import pandas as pd

Kural Tabanlı Sınıflandırma ile Potansiyel MüÅŸteri Getirisi Hesaplama

Abdurrahman Bulut

İÅŸ Problemi

Bir oyun şirketi mþÅŸterilerinin bazı özelliklerini kullanarak seviye tabanlı (level based) yeni müÅŸteri tanımları (persona)

oluşturmak ve bu yeni müÅŸteri tanımlarına göre segmentler oluÅŸturup bu segmentlere göre yeni gelebilecek müÅŸterilerin ÅŸirkete

ortalama ne kadar kazandırabileceÄŸini tahmin etmek istemektedir.

Ã-rneÄŸin: Týrkiye'den IOS kullanıcısı olan 25 yaşındaki bir erkek kullanıcının ortalama ne kadar kazandırabileceÄŸi belirlenmek isteniyor.

Veri Seti Hikayesi

Persona.csv veri seti uluslararası bir oyun ÅŸirketinin sattığı þrþnlerin fiyatlarını ve bu þrþnleri satın alan kullanıcıların bazı

demografik bilgilerini barındırmaktadır. Veri seti her satıÅŸ iÅŸleminde oluÅŸan kayıtlardan meydana gelmektedir. Bunun anlamı tablo

tekilleştirilmemiştir. Diğer bir ifade ile belirli demografik özelliklere sahip bir kullanıcı birden fazla alıÅŸveriÅŸ yapmıÅŸ olabilir.

Price: MüÅŸterinin harcama tutarı

Source: MüÅŸterinin baÄŸlandığı cihaz türü

Sex: MüÅŸterinin cinsiyeti # Country: MüÅŸterinin ülkesi # Age: MüÅŸterinin yaşı

#		PRICE	SOURCE	SEX	COUNTRY	AGE
#	0	39	android	male	bra	17
#	1	39	android	male	bra	17
#	2	49	android	male	bra	17
#	3	29	android	male	tur	17
#	4	49	android	male	tur	17

#	customers_level_based	PRICE	SEGMENT
# 0	BRA_ANDROID_FEMALE_0_18	1139.800000	Α
# 1	BRA_ANDROID_FEMALE_19_23	1070.600000	Α
# 2	BRA_ANDROID_FEMALE_24_30	508.142857	Α
# 3	BRA_ANDROID_FEMALE_31_40	233.166667	C
# 4	BRA ANDROID FEMALE 41 66	236.666667	C

PROJE GÃ-REVLERİ

GÃ-REV 1: AÅŸağıdaki soruları yanıtlayınız.


```
# Soru 1: persona.csv dosyasını okutunuz ve veri seti ile ilgili genel bilgileri gösteriniz.
df = pd.read_csv("modul2/tasks/persona.csv")
df.head()
df.describe().T
df.info()
```

```
# Soru 2: Kaç unique SOURCE vardır? Frekansları nedir?
df['SOURCE'].nunique() # unique source sayısı
df['SOURCE'].value_counts() # Frekansları
# Soru 3: Kaç unique PRICE vardır?
df['PRICE'].nunique()
# Soru 4: Hangi PRICE'dan kaçar tane satıÅŸ gerçekleÅŸmiÅŸ?
df['PRICE'].value_counts()
# Soru 5: Hangi ülkeden kaçar tane satıÅŸ olmuÅŸ?
df['COUNTRY'].value_counts()
# Soru 6: Ülkelere göre satıÅŸlardan toplam ne kadar kazanılmıÅŸ?
df.groupby('COUNTRY')['PRICE'].sum()
# Soru 7: SOURCE tÃ%rlerine göre göre satıÅŸ sayıları nedir?
df.groupby('SOURCE').size()
# Soru 8: Ülkelere göre PRICE ortalamaları nedir?
df.groupby('COUNTRY')['PRICE'].mean()
# Soru 9: SOURCE'lara göre PRICE ortalamaları nedir?
df.groupby('SOURCE')['PRICE'].mean()
# Soru 10: COUNTRY-SOURCE kä±rä±lä±mä±nda PRICE ortalamalarä± nedir?
df.groupby(['COUNTRY', 'SOURCE'])['PRICE'].mean()
# GÃ-REV 2: COUNTRY, SOURCE, SEX, AGE kırılımında ortalama kazançlar nedir?
task2Df = df.groupby(['COUNTRY', 'SOURCE', 'SEX', 'AGE'])['PRICE'].mean()
# GÃ-REV 3: Çıktıyı PRICE'a göre sıralayınız.
# Ã-nceki sorudaki çıktıyı daha iyi görebilmek için sort_values metodunu azalan olacak
şekilde PRICE'a uygulayınız.
# ćıktıyı agg_df olarak kaydediniz.
agg_df = task2Df.sort_values(ascending=False)
agg_df
# GÃ-REV 4: Indekste yer alan isimleri deÄŸiÅŸken ismine çeviriniz.
# Üçýncý sorunun çıktısında yer alan PRICE dışındaki tým deÄŸiÅŸkenler index
isimleridir.
# Bu isimleri değişken isimlerine çeviriniz.
# İpucu: reset index()
# agg df.reset index(inplace=True) # Cannot reset index inplace on a Series to create a DataFrame
agg df = agg df.reset index() # Works
agg df
# GÃ-REV 5: AGE deÄŸiÅŸkenini kategorik deÄŸiÅŸkene çeviriniz ve agg df'e ekleyiniz.
# Age sayısal deÄŸiÅŸkenini kategorik deÄŸiÅŸkene çeviriniz.
# Aralıkları ikna edici olacağını düşündüğünüz ÅŸekilde oluÅŸturunuz.
# \tilde{A}-rne\tilde{A}in: '0_18', '19_23', '24_30', '31_40', '41_70' quantiles = ['0_18', '19_23', '24_30', '31_40', '41_70']
agg_df['AGE_CAT'] = pd.qcut(agg_df['AGE'], q=5, labels=quantiles)
agg df.head()
agg df.tail()
# GÃ-REV 6: Yeni level based müÅŸterileri tanımlayınız ve veri setine deÄŸiÅŸken olarak
# customers level based adında bir deÄŸiÅŸken tanımlayınız ve veri setine bu deÄŸiÅŸkeni
ekleyiniz.
# Dikkat!
```

```
# list comp ile customers level based deÄŸerleri oluÅŸturulduktan sonra bu deÄŸerlerin
tekilleÅŸtirilmesi gerekmektedir.
# Ã-rneÄŸin birden fazla ÅŸu ifadeden olabilir: USA_ANDROID_MALE_0_18
# Bunları groupby'a alıp price ortalamalarını almak gerekmektedir.
# Yeni seviye tabanlä± mã¼ÅŸterileri tanä±mlama
agg_df['customers_level_based'] = agg_df['COUNTRY'].str.upper() + '_' +
agg_df['SOURCE'].str.upper() + '_' + agg_df[
   'SEX'].str.upper() + '_' + agg_df['AGE_CAT'].str.upper()
agg_df = agg_df.groupby('customers_level_based')
['PRICE'].mean().reset_index().sort_values(ascending=False, by='PRICE')
agg df
# GÃ-REV 7: Yeni müÅŸterileri (USA_ANDROID_MALE_0_18) segmentlere ayırınız.
# PRICE'a göre segmentlere ayırınız,
# segmentleri "SEGMENT" isimlendirmesi ile agg_df'e ekleyiniz,
# segmentleri betimleyiniz,
agg_df['SEGMENT'] = pd.qcut(agg_df['PRICE'], q=4, labels=['D', 'C', 'B', 'A'])
# default observed=true deprecated oluyormuÅŸ
segment_summary = agg_df.groupby('SEGMENT', observed=False)['PRICE'].agg(['mean', 'max', 'sum'])
segment_summary
# GÃ-REV 8: Yeni gelen müÅŸterileri sınıflandırız ne kadar gelir getirebileceÄŸini tahmin
ediniz.
# 33 yaşında ANDROID kullanan bir Türk kadını hangi segmente aittir ve ortalama ne kadar
gelir kazandırması beklenir?
new_user = "TUR_ANDROID_FEMALE_31_40"
agg_df[agg_df['customers_level_based'] == new_user].SEGMENT # B segmenti
# 35 yaşında IOS kullanan bir Fransız kadını hangi segmente ve ortalama ne kadar gelir
kazandırması beklenir?
# Yeni gelen müÅŸterilerin sınıflandırılması ve gelir tahmini
new_user = "FRA_IOS_FEMALE_31_40"
agg_df[agg_df['customers_level_based'] == new_user].SEGMENT # C segmenti
```