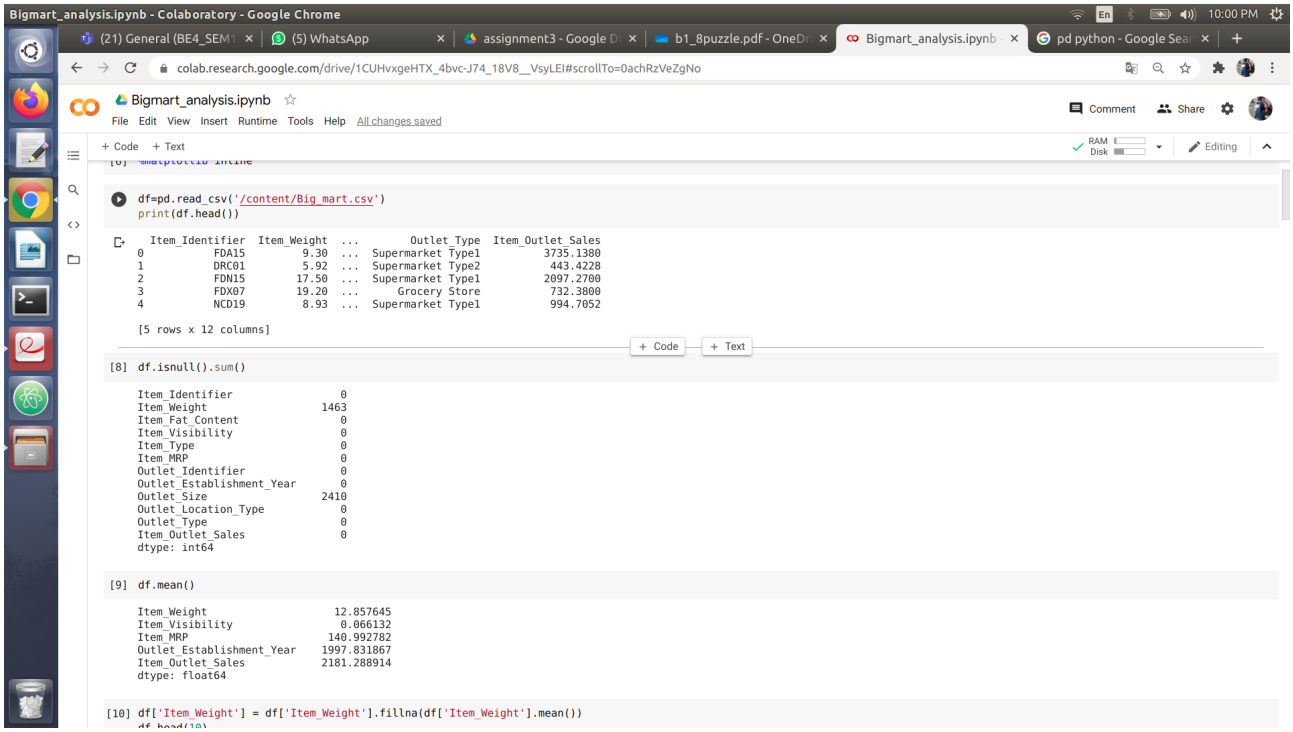
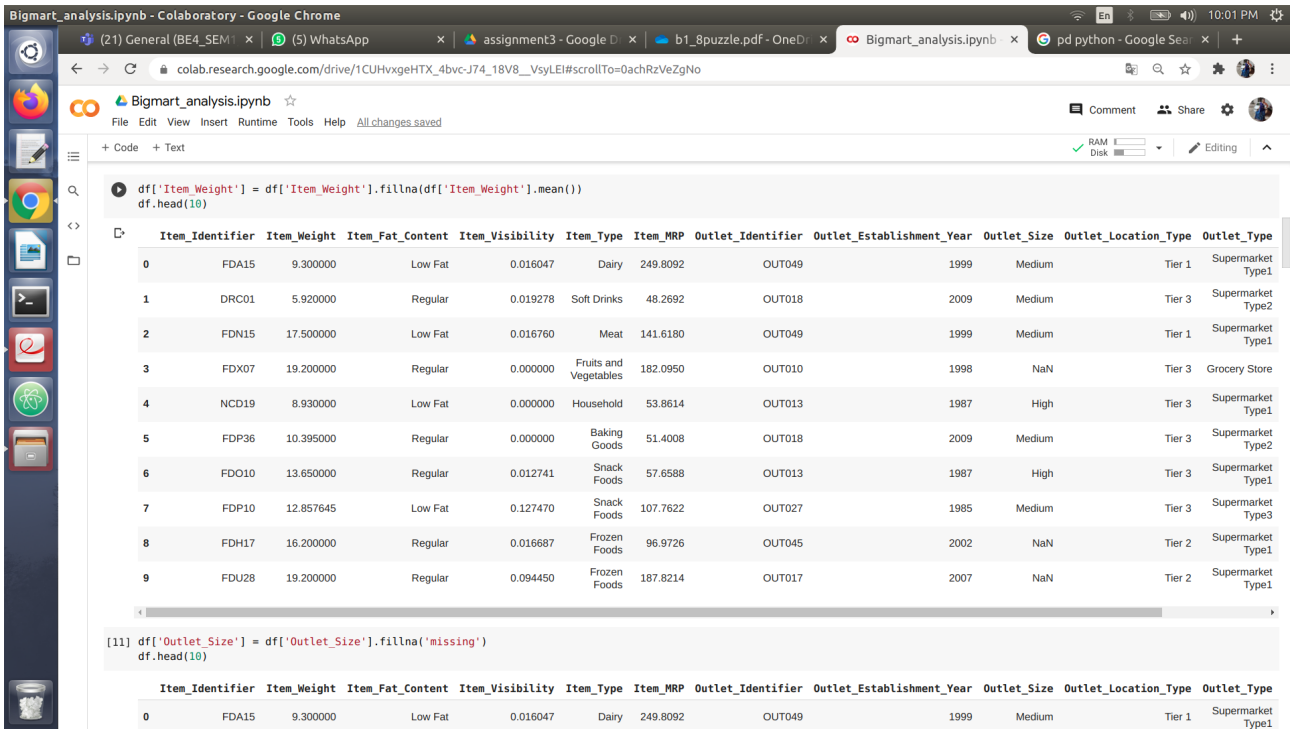
CODE:::

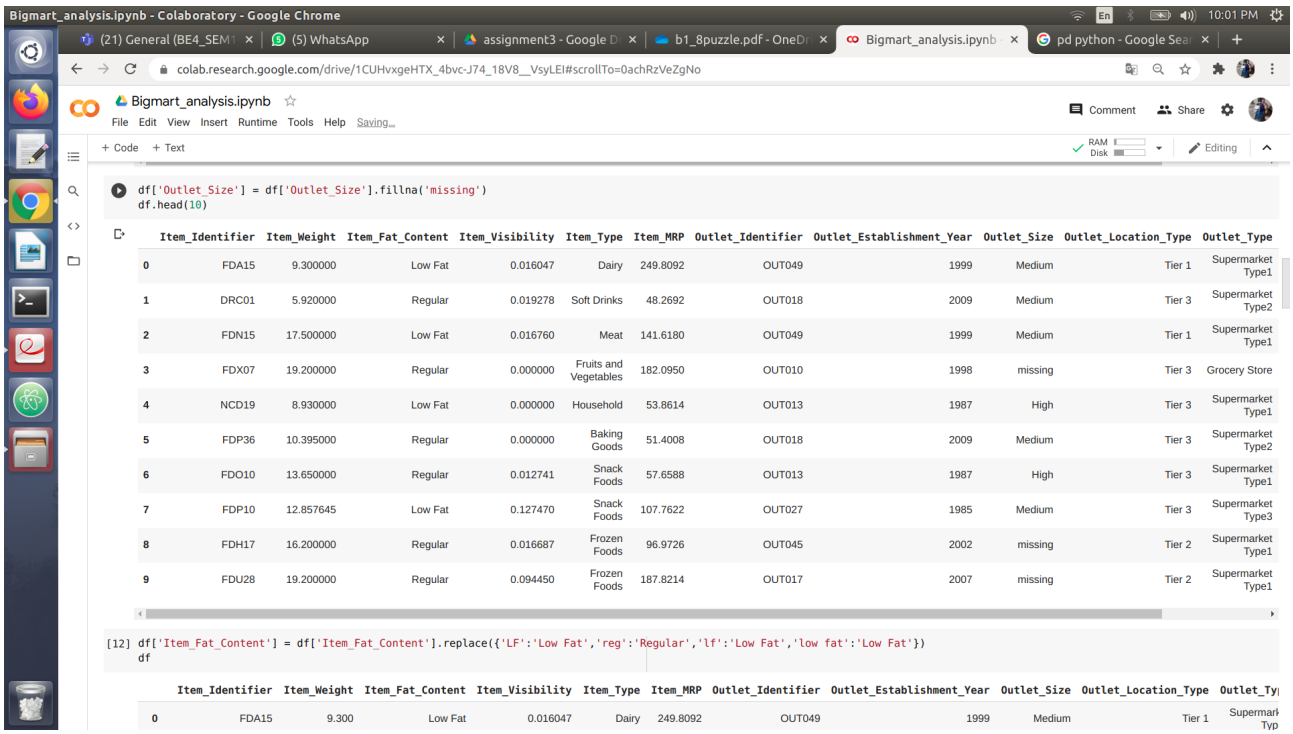
```
# -*- coding: utf-8 -*-
"""Bigmart_analysis.ipynb

Automatically generated by Colaboratory.

Original file is located at
    https://colab.research.google.com/drive/1CUHvxgeHTX_4bvc-J74_18V8__VsyLEI
"""

# Commented out IPython magic to ensure Python compatibility.
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

# %matplotlib inline

df=pd.read_csv('/content/Big_mart.csv')
print(df.head())

df.isnull().sum()

df.mean()

df['Item_Weight'] = df['Item_Weight'].fillna(df['Item_Weight'].mean())
df.head(10)

df['Outlet_Size'] = df['Outlet_Size'].fillna('missing')
df.head(10)

df['Item_Fat_Content'] = df['Item_Fat_Content'].replace({'LF':'Low Fat','reg':'Regular','lf':'Low
Fat','low fat':'Low Fat'})
df

df['Item_Visibility'] = df['Item_Visibility'].replace(0,df['Item_Visibility'].mean())
df.head(10)

sns.countplot(y="Outlet_Size", data=df)
sns.despine()

sns.countplot(y = 'Item_Fat_Content', data = df)
sns.despine()

sns.countplot(y = 'Outlet_Type', data = df)
sns.despine()

sns.countplot(y = 'Outlet_Establishment_Year', data = df)
sns.despine()

sns.countplot(y = 'Outlet_Location_Type', data = df)
sns.despine()
```

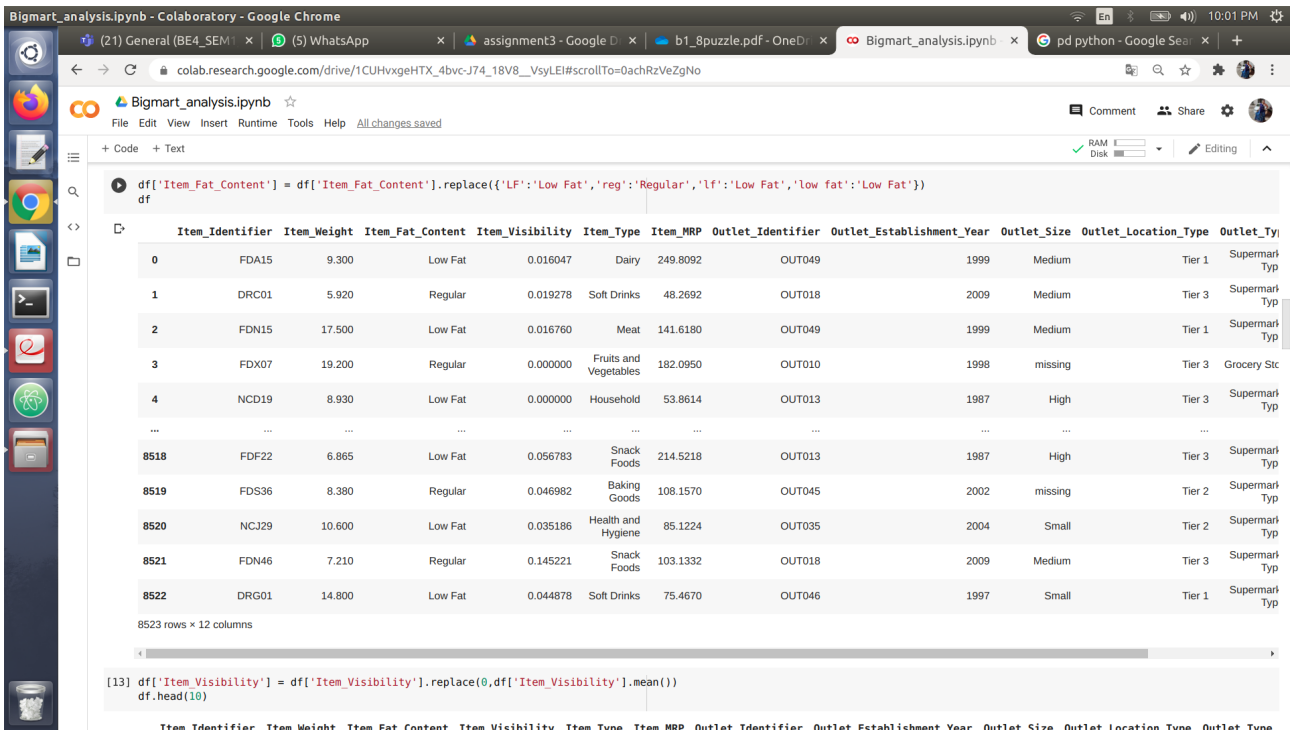
```
sns.regplot(x = 'Item_MRP',y ='Item_Outlet_Sales',data = df , x_jitter=0.2,
scatter_kws={'alpha':0.1})
sns.despine()

grid = sns.FacetGrid(df, col='Outlet_Establishment_Year',col_wrap = 2)
grid.map(plt.scatter,'Item_MRP','Item_Outlet_Sales',alpha = 0.1)
sns.despine()

from sklearn import preprocessing
le = preprocessing.LabelEncoder()

li = ['Outlet_Type','Outlet_Location_Type','Outlet_Size','Item_Fat_Content','Outlet_Identifier']
for i in li:
    df[i] = le.fit_transform(df[i])

df.head()

#splitting data dor train and test
dum = pd.get_dummies(df['Outlet_Identifier'])
df1 = pd.concat([df,dum],axis=1)

label = df1['Item_Outlet_Sales']

train =
df1.drop(columns=['Item_Identifier','Item_Weight','Item_Outlet_Sales','Item_Type','Outlet_Establis
hment_Year'])

from sklearn.model_selection import train_test_split

x_train , x_test , y_train , y_test = train_test_split(train , label , test_size = 0.40,random_state = 101)

"""##  Linear Regression

### Preparing the model and importing necessary packages
"""

from sklearn.linear_model import LinearRegression

reg = LinearRegression()

"""### Fitting the model """

reg.fit(x_train,y_train)

"""### Here we find the accuracy score of our Linear Regression model"""

reg.score(x_test,y_test)
```

OUTPUT:::


Bigmart_analysis.ipynb - Colaboratory - Google Chrome
colab.research.google.com/drive/1CUHvxgeHTX_4bvc-J74_18V8__VsyLEI#scrollTo=0achRzVeZgNo
Bigmart_analysis.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
df=pd.read_csv('/content/Big_mart.csv')
print(df.head())
Item_Identifier Item_Weight ... Outlet_Type Item_Outlet_Sales
0 FDA15 9.30 ... Supermarket Type1 3735.1380
1 DRC01 5.92 ... Supermarket Type2 443.4228
2 FDN15 17.50 ... Supermarket Type1 2097.2700
3 FDX07 19.20 ... Grocery Store 732.3800
4 NCD19 8.93 ... Supermarket Type1 994.7052
[5 rows x 12 columns]
+ Code + Text
[8] df.isnull().sum()
Item_Identifier 0
Item_Weight 1463
Item_Fat_Content 0
Item_Visibility 0
Item_Type 0
Item_MRP 0
Outlet_Identifier 0
Outlet_Establishment_Year 0
Outlet_Size 2410
Outlet_Location_Type 0
Outlet_Type 0
Item_Outlet_Sales 0
dtype: int64
[9] df.mean()
Item_Weight 12.857645
Item_Visibility 0.066132
Item_MRP 140.992782
Outlet_Establishment_Year 1997.831867
Item_Outlet_Sales 2181.288914
dtype: float64
[10] df['Item_Weight'] = df['Item_Weight'].fillna(df['Item_Weight'].mean())


Bigmart_analysis.ipynb - Colaboratory - Google Chrome
colab.research.google.com/drive/1CUHvxgeHTX_4bvc-J74_18V8__VsyLEI#scrollTo=0achRzVeZgNo
Bigmart_analysis.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
df['Item_Weight'] = df['Item_Weight'].fillna(df['Item_Weight'].mean())
df.head(10)
Item_Identifier Item_Weight Item_Fat_Content Item_Visibility Item_Type Item_MRP Outlet_Identifier Outlet_Establishment_Year Outlet_Size Outlet_Location_Type Outlet_Type
0 FDA15 9.300000 Low Fat 0.016047 Dairy 249.8092 OUT049 1999 Medium Tier 1 Supermarket Type1
1 DRC01 5.920000 Regular 0.019278 Soft Drinks 48.2692 OUT018 2009 Medium Tier 3 Supermarket Type2
2 FDN15 17.500000 Low Fat 0.016760 Meat 141.6180 OUT049 1999 Medium Tier 1 Supermarket Type1
3 FDX07 19.200000 Regular 0.000000 Fruits and Vegetables 182.0950 OUT010 1998 NaN Tier 3 Grocery Store
4 NCD19 8.930000 Low Fat 0.000000 Household 53.8614 OUT013 1987 High Tier 3 Supermarket Type1
5 FDP36 10.395000 Regular 0.000000 Baking Goods 51.4008 OUT018 2009 Medium Tier 3 Supermarket Type2
6 FDO10 13.650000 Regular 0.012741 Snack Foods 57.6588 OUT013 1987 High Tier 3 Supermarket Type1
7 FDP10 12.857645 Low Fat 0.127470 Snack Foods 107.7622 OUT027 1985 Medium Tier 3 Supermarket Type3
8 FDH17 16.200000 Regular 0.016687 Frozen Foods 96.9726 OUT045 2002 NaN Tier 2 Supermarket Type1
9 FDU28 19.200000 Regular 0.094450 Frozen Foods 187.8214 OUT017 2007 NaN Tier 2 Supermarket Type1
[11] df['Outlet_Size'] = df['Outlet_Size'].fillna('missing')
df.head(10)
Item_Identifier Item_Weight Item_Fat_Content Item_Visibility Item_Type Item_MRP Outlet_Identifier Outlet_Establishment_Year Outlet_Size Outlet_Location_Type Outlet_Type
0 FDA15 9.300000 Low Fat 0.016047 Dairy 249.8092 OUT049 1999 Medium Tier 1 Supermarket Type1


Bigmart_analysis.ipynb - Colaboratory - Google Chrome
10:01 PM
(21) General (BE4_SEM1
(5) WhatsApp
assignment3 - Google D
b1_8puzzle.pdf - OneDr
Bigmart_analysis.ipynb
pd python - Google Sear
colab.research.google.com/drive/1CUHvxgeHTX_4bvc-J74_18V8__VsyLEI#scrollTo=0achRzVeZgNo
Bigmart_analysis.ipynb
File Edit View Insert Runtime Tools Help Saving...
Comment
Share
+ Code + Text
RAM
Disk
Editing
df['Outlet_Size'] = df['Outlet_Size'].fillna('missing')
df.head(10)
Item_Identifier Item_Weight Item_Fat_Content Item_Visibility Item_Type Item_MRP Outlet_Identifier Outlet_Establishment_Year Outlet_Size Outlet_Location_Type Outlet_Type
0 FDA15 9.300000 Low Fat 0.016047 Dairy 249.8092 OUT049 1999 Medium Tier 1 Supermarket Type1
1 DRC01 5.920000 Regular 0.019278 Soft Drinks 48.2692 OUT018 2009 Medium Tier 3 Supermarket Type2
2 FDN15 17.500000 Low Fat 0.016760 Meat 141.6180 OUT049 1999 Medium Tier 1 Supermarket Type1
3 FDX07 19.200000 Regular 0.000000 Fruits and Vegetables 182.0950 OUT010 1998 missing Tier 3 Grocery Store
4 NCD19 8.930000 Low Fat 0.000000 Household 53.8614 OUT013 1987 High Tier 3 Supermarket Type1
5 FDP36 10.395000 Regular 0.000000 Baking Goods 51.4008 OUT018 2009 Medium Tier 3 Supermarket Type2
6 FDO10 13.650000 Regular 0.012741 Snack Foods 57.6588 OUT013 1987 High Tier 3 Supermarket Type1
7 FDP10 12.857645 Low Fat 0.127470 Snack Foods 107.7622 OUT027 1985 Medium Tier 3 Supermarket Type3
8 FDH17 16.200000 Regular 0.016687 Frozen Foods 96.9726 OUT045 2002 missing Tier 2 Supermarket Type1
9 FDU28 19.200000 Regular 0.094450 Frozen Foods 187.8214 OUT017 2007 missing Tier 2 Supermarket Type1
[12] df['Item_Fat_Content'] = df['Item_Fat_Content'].replace({'LF':'Low Fat','reg':'Regular','lf':'Low Fat','low fat':'Low Fat'})
df
Item_Identifier Item_Weight Item_Fat_Content Item_Visibility Item_Type Item_MRP Outlet_Identifier Outlet_Establishment_Year Outlet_Size Outlet_Location_Type Outlet_Ty
0 FDA15 9.300 Low Fat 0.016047 Dairy 249.8092 OUT049 1999 Medium Tier 1 Supermark Typ


Bigmart_analysis.ipynb - Colaboratory - Google Chrome
10:01 PM
(21) General (BE4_SEM1
(5) WhatsApp
assignment3 - Google D
b1_8puzzle.pdf - OneDr
Bigmart_analysis.ipynb
pd python - Google Sear
colab.research.google.com/drive/1CUHvxgeHTX_4bvc-J74_18V8__VsyLEI#scrollTo=0achRzVeZgNo
Bigmart_analysis.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
df['Item_Fat_Content'] = df['Item_Fat_Content'].replace({'LF':'Low Fat','reg':'Regular','lf':'Low Fat','low fat':'Low Fat'})
df
Item_Identifier Item_Weight Item_Fat_Content Item_Visibility Item_Type Item_MRP Outlet_Identifier Outlet_Establishment_Year Outlet_Size Outlet_Location_Type Outlet_Ty
0 FDA15 9.300 Low Fat 0.016047 Dairy 249.8092 OUT049 1999 Medium Tier 1 Supermark Typ
1 DRC01 5.920 Regular 0.019278 Soft Drinks 48.2692 OUT018 2009 Medium Tier 3 Supermark Typ
2 FDN15 17.500 Low Fat 0.016760 Meat 141.6180 OUT049 1999 Medium Tier 1 Supermark Typ
3 FDX07 19.200 Regular 0.000000 Fruits and Vegetables 182.0950 OUT010 1998 missing Tier 3 Grocery Sto
4 NCD19 8.930 Low Fat 0.000000 Household 53.8614 OUT013 1987 High Tier 3 Supermark Typ
... ...
8518 FDF22 6.865 Low Fat 0.056783 Snack Foods 214.5218 OUT013 1987 High Tier 3 Supermark Typ
8519 FDS36 8.380 Regular 0.046982 Baking Goods 108.1570 OUT045 2002 missing Tier 2 Supermark Typ
8520 NCJ29 10.600 Low Fat 0.035186 Health and Hygiene 85.1224 OUT035 2004 Small Tier 2 Supermark Typ
8521 FDN46 7.210 Regular 0.145221 Snack Foods 103.1332 OUT018 2009 Medium Tier 3 Supermark Typ
8522 DRG01 14.800 Low Fat 0.044878 Soft Drinks 75.4670 OUT046 1997 Small Tier 1 Supermark Typ
8523 rows × 12 columns
[13] df['Item_Visibility'] = df['Item_Visibility'].replace(0,df['Item_Visibility'].mean())
df.head(10)

Bigmart_analysis.ipynb - Colaboratory - Google Chrome

colab.research.google.com/drive/1CUHvxgeHTX_4bvc-J74_18V8__VsyLEI#scrollTo=0achRzVeZgNo

Bigmart_analysis.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share Editing

+ Code + Text

```
df['Item_Visibility'] = df['Item_Visibility'].replace(0,df['Item_Visibility'].mean())
df.head(10)
```

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Outlet_Location_Type	Outlet_Type
0	FDA15	9.300000	Low Fat	0.016047	Dairy	249.8092	OUT049	1999	Medium	Tier 1	Supermarket Type1
1	DRC01	5.920000	Regular	0.019278	Soft Drinks	48.2692	OUT018	2009	Medium	Tier 3	Supermarket Type2
2	FDN15	17.500000	Low Fat	0.016760	Meat	141.6180	OUT049	1999	Medium	Tier 1	Supermarket Type1
3	FDX07	19.200000	Regular	0.066132	Fruits and Vegetables	182.0950	OUT010	1998	missing	Tier 3	Grocery Store
4	NCD19	8.930000	Low Fat	0.066132	Household	53.8614	OUT013	1987	High	Tier 3	Supermarket Type1
5	FDP36	10.395000	Regular	0.066132	Baking Goods	51.4008	OUT018	2009	Medium	Tier 3	Supermarket Type2
6	FDO10	13.650000	Regular	0.012741	Snack Foods	57.6588	OUT013	1987	High	Tier 3	Supermarket Type1
7	FDP10	12.857645	Low Fat	0.127470	Snack Foods	107.7622	OUT027	1985	Medium	Tier 3	Supermarket Type3
8	FDH17	16.200000	Regular	0.016687	Frozen Foods	96.9726	OUT045	2002	missing	Tier 2	Supermarket Type1
9	FDU28	19.200000	Regular	0.094450	Frozen Foods	187.8214	OUT017	2007	missing	Tier 2	Supermarket Type1

```
[14] sns.countplot(y="Outlet_Size", data=df)
     sns.despine()
```

Bigmart_analysis.ipynb - Colaboratory - Google Chrome

colab.research.google.com/drive/1CUHvxgeHTX_4bvc-J74_18V8__VsyLEI#scrollTo=0achRzVeZgNo

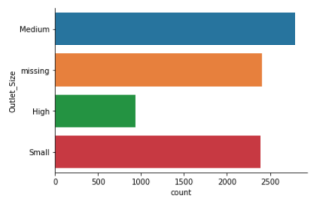
Bigmart_analysis.ipynb

File Edit View Insert Runtime Tools Help All changes saved

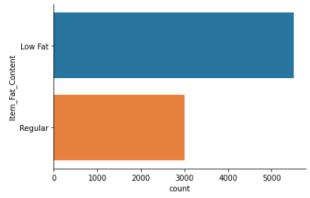
Comment Share Editing

+ Code + Text

```
sns.countplot(y="Outlet_Size", data=df)
sns.despine()
```




```
[15] sns.countplot(y = 'Item_Fat_Content', data = df)
     sns.despine()
```




```
[16] sns.countplot(y = 'Outlet_Type', data = df)
     sns.despine()
```

Bigmart_analysis.ipynb - Colaboratory - Google Chrome

colab.research.google.com/drive/1CUHvxgeHTX_4bvc-J74_18V8__VsyLEI#scrollTo=0achRzVeZgNo

Bigmart_analysis.ipynb

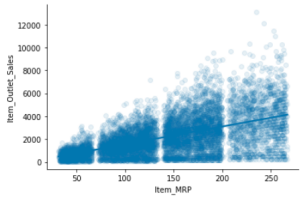
File Edit View Insert Runtime Tools Help Saving...

Comment Share

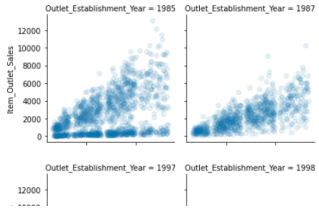
+ Code + Text

RAM Disk Editing

```
sns.regplot(x = 'Item_MRP',y ='Item_Outlet_Sales',data = df , x_jitter=0.2, scatter_kws={'alpha':0.1})
sns.despine()
```




```
[20] grid = sns.FacetGrid(df, col='Outlet_Establishment_Year',col_wrap = 2)
     grid.map(plt.scatter,'Item_MRP','Item_Outlet_Sales',alpha = 0.1)
     sns.despine()
```



Bigmart_analysis.ipynb - Colaboratory - Google Chrome

colab.research.google.com/drive/1CUHvxgeHTX_4bvc-J74_18V8__VsyLEI#scrollTo=0achRzVeZgNo

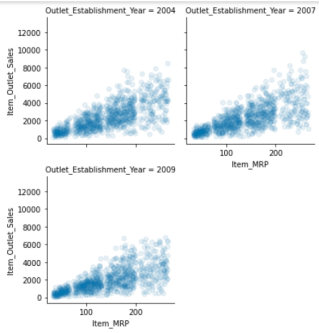
Bigmart_analysis.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

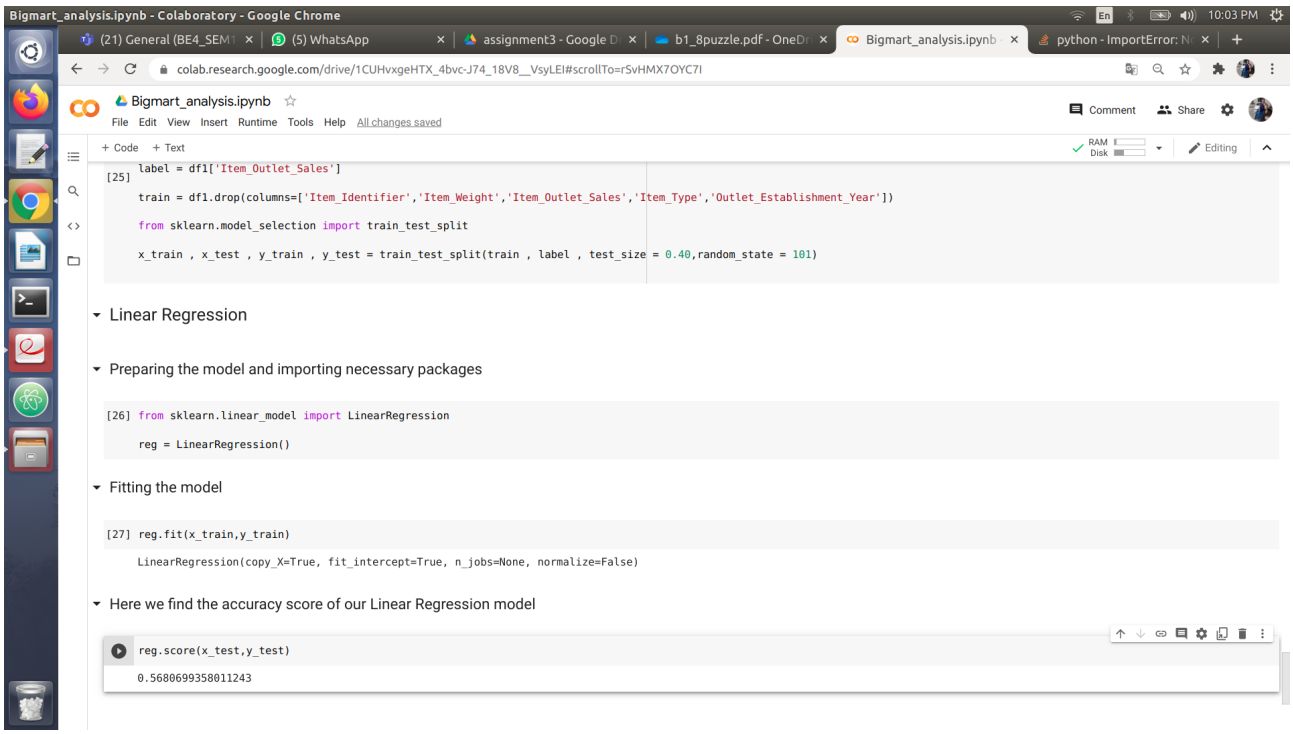
+ Code + Text

RAM Disk Editing




```
[21] from sklearn import preprocessing
     le = preprocessing.LabelEncoder()

     li = ['Outlet_Type','Outlet_Location_Type','Outlet_Size','Item_Fat_Content','Outlet_Identifier']
     for i in li:
         df[i] = le.fit_transform(df[i])

     df.head()
```

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Outlet_Location_Type	Outlet_Type
0	FDA15	9.30	0	0.016047	Dairy	249.8092	9	1999	1	0	1
1	DRC01	5.92	1	0.019278	Soft Drinks	48.2692	3	2009	1	2	2


Bigmart_analysis.ipynb - Colaboratory - Google Chrome
10:03 PM
(21) General (BE4_SEM1
(5) WhatsApp
assignment3 - Google D
b1_8puzzle.pdf - OneDr
Bigmart_analysis.ipynb
python - ImportError: N
colab.research.google.com/drive/1CUHvxgeHTX_4bvc-J74_18V8__VsyLEI#scrollTo=rSvHMX7OYC7I
Bigmart_analysis.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
label = df1['Item_Outlet_Sales']
[25]
train = df1.drop(columns=['Item_Identifier','Item_Weight','Item_Outlet_Sales','Item_Type','Outlet_Establishment_Year'])
from sklearn.model_selection import train_test_split
x_train , x_test , y_train , y_test = train_test_split(train , label , test_size = 0.40,random_state = 101)
Linear Regression
Preparing the model and importing necessary packages
[26] from sklearn.linear_model import LinearRegression
reg = LinearRegression()
Fitting the model
[27] reg.fit(x_train,y_train)
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
Here we find the accuracy score of our Linear Regression model
reg.score(x_test,y_test)
0.5680699358011243