In [2]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [6]:
```
train_data = pd.read_csv('train.csv')
```

In [18]:
```
print(train_data.head())
print(train_data.tail())
```

```
   id        date  store_nbr      family  sales  onpromotion
0   0  2013-01-01        1.0  AUTOMOTIVE    0.0          0.0
1   1  2013-01-01        1.0   BABY CARE    0.0          0.0
2   2  2013-01-01        1.0      BEAUTY    0.0          0.0
3   3  2013-01-01        1.0   BEVERAGES    0.0          0.0
4   4  2013-01-01        1.0       BOOKS    0.0          0.0
          id        date  store_nbr                family  sales  onpromotion
27504  27504  2013-01-16       30.0   HOME AND KITCHEN I    0.0          0.0
27505  27505  2013-01-16       30.0  HOME AND KITCHEN II    0.0          0.0
27506  27506  2013-01-16       30.0      HOME APPLIANCES    0.0          0.0
27507  27507  2013-01-16       30.0            HOME CARE    0.0          0.0
27508  27508     2013-01        NaN                  NaN    NaN          NaN
```

In [19]:
```
print(train_data.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 27509 entries, 0 to 27508
Data columns (total 6 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   id           27509 non-null  int64  
 1   date         27509 non-null  object 
 2   store_nbr    27508 non-null  float64
 3   family       27508 non-null  object 
 4   sales        27508 non-null  float64
 5   onpromotion  27508 non-null  float64
dtypes: float64(3), int64(1), object(2)
memory usage: 1.3+ MB
None
```

In [20]:
```
print(train_data.describe())
```

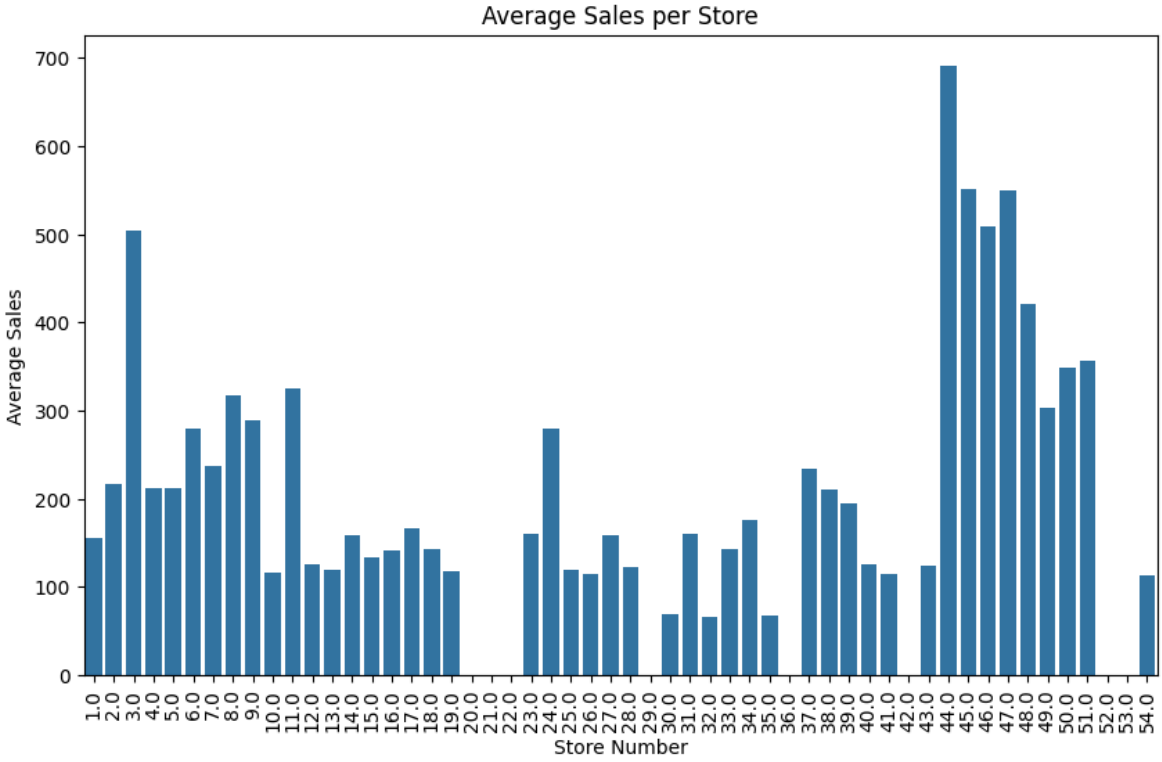
```
                 id     store_nbr         sales  onpromotion
count  27509.000000  27508.000000  27508.000000      27508.0
mean   13754.000000     27.218009    192.067328          0.0
std     7941.308614     15.513085    675.606573          0.0
min        0.000000      1.000000      0.000000          0.0
25%     6877.000000     14.000000      0.000000          0.0
50%    13754.000000     27.000000      0.000000          0.0
75%    20631.000000     41.000000     71.000000          0.0
max    27508.000000     54.000000  19849.000000          0.0
```

In [21]:
```
print(train_data.isnull().sum())
```

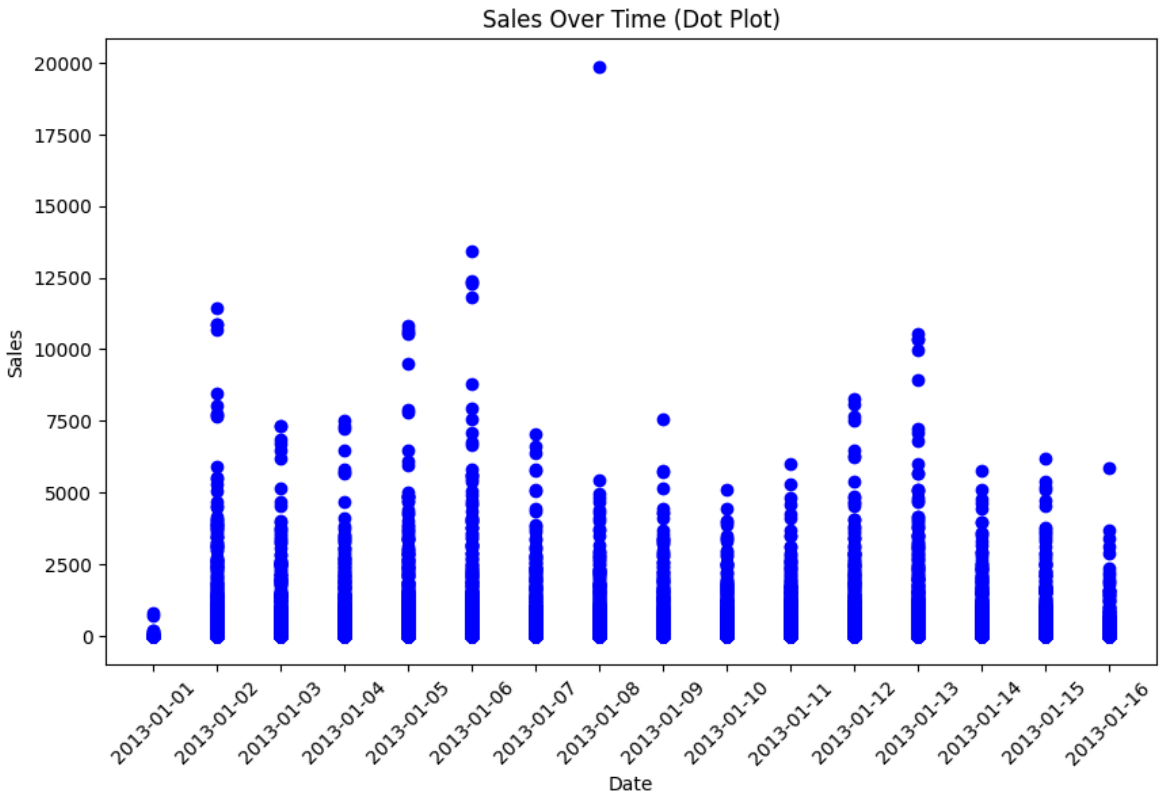
```
id              0
date            0
store_nbr       1
family          1
sales           1
onpromotion     1
dtype: int64
```

In [22]:
```
# Group by store_id and calculate the average sales
avg_sales_per_store = train_data.groupby('store_nbr')['sales'].mean()
```

In [23]:
```
# Create bar plot
plt.figure(figsize=(10, 6))
sns.barplot(x=avg_sales_per_store.index, y=avg_sales_per_store.values)
plt.title('Average Sales per Store')
plt.xlabel('Store Number')
plt.ylabel('Average Sales')
plt.xticks(rotation=90)
plt.show()
```



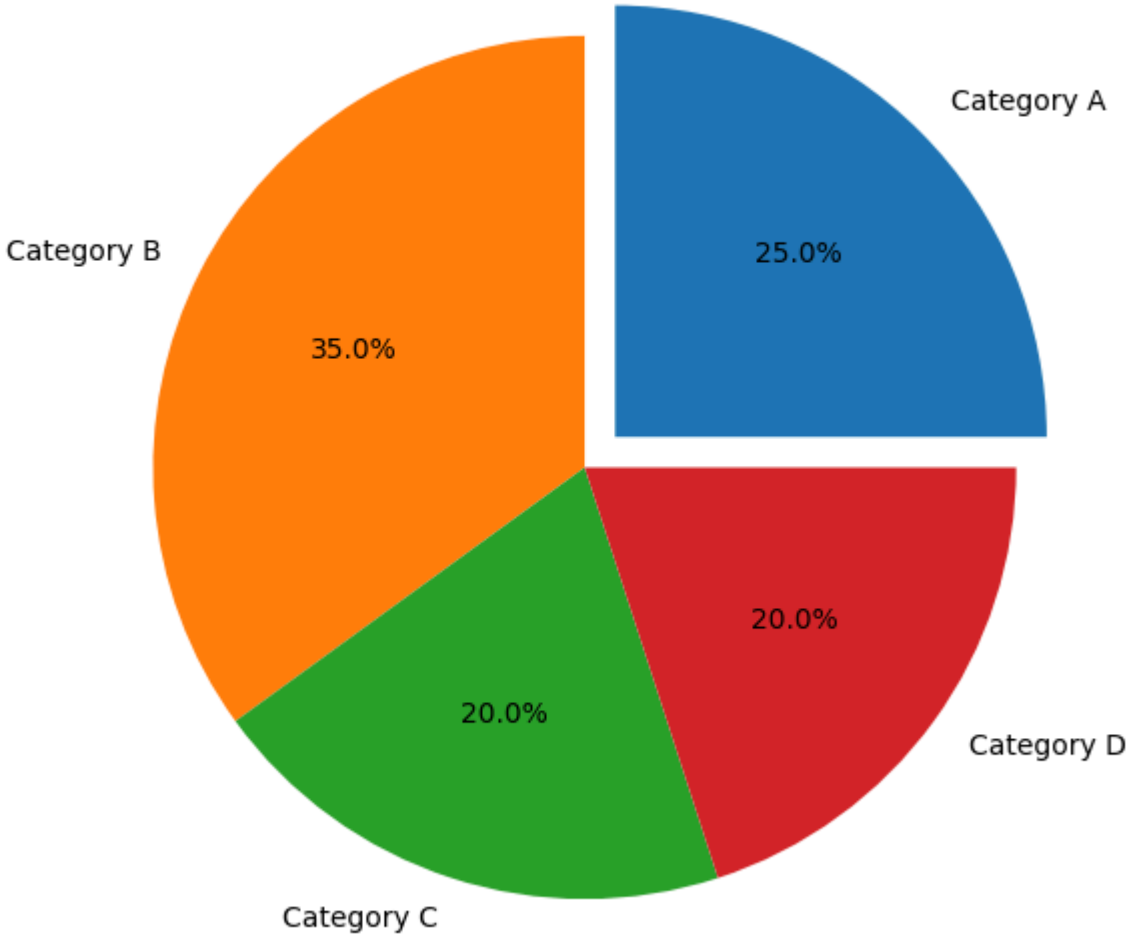
In [24]:
```
# Create dot plot
plt.figure(figsize=(10, 6))
plt.plot(train_data['date'], train_data['sales'], 'o', color='blue')
plt.title('Sales Over Time (Dot Plot)')
plt.xlabel('Date')
plt.ylabel('Sales')
plt.xticks(rotation=45)
plt.show()
```



In [25]:
```
# Group by store_id and count the transactions
transactions_per_store = train_data['store_nbr'].value_counts()
```

In [28]:
```
# Example pie chart with no overlap
labels = ['Category A', 'Category B', 'Category C', 'Category D']
sizes = [25, 35, 20, 20]

plt.figure(figsize=(7,7))  # Increase the figure size
plt.pie(sizes, labels=labels, autopct='%1.1f%%', explode=(0.1, 0, 0, 0))  # Expl
plt.show()
```



In [ ]: