YÜKSEK DÜZEY PROGRAMLAMA ÖDEVİ

DUYGU MERT 20201317207

```
import matplotlib.pyplot as plt
import pandas as pd
```

Matplotlib: Python'da veri görselleştirmesi için kullanılan en yaygın kütüphanedir. Kullanıcıların, verilerden grafikler, çizgi grafikleri, histogramlar, çubuk grafikler ve daha fazlasını oluşturmasına olanak tanır. Özellikle bilimsel verileri görselleştirmek için sıklıkla tercih edilir. pyplot modülü, bu grafiklerin hızlı ve kolay bir şekilde çizilmesini sağlayan fonksiyonlar sunar.

Pandas: Veri manipülasyonu ve analizinde en çok kullanılan kütüphanedir. Verileri okuma, yazma, temizleme, dönüştürme ve filtreleme gibi işlemleri kolayca yapmayı sağlar. DataFrame adı verilen veri yapısı, tablolarla çalışmayı çok daha verimli hale getirir. Pandas, özellikle büyük veri setlerinde hızlı analizler yapabilmek için tasarlanmıştır ve veri bilimi, finansal analiz ve zaman serisi analizi gibi birçok alanda yaygın olarak kullanılır.

```
# CSV dosyalarını okuma
item_categories = pd.read_csv("/Users/duygumert/PycharmProjects/pythonProject3/Predict_future_sales/item_categories.csv"
items = pd.read_csv("/Users/duygumert/PycharmProjects/pythonProject3/Predict_future_sales/items.csv")
sales_train = pd.read_csv("/Users/duygumert/PycharmProjects/pythonProject3/Predict_future_sales/sales_train.csv")
sample_submission = pd.read_csv("/Users/duygumert/PycharmProjects/pythonProject3/Predict_future_sales/sample_submission.
shops = pd.read_csv("/Users/duygumert/PycharmProjects/pythonProject3/Predict_future_sales/shops.csv")
test = pd.read_csv("/Users/duygumert/PycharmProjects/pythonProject3/Predict_future_sales/test.csv")
```

Farklı veri kümelerini yükleyerek bir veri analizi ve modelleme sürecine başlamak için **pandas** kütüphanesini kullanır. Yüklenen dosyalar şunlardır: item_categories.csv, ürünlerin kategorileri hakkında bilgi; items.csv, her bir ürünün detaylarını içerir; sales_train.csv, geçmiş satış verilerini barındırır; sample_submission.csv, tahmin formatı için örnek bir dosyadır; shops.csv, mağaza bilgilerini içerir; ve test.csv, gelecekteki satış tahminlerinin yapılacağı test verileridir. Bu dosyalar, verileri analiz etmek ve gelecekteki satışları tahmin etmek için gerekli veriyi sağlar. Yüklenen bu veri setleri, genellikle bir makine öğrenimi modelinin eğitilmesi ve değerlendirilmesi amacıyla kullanılır.

```
# İlk 10 satırı görüntüleme
print("Item Categories:")
print(item_categories.head(10))
print("\nItems:")
print(items.head(10))
print("\nSales Train:")
print(sales_train.head(10))
print("\nSample Submission:")
print(sample_submission.head(10))
print("\nShops:")
print(shops.head(10))
print("\nTest:")
print(test.head(10))
```

Bu kısım, çeşitli veri setlerinin ilk 10 satırını görüntüleyerek her dosyanın yapısı hakkında hızlı bir inceleme yapmanızı sağlar. İlk olarak, **Item Categories** veri seti, ürün kategorilerinin kimlik numaralarını ve isimlerini içerir. Bu, hangi ürünlerin hangi kategoriye ait olduğunu anlamanızı sağlar. **Items** veri seti, her bir ürünün ID'si, adı ve kategorisi gibi bilgileri içerir, böylece her ürünün özelliklerini inceleyebilirsiniz. **Sales Train** veri seti, geçmiş satışlarla ilgili detayları sunar; tarih, mağaza kimliği, ürün kimliği ve satış miktarı gibi bilgiler içerir. **Sample Submission** dosyası, tahmin sonuçlarının formatını gösteren örnek bir şablondur ve model çıktılarını bu formatta kaydetmek için kullanılır. **Shops** veri seti, mağazaların kimlik numaraları ve adları hakkında bilgi sunar, bu da mağaza bazında analiz yapmanızı sağlar. Son olarak, **Test** veri seti, gelecekteki satış tahminleri yapılacak ürün-mağaza kombinasyonlarını içerir. Bu inceleme, veri setlerinin yapısını hızlıca anlamanıza ve modelleme sürecinde nasıl kullanılacaklarını belirlemenize yardımcı olur.

```
# Veri hakkında bilgi
print("\nSales Train Info:")
print(sales_train.info())
```

print(sales_train.info()) komutu, sales_train veri setinin genel bilgilerini görüntüler. Bu fonksiyon, veri setindeki sütun sayısı, her sütunun veri türü, eksik değerlerin olup olmadığı, bellek kullanımı gibi bilgileri gösterir. Bu şekilde, veri seti hakkında temel bir özet alabilirsiniz. Örneğin, veri türlerinin doğru olup olmadığını kontrol edebilir, eksik verileri analiz edebilir ve veri setinin büyüklüğü hakkında bilgi sahibi olabilirsiniz. info() fonksiyonu, genellikle veri temizliği ve analiz sürecine başlamadan önce ilk adım olarak kullanılır.

```
# İstatistiksel özet
print("\nSales Train Describe:")
print(sales_train.describe())
```

print(sales_train.describe()) komutu, sales_train veri setindeki sayısal sütunların istatistiksel özetini sağlar. Bu özet, her bir sayısal sütunun temel özelliklerini içerir. Veri setindeki her bir sütun için toplam geçerli değer sayısı (count), ortalama (mean), standart sapma (std), minimum ve maksimum değerler (min, max), ayrıca yüzde 25, 50 (medyan) ve 75'lik dilimler gibi çeyrek değerler yer alır. Bu istatistikler, verinin dağılımı hakkında genel bir bakış açısı sunar. Örneğin, veri setindeki anormal uç değerleri, genel eğilimleri ve verinin nasıl dağıldığını incelemenizi sağlar. Bu tür bir analiz, veri temizleme ve modelleme sürecinde önemli bir ilk adımdır, çünkü verinin yapısını anlamak ve gerektiğinde ön işleme yapmak için temel bilgiler sunar.

```
# Belirli bir sütunun değerlerini sayma
print("\nValue Counts for 'item_id':")
print(sales_train['item_id'].value_counts())
```

print(sales_train['item_id'].value_counts()) komutu, sales_train veri setindeki item_id sütunundaki her bir benzersiz ürün kimliğinin kaç kez tekrar ettiğini sayar. Yani, her ürünün kaç kez satıldığını veya veri setinde kaç kez göründüğünü belirler. Bu işlem, hangi ürünlerin daha sık satıldığını ya da veri setinde hangi ürünlerin yoğun olduğunu görmek için kullanılır. Bu tür bir analiz, veri setindeki dağılımı anlamanızı sağlar ve ürünlerin satış performansına dair bazı çıkarımlar yapmanıza yardımcı olabilir. Örneğin, bazı ürünlerin çok sayıda kez satıldığını, bazılarının ise nadiren görüldüğünü fark edebilirsiniz.

```
# Histogram çizimi
sales_train.hist(bins=50, figsize=(15, 10))
plt.suptitle("Sales Train Data Histograms")
plt.show()
```

print(sales_train.hist(bins=50, figsize=(15, 10))) komutu, sales_train veri setindeki sayısal sütunlar için histogramlar çizer. Histogram, verilerin dağılımını görsel olarak gösteren bir araçtır ve her sütunun değerlerinin frekansını görselleştirir. Burada, bins=50 parametresi, histogramın 50 farklı aralığa (bin) bölünmesini sağlar, yani veri aralığındaki farklı değerler 50 eşit bölüme ayrılır. figsize=(15, 10) parametresi ise grafiğin boyutlarını belirler, burada genişlik 15 inç, yükseklik ise 10 inç olarak ayarlanmıştır. plt.suptitle("Sales Train Data Histograms") komutu, histogramlar için başlık ekler. Son olarak, plt.show() komutu, çizilen histogramları ekranda gösterir.

Bu görselleştirme, verinin dağılımını anlamanıza yardımcı olur. Örneğin, bazı sütunların değerlerinin yoğunlaşıp yoğunlaşmadığını veya uç değerlere sahip olup olmadığını görmek için kullanılır. Histogramlar, veri setindeki belirgin eğilimleri ve dağılımı hızlı bir şekilde görselleştirir.

PROGRAM ÇIKTILARI:

	item_name	item_id	item_category_id
0	! ВО ВЛАСТИ НАВАЖДЕНИЯ (ПЛАСТ.) D	0	40
1	!ABBYY FineReader 12 Professional Edition Full...	1	76
2	***В ЛУЧАХ СЛАВЫ (UNV) D	2	40
3	***ГОЛУБАЯ ВОЛНА (Univ) D	3	40
4	***КОРОБКА (СТЕКЛО) D	4	40
5	***НОВЫЕ АМЕРИКАНСКИЕ ГРАФФИТИ (UNI) ...	5	40
6	***УДАР ПО ВОРОТАМ (UNI) D	6	40
7	***УДАР ПО ВОРОТАМ-2 (UNI) D	7	40
8	***ЧАЙ С МУССОЛИНИ D	8	40
9	***ШУГАРЛЭНДСКИЙ ЭКСПРЕСС (UNI) D	9	40

	date	date_block_num	shop_id	item_id	item_price	item_cnt_day
0	02.01.2013	0	59	22154	999.00	1.0
1	03.01.2013	0	25	2552	899.00	1.0
2	05.01.2013	0	25	2552	899.00	-1.0
3	06.01.2013	0	25	2554	1709.05	1.0
4	15.01.2013	0	25	2555	1099.00	1.0
5	10.01.2013	0	25	2564	349.00	1.0
6	02.01.2013	0	25	2565	549.00	1.0
7	04.01.2013	0	25	2572	239.00	1.0
8	11.01.2013	0	25	2572	299.00	1.0
9	03.01.2013	0	25	2573	299.00	3.0

	ID	item_cnt_month
0	0	0.5
1	1	0.5
2	2	0.5
3	3	0.5
4	4	0.5
5	5	0.5
6	6	0.5
7	7	0.5
8	8	0.5
9	9	0.5

	shop_name	shop_id
0	!Якутск Орджоникидзе, 56 фран	0
1	!Якутск ТЦ "Центральный" фран	1
2	Адыгея ТЦ "Мега"	2
3	Балашиха ТРК "Октябрь-Киномир"	3
4	Волжский ТЦ "Волга Молл"	4
5	Вологда ТРЦ "Мармелад"	5
6	Воронеж (Плехановская, 13)	6
7	Воронеж ТРЦ "Максимир"	7
8	Воронеж ТРЦ Сити-Парк "Град"	8
9	Выездная Торговля	9

	ID	shop_id	item_id
0	0	5	5037
1	1	5	5320
2	2	5	5233
3	3	5	5232
4	4	5	5268
5	5	5	5039
6	6	5	5041
7	7	5	5046
8	8	5	5319
9	9	5	5003

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2935849 entries, 0 to 2935848
Data columns (total 6 columns):
 #   Column          Dtype  
---  ------          -----  
 0   date            object 
 1   date_block_num  int64  
 2   shop_id         int64  
 3   item_id         int64  
 4   item_price      float64
 5   item_cnt_day    float64
dtypes: float64(2), int64(3), object(1)
memory usage: 134.4+ MB
```

	date_block_num	shop_id	item_id	item_price	item_cnt_day
count	2.935849e+06	2.935849e+06	2.935849e+06	2.935849e+06	2.935849e+06
mean	1.456991e+01	3.300173e+01	1.019723e+04	8.908532e+02	1.242641e+00
std	9.422988e+00	1.622697e+01	6.324297e+03	1.729800e+03	2.618834e+00
min	0.000000e+00	0.000000e+00	0.000000e+00	-1.000000e+00	-2.200000e+01
25%	7.000000e+00	2.200000e+01	4.476000e+03	2.490000e+02	1.000000e+00
50%	1.400000e+01	3.100000e+01	9.343000e+03	3.990000e+02	1.000000e+00
75%	2.300000e+01	4.700000e+01	1.568400e+04	9.990000e+02	1.000000e+00
max	3.300000e+01	5.900000e+01	2.216900e+04	3.079800e+05	2.169000e+03

```
date        date_block_num  shop_id  item_id  item_price  item_cnt_day
23.02.2014  13              50       3423     999.0       1.0             2
01.05.2014  16              50       3423     999.0       1.0             2
05.01.2013  0               54       20130    149.0       1.0             2
23.03.2014  14              21       3423     999.0       1.0             2
31.12.2014  23              42       21619    499.0       1.0             2
                                                                          ..
11.01.2015  24              27       3077     799.0       1.0             1
                                     3335     1499.0      1.0             1
                                     3340     1799.0      1.0             1
                                     3392     499.5       1.0             1
31.12.2014  23              59       22087    99.0        1.0             1
Name: count, Length: 2935843, dtype: int64
```

```
array([[<Axes: title={'center': 'date_block_num'}>,
        <Axes: title={'center': 'shop_id'}>],
       [<Axes: title={'center': 'item_id'}>,
        <Axes: title={'center': 'item_price'}>],
       [<Axes: title={'center': 'item_cnt_day'}>, <Axes: >]], dtype=object)
```