Blinkit Analysis

May 9, 2025

1 PYTHON PROJECT – BLINKIT ANALYSIS

1.0.1 Import Libraries

[1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

1.0.2 Import Raw Data

[2]:

```
df=pd.read_csv('blinkit_data.csv')
```

1.0.3 Sample Data

[3]:

```
df.head()
```

[3]:

```
  Item Fat Content Item Identifier              Item Type  \
0          Regular           FDX32  Fruits and Vegetables
1          Low Fat           NCB42     Health and Hygiene
2          Regular           FDR28           Frozen Foods
3          Regular           FDL50                 Canned
4          Low Fat           DRI25            Soft Drinks

   Outlet Establishment Year Outlet Identifier Outlet Location Type  \
0                       2012            OUT049               Tier 1
1                       2022            OUT018               Tier 3
2                       2010            OUT046               Tier 1
3                       2000            OUT013               Tier 3
4                       2015            OUT045               Tier 2

  Outlet Size        Outlet Type  Item Visibility  Item Weight     Sales  \
0      Medium  Supermarket Type1         0.100014        15.10  145.4786
1      Medium  Supermarket Type2         0.008596        11.80  115.3492
2       Small  Supermarket Type1         0.025896        13.85  165.0210
3        High  Supermarket Type1         0.042278        12.15  126.5046
4       Small  Supermarket Type1         0.033970        19.60   55.1614
```

```
   Rating
0     5.0
1     5.0
2     5.0
3     5.0
4     5.0
```

[4]:
```
df.tail()
```

[4]:
```
     Item Fat Content Item Identifier           Item Type  \
8518          low fat           NCT53  Health and Hygiene   
8519          low fat           FDN09         Snack Foods   
8520          low fat           DRE13         Soft Drinks   
8521              reg           FDT50               Dairy   
8522              reg           FDM58         Snack Foods   

      Outlet Establishment Year Outlet Identifier Outlet Location Type  \
8518                       1998            OUT027               Tier 3   
8519                       1998            OUT027               Tier 3   
8520                       1998            OUT027               Tier 3   
8521                       1998            OUT027               Tier 3   
8522                       1998            OUT027               Tier 3   

     Outlet Size        Outlet Type  Item Visibility  Item Weight     Sales  \
8518      Medium  Supermarket Type3         0.000000          NaN  164.5526   
8519      Medium  Supermarket Type3         0.034706          NaN  241.6828   
8520      Medium  Supermarket Type3         0.027571          NaN   86.6198   
8521      Medium  Supermarket Type3         0.107715          NaN   97.8752   
8522      Medium  Supermarket Type3         0.000000          NaN  112.2544   

      Rating
8518     4.0
8519     4.0
8520     4.0
8521     4.0
8522     4.0
```

[5]:
```
df.sample(10)
```

[5]:
```
     Item Fat Content Item Identifier              Item Type  \
7462          Low Fat           NCL31                 Others   
3972          Low Fat           FDF59          Starchy Foods   
2108          Regular           FDU20  Fruits and Vegetables   
289           Regular           FDW03                   Meat   
4183          Low Fat           DRI11            Hard Drinks   
3722          Low Fat           FDC20  Fruits and Vegetables   
6728          Low Fat           DRG36            Soft Drinks   
```

```
7733               LF           FDQ16           Frozen Foods
2938          low fat           FDJ45                Seafood
1632          Regular           FDC03                  Dairy

      Outlet Establishment Year Outlet Identifier Outlet Location Type  \
7462                       2017            OUT035               Tier 2
3972                       2020            OUT017               Tier 2
2108                       2012            OUT049               Tier 1
289                        2017            OUT035               Tier 2
4183                       1998            OUT019               Tier 1
3722                       2022            OUT018               Tier 3
6728                       2010            OUT046               Tier 1
7733                       2000            OUT013               Tier 3
2938                       2015            OUT045               Tier 2
1632                       2020            OUT017               Tier 2

     Outlet Size        Outlet Type  Item Visibility  Item Weight     Sales  \
7462       Small  Supermarket Type1         0.120258        7.390  141.5470
3972        High  Supermarket Type1         0.000000       12.500  127.1020
2108      Medium  Supermarket Type1         0.021491       19.350  122.1098
289        Small  Supermarket Type1         0.024537        5.630  106.1306
4183       Small      Grocery Store         0.060237          NaN  115.9834
3722      Medium  Supermarket Type2         0.024069       10.650   56.2272
6728       Small  Supermarket Type1         0.095378       14.150  170.5106
7733        High  Supermarket Type1         0.041704       19.700  109.1912
2938      Medium  Supermarket Type1         0.073560       17.750   33.9216
1632        High  Supermarket Type1         0.072253        8.575  196.5794

      Rating
7462     4.0
3972     3.9
2108     4.2
289      5.0
4183     3.8
3722     4.0
6728     4.0
7733     4.0
2938     4.1
1632     4.3
```

1.0.4 Shape of Data

[6]:
```
print("No of Rows and Columns:",df.shape)
```

```
No of Rows and Columns: (8523, 12)
```

1.0.5 Size of Data

```
[7]: print("Total number of records present: ",df.size)
```

```
Total number of records present:  102276
```

1.0.6 Field Info

```
[8]: df.columns
```

```
[8]: Index(['Item Fat Content', 'Item Identifier', 'Item Type',
            'Outlet Establishment Year', 'Outlet Identifier',
            'Outlet Location Type', 'Outlet Size', 'Outlet Type', 'Item Visibility',
            'Item Weight', 'Sales', 'Rating'],
           dtype='object')
```

1.0.7 Data Types

```
[9]: df.dtypes
```

```
[9]: Item Fat Content              object
     Item Identifier               object
     Item Type                     object
     Outlet Establishment Year      int64
     Outlet Identifier             object
     Outlet Location Type          object
     Outlet Size                   object
     Outlet Type                   object
     Item Visibility              float64
     Item Weight                  float64
     Sales                        float64
     Rating                       float64
     dtype: object
```

1.0.8 Data Cleaning

```
[10]: df['Item Fat Content'].unique()
```

```
[10]: array(['Regular', 'Low Fat', 'low fat', 'LF', 'reg'], dtype=object)
```

```
[11]: df['Item Fat Content']=df['Item Fat Content'].replace({'low fat':'Low Fat',
                                                            'LF':'Low Fat',
                                                             'reg': 'Regular' })
```

```
[12]: df['Item Fat Content'].unique()
```

```
[12]: array(['Regular', 'Low Fat'], dtype=object)
```

1.1 Business Requirements

1.1.1 KPI's Requirements

[13]:

```
#Total sales
total_sales=df['Sales'].sum()

#Average sales
average_sales=df['Sales'].mean()

#Number Of Items
no_of_items_sold=df['Sales'].count()

#Average Rating
average_rating=df['Rating'].mean()


#output
print(f"Total sales :${total_sales:,.0f}")
print(f"Average Sales :${average_sales:,.1f}")
print(f"Number Of Items Sold : {no_of_items_sold:,.1f}")
print(f"Average Rating : {average_rating:,.1f}")
```

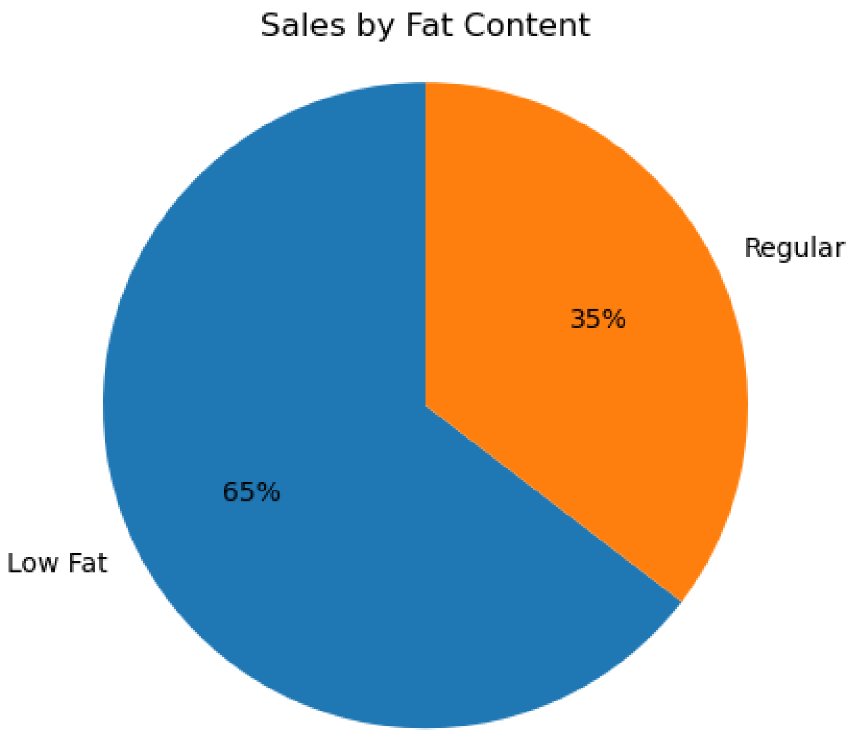
```
Total sales :$1,201,681
Average Sales :$141.0
Number Of Items Sold : 8,523.0
Average Rating : 4.0
```

1.1.2 Chart Requirements

Total Sales by Fats Content

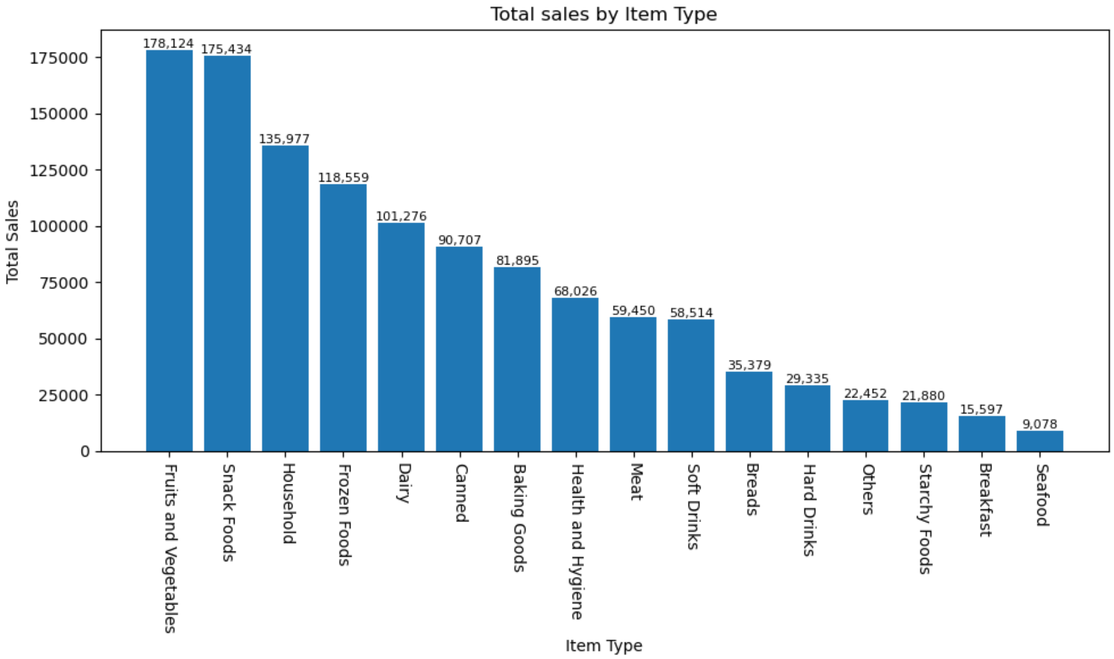
[14]:

```
sales_by_fat=df.groupby('Item Fat Content')['Sales'].sum()
plt.pie(sales_by_fat,labels=sales_by_fat.index,autopct='%1.f%%',startangle=90)
plt.title('Sales by Fat Content')
plt.axis('equal')
plt.show()
```



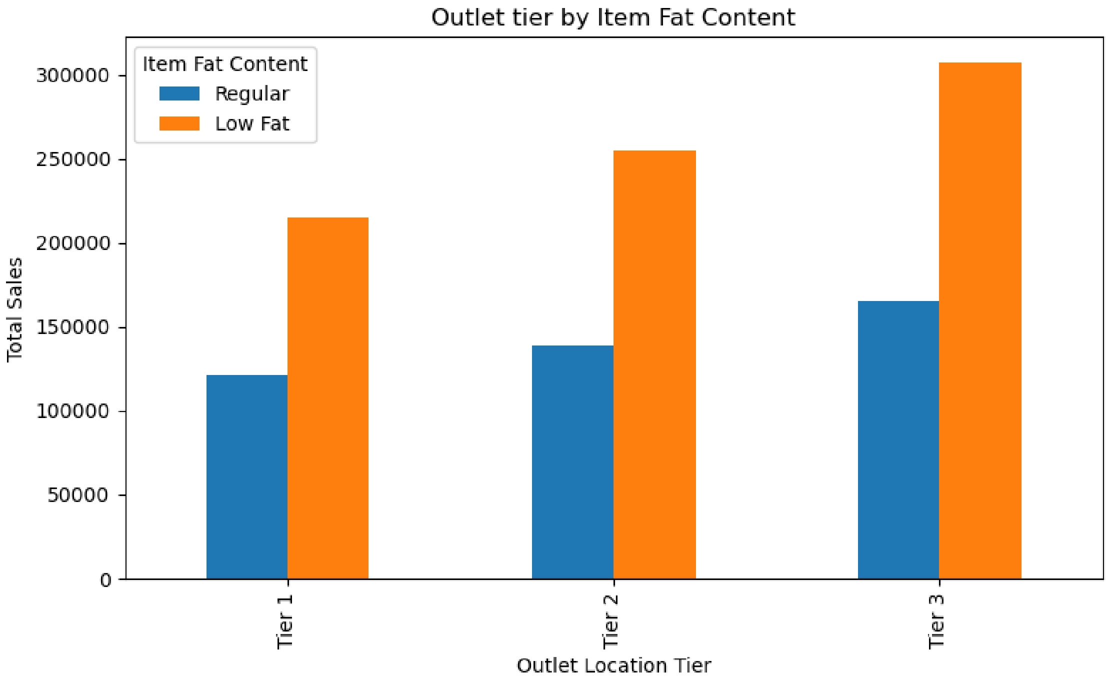
Total Sales by Item Type

[22]:
```
sales_by_type=df.groupby('Item Type')['Sales'].sum().sort_values(ascending=False)
plt.figure(figsize=(10,6))
bars=plt.bar(sales_by_type.index,sales_by_type.values)
plt.xticks(rotation=-90)
plt.xlabel('Item Type')
plt.ylabel('Total Sales')
plt.title('Total sales by Item Type')

for bar in bars:
    plt.text(bar.get_x()+bar.get_width()/2,bar.get_height(),f'{bar.get_height():,.0f}',ha='center',va='bottom',fontsize=8)


plt.tight_layout()
plt.show()
```



Fat Content by Outlet for Total Sales

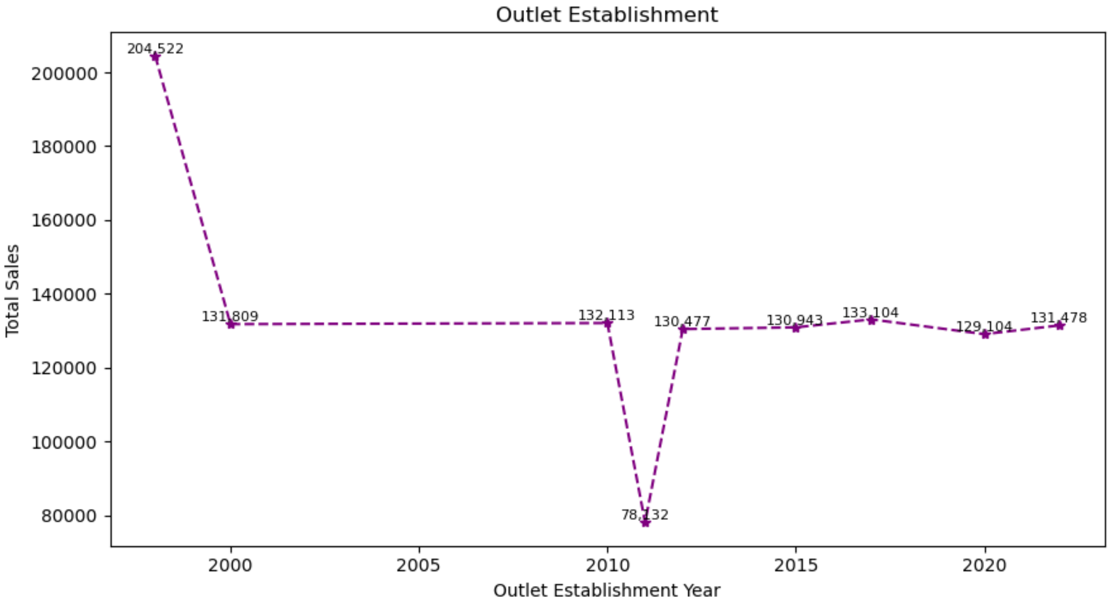
```
[26]: grouped = df.groupby(['Outlet Location Type','Item Fat Content'])['Sales'].
       ↪sum().unstack()
      grouped=grouped[['Regular','Low Fat']]
      ax=grouped.plot(kind='bar',figsize=(8,5),title='Outlet tier by Item Fat␣
       ↪Content')
      plt.xlabel('Outlet Location Tier')
      plt.ylabel('Total Sales')
      plt.legend(title='Item Fat Content')
      plt.tight_layout()
      plt.show()
```



Total Sales by Outlet Establishment

[28]:

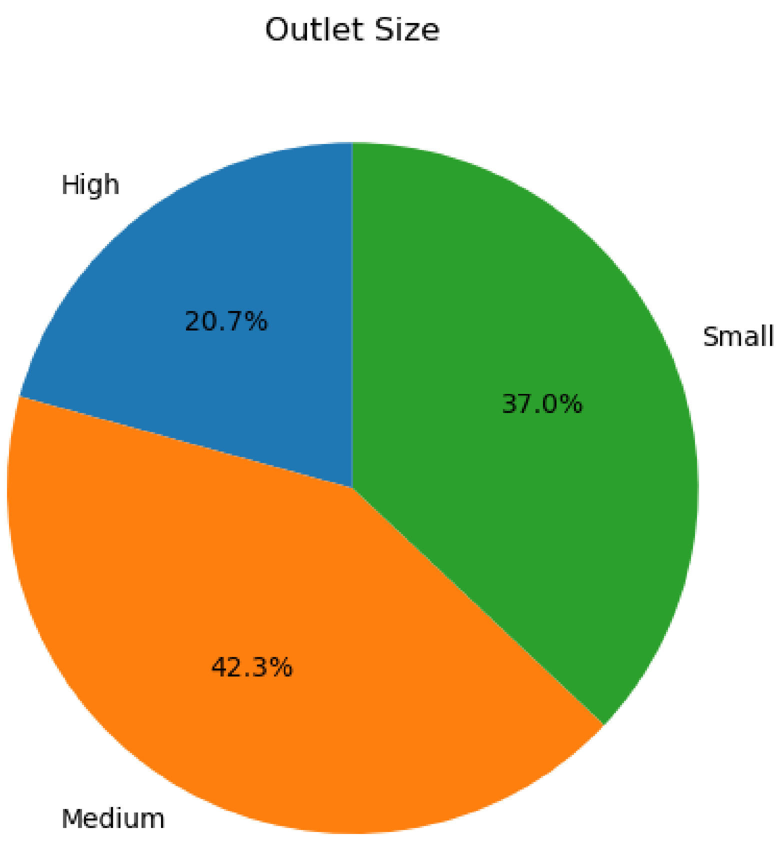
```
sales_by_year=df.groupby('Outlet Establishment Year')['Sales'].sum().
  ↪sort_index()
plt.figure(figsize=(9,5))
plt.plot(sales_by_year.index,sales_by_year.
  ↪values,marker='*',linestyle='--',color='purple')

plt.xlabel('Outlet Establishment Year')
plt.ylabel('Total Sales')
plt.title('Outlet Establishment')

for x,y in zip(sales_by_year.index,sales_by_year.values):
    plt.text(x,y,f'{y:,.0f}',ha='center',va='bottom',fontsize=8)

plt.tight_layout()
plt.show()
```



Sales by Outlet Size

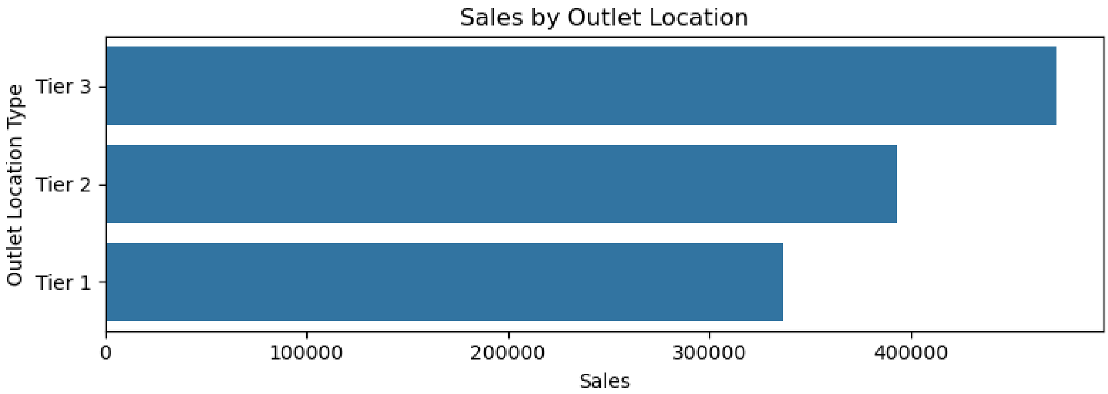
[31]:

```
sales_by_size=df.groupby('Outlet Size')['Sales'].sum()
plt.figure(figsize=(5,5))
plt.pie(sales_by_size,labels=sales_by_size.index,autopct='%.1f%%',startangle=90)
plt.title('Outlet Size')
plt.tight_layout()
plt.show()
```



Sales by Outlet Location

[33]:
```
sales_by_location=df.groupby('Outlet Location Type')['Sales'].sum().
  ↪reset_index()
sales_by_location=sales_by_location.sort_values('Sales',ascending=False)

plt.figure(figsize=(8,3))
sns.barplot(x='Sales',y='Outlet Location Type',data=sales_by_location)
plt.xlabel('Sales')
plt.ylabel('Outlet Location Type')
plt.title('Sales by Outlet Location')
plt.tight_layout()
plt.show()
```


Sales by Outlet Location
Outlet Location Type
Tier 3
Tier 2
Tier 1
0
100000
200000
300000
400000
Sales

[ ]: