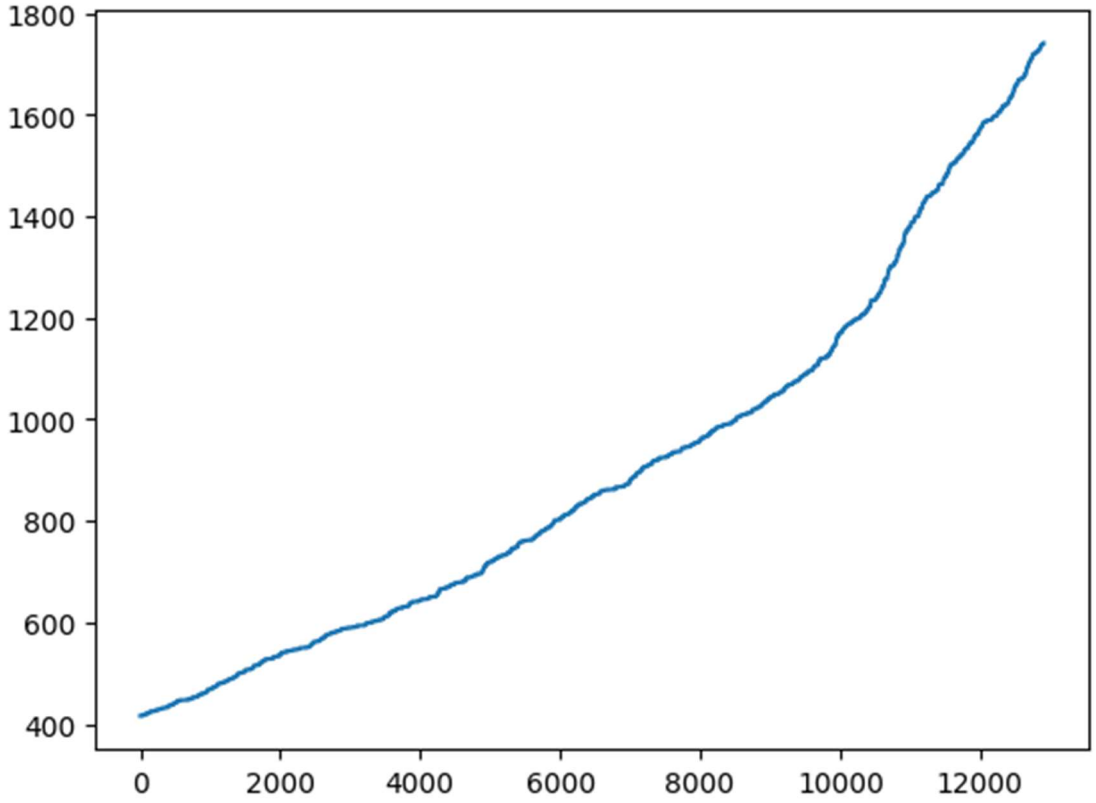


Az ábra az elégedettség és a többi oszlop korrelációját vizsgálja. A ’Class’ és az ’Online boarding’ mutatja a legmagasabb korrelációt a célváltozóval.

5. Hipotézis

A feladatot kétféleképpen tudjuk megfogalmazni: klasszifikációként vagy regresszióként. A regresszió kihasználja, hogy numerikus célfüggvénnyel dolgozunk a klasszifikációval ellentétben, de folytonos értéket ad vissza. A klasszifikáció figyelembe veszi az adathalmaz diszkrétségét.

Mivel a ’Flight Distance’ oszlopban a 3. és az 1. kvartilis különbsége 1330, ezért a repülési távolságok különbsége miatt úgy gondoljuk, hogy az adathalmazunk diszkrét, ezért a klasszifikációs modell fog pontosabb eredményt adni. Az ábrán látszódik, hogy ezek a távolságok eltérnek egymástól.



6. Adatok feldolgozása

Beolvasás után kiírattuk az első néhány sort, hogy információt kapjunk a változók típusáról. Ezt követően töröltük az összes ’na’ mezőt a táblázatból, hogy ne zavarjanak be a modellezés alatt.

A ’satisfaction’ és a ’Class’ oszlopokat átkonvertáltuk numerikus változókká. A ’satisfaction’ oszlopban a ’satisfied’ ’1’, a ’neutral or dissatisfied’ ’0’ lett. A ’Class’ oszlopban az ’Eco’ ’0’, az ’Eco Plus’ ’1’ és a ’Business’ ’2’ lett.

A ’train’ és ’test’ adathalmazokban a kategórikus és numerikus változókat tartalmazó oszlopokat különválasztva a hiányzó adatok helyére a kategórikusoknál a leggyakrabban előforduló elemet, a numerikusoknál az átlagot adtuk meg.

Ahhoz, hogy az átírások meg tudjanak valósulni, Pipeline-t és Seaborn-t használtunk.

7. Elégedettség kor és osztály szerint

Készítettünk 4 korcsoportot. Érdekelt minket, hogy melyik korosztály melyik osztályon mennyire van megelégedve.

A korosztályok az alábbiak szerint lettek létrehozva:

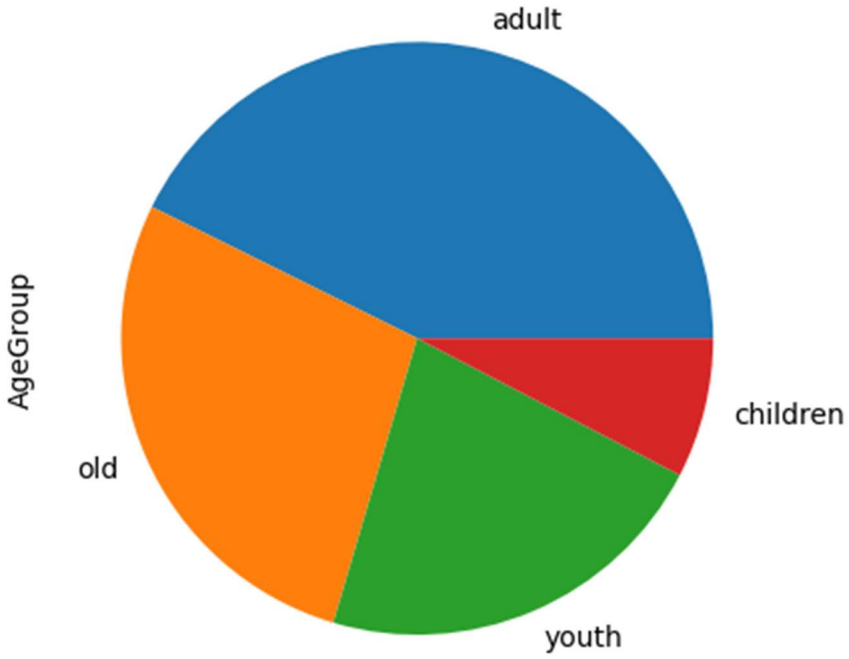
0 <= age < 18 : children

18 <= age < 30: youth

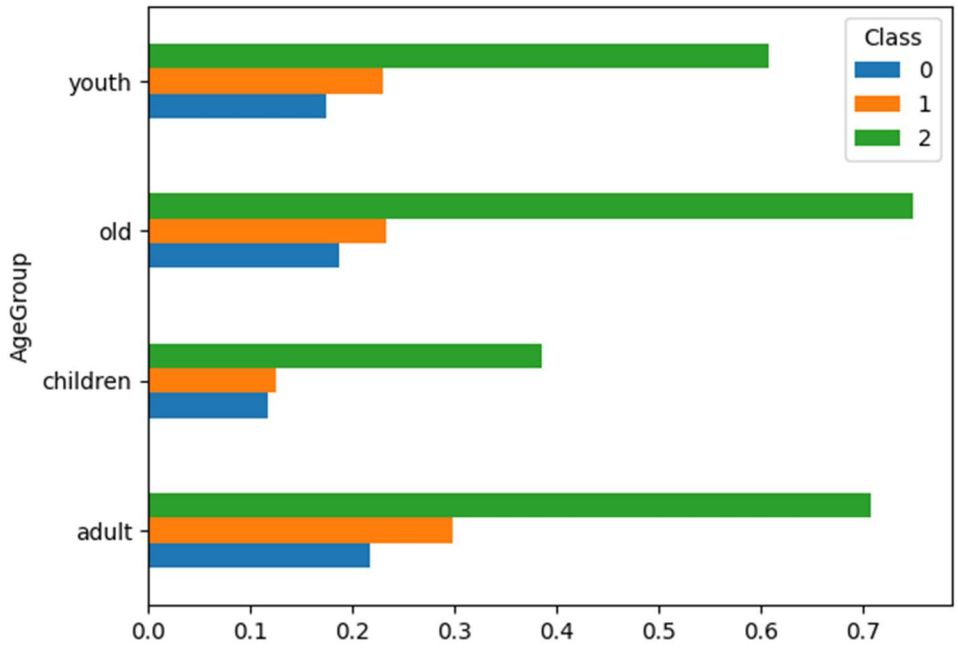
30 <= age < 50: adult

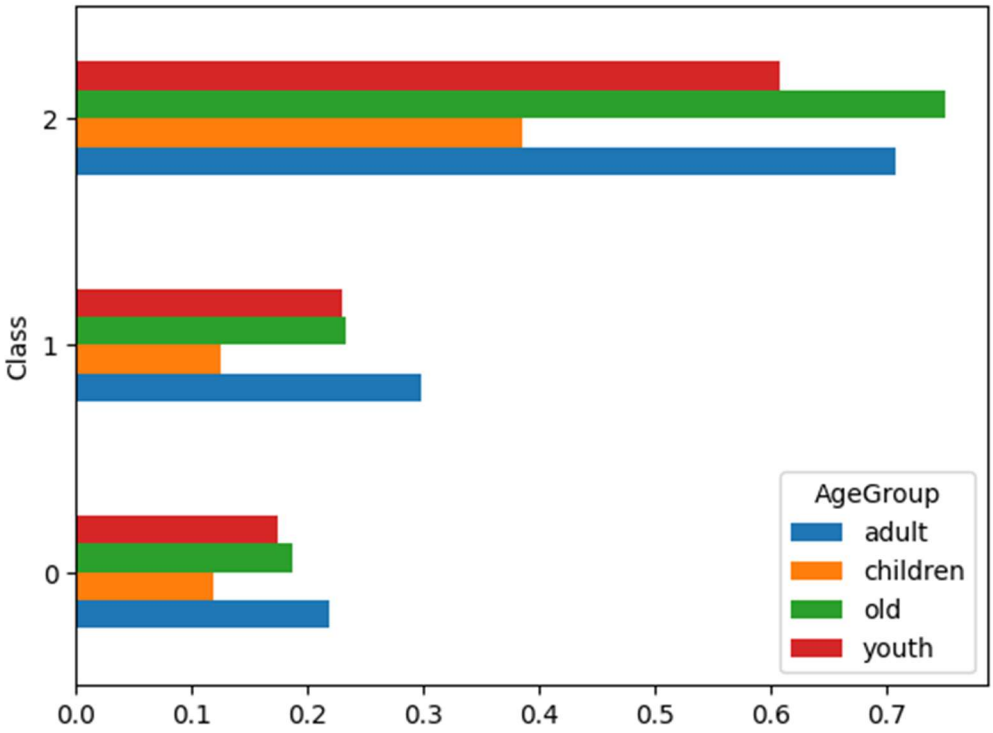
50 <= age : old

Az utasok eloszlása kor szerint:



Az utasok elégedettsége kor és osztály szerint:





Látható, hogy a ’2’-es(Business) jelzésű osztály Kiemelkedik a másik 2 osztálytól elégedettség terén minden korosztálynál.

8. Modellek

Mi a Random Forest Classifier-t és a Logistic Regression-t próbáltuk ki. Ahhoz, hogy a modelljeink működjenek az előre feldolgozott adatokkal Pipeline-t használtunk, ami lehetővé teszi, hogy a különböző adatmódosításokkal tudjunk predikciót adni.

9. Kiértékelés

A predikcióinkat összehasonlítva a ’test’ halmaz ’satisfaction’ oszlopával, amin tudjuk ellenőrizni a modellünk eredményét, mivel a satisfaction oszlop tartalmazza, hogy valaki elégedett-e a szolgáltatással vagy sem.

Mind a két modellünkön használtunk accuracy_score-t és confusion_matrix-ot.

A Random Forest Classifier 96,52 %, a Logistic Regression 65,49 % pontossággal adta meg az utasok elégedettségét.

Ezeket confusion matrix-ban is ábrázoltuk:

```
array([[14289,   284],
       [  620, 10783]])
```

```
array([[7043, 7530],
       [1435, 9968]])
```

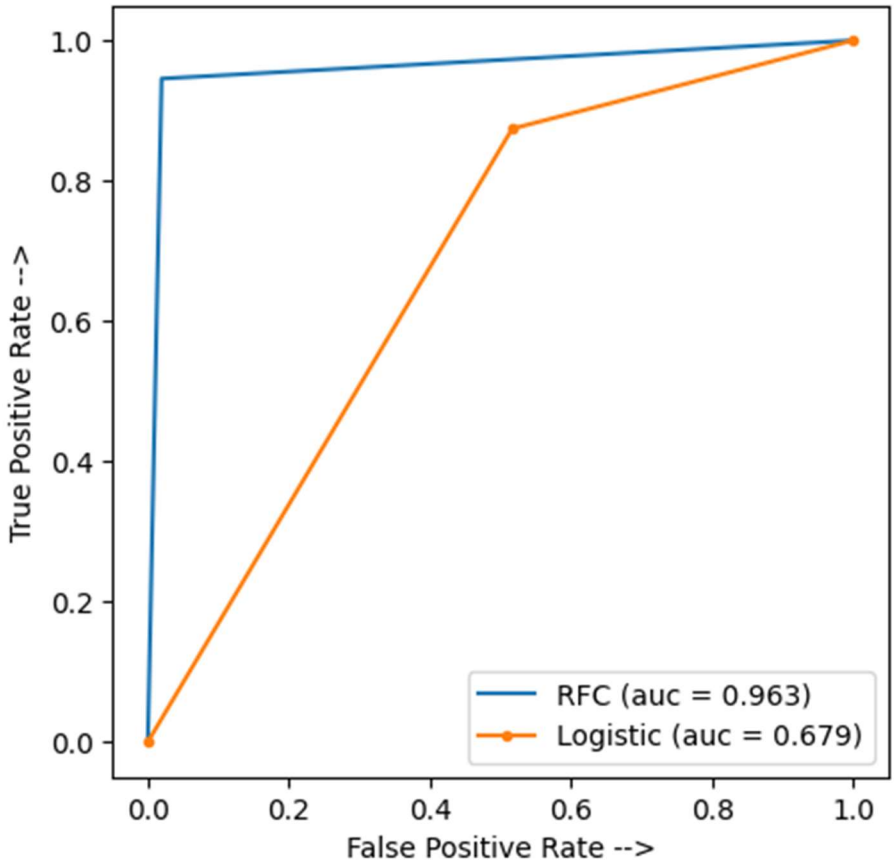
A főátlóban azok szerepelnek, akiket jól határoztunk meg. Az (1,1) elem azok, akiket ’neutral or dissatisfied’-nak gondoltunk és ténylegesen azok is. Az (2,1) elem azok, akiket ’neutral or dissatisfied’-nak gondoltunk, pedig ’satisfied’. A (2,2) elem azok, akiket

’satisfied’-nak gondoltunk és ténylegesen azok is. Az (1,2) elem azok, akiket ’ satisfied’-nak gondoltunk, pedig ’ neutral or dissatisfied’.

A Random Forest Classifier-nél 14289 utasról gondoltuk jól, hogy ’neutral or dissatisfied’, ez 98%-os pontosság. 10783 utasról gondoltuk jól, hogy ’satisfied’ ez 93,8%-os pontosság. 284 utasról gondoltuk rosszul, hogy ’satisfied’, és 620-ról, hogy ’neutral or dissatisfied’.

A Logistic Regression-nél 7043 utasról gondoltuk jól, hogy ’neutral or dissatisfied’, ez 48%-os pontosság. 9968 utasról gondoltuk jól, hogy ’satisfied’ ez 87%-os pontosság. 7530 utasról gondoltuk rosszul, hogy ’satisfied’, és 1435-ról, hogy ’neutral or dissatisfied’.

Ezt követően összehasonlítottuk a két modellt a ’roc_curve’ segítségével.



10.Összefoglalás

A projekt során többször is belefutottunk hibakódokba, de szerencsére sikerült leküzdeni az akadályokat. Ennek köszönhetően otthonosabban mozgunk a python világában.

Arra számítottunk, hogy a klasszifikációs modell fog pontosabb eredményt adni. Meglepetésként ért minket, hogy mennyivel jobban szerepelt a ’Random Forest Classifier’ a ’Logistic Regression’-nél.