```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt

from sklearn import preprocessing
from sklearn.model_selection import train_test_split
from xgboost import XGBRegressor
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.linear_model import LinearRegression

import seaborn as sn
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
from sklearn.svm import SVR
from sklearn.svm import SVC

from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeRegressor
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import RandomForestRegressor
from sklearn.neighbors import KNeighborsRegressor
from sklearn.neighbors import KNeighborsClassifier
from sklearn.datasets import load_iris
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import AdaBoostClassifier
from sklearn.ensemble import AdaBoostRegressor
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.ensemble import GradientBoostingRegressor
from xgboost import XGBRegressor
```

```
df=pd.read_excel('/content/drive/MyDrive/DA FILES/shopping trend excel.xlsx')
df.head()
```

	Customer ID	Age	Gender	Item Purchased	Category	Purchase Amount (USD)	Location	Size	
0	1	55	Male	Blouse	Clothing	53	Kentucky	L	
1	2	19	Male	Sweater	Clothing	64	Maine	L	M
2	3	50	Male	Jeans	Clothing	73	Massachusetts	S	M
3	4	21	Male	Sandals	Footwear	90	Rhode Island	M	M
4	5	45	Male	Blouse	Clothing	49	Oregon	M	Turc

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3900 entries, 0 to 3899
Data columns (total 19 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   Customer ID               3900 non-null   int64  
 1   Age                       3900 non-null   int64  
 2   Gender                    3900 non-null   object 
 3   Item Purchased            3900 non-null   object 
 4   Category                  3900 non-null   object 
 5   Purchase Amount (USD)     3900 non-null   int64  
 6   Location                  3900 non-null   object 
 7   Size                      3900 non-null   object 
 8   Color                     3900 non-null   object 
 9   Season                    3900 non-null   object 
 10  Review Rating             3900 non-null   float64
 11  Subscription Status       3900 non-null   object 
 12  Payment Method            3900 non-null   object 
 13  Shipping Type             3900 non-null   object 
 14  Discount Applied          3900 non-null   object 
 15  Promo Code Used           3900 non-null   object 
 16  Previous Purchases        3900 non-null   int64  
 17  Preferred Payment Method  3900 non-null   object 
 18  Frequency of Purchases    3900 non-null   object 
dtypes: float64(1), int64(4), object(14)
memory usage: 579.0+ KB
```

```
df.shape
```

```
(3900, 19)
```

```
df.dtypes
```

	0
Customer ID	int64
Age	int64
Gender	object
Item Purchased	object
Category	object
Purchase Amount (USD)	int64
Location	object
Size	object
Color	object
Season	object
Review Rating	float64
Subscription Status	object
Payment Method	object
Shipping Type	object
Discount Applied	object
Promo Code Used	object
Previous Purchases	int64
Preferred Payment Method	object
Frequency of Purchases	object

dtype: object

```
df.describe()
```

	Customer ID	Age	Purchase Amount (USD)	Review Rating	Previous Purchases
count	3900.000000	3900.000000	3900.000000	3900.000000	3900.000000
mean	1950.500000	44.068462	59.764359	3.749949	25.351538
std	1125.977353	15.207589	23.685392	0.716223	14.447125
min	1.000000	18.000000	20.000000	2.500000	1.000000
25%	975.750000	31.000000	39.000000	3.100000	13.000000
50%	1950.500000	44.000000	60.000000	3.700000	25.000000
75%	2925.250000	57.000000	81.000000	4.400000	38.000000
max	3900.000000	70.000000	100.000000	5.000000	50.000000

```
df.isnull().sum()
```

	0
Customer ID	0
Age	0
Gender	0
Item Purchased	0
Category	0
Purchase Amount (USD)	0
Location	0
Size	0
Color	0
Season	0
Review Rating	0
Subscription Status	0
Payment Method	0
Shipping Type	0
Discount Applied	0
Promo Code Used	0
Previous Purchases	0
Preferred Payment Method	0
Frequency of Purchases	0

dtype: int64

```
df.columns = df.columns.str.lower()
df.columns = df.columns.str.replace(' ', '_')
df =df.rename(columns={'purchase_amount_(usd)': 'purchase_amount'})
df.columns
```

```
Index(['customer_id', 'age', 'gender', 'item_purchased', 'category',
       'purchase_amount', 'location', 'size', 'color', 'season',
       'review_rating', 'subscription_status', 'payment_method',
       'shipping_type', 'discount_applied', 'promo_code_used',
       'previous_purchases', 'preferred_payment_method',
       'frequency_of_purchases'],
      dtype='object')
```

```
#creating a column age group
labels = ['Young Adult', 'Adult', 'Middle_aged', 'Senior']
df['age_group']=pd.qcut(df['age'], q=4, labels=labels)
df[['age','age_group']].head(10)
```

	age	age_group
0	55	Middle_aged
1	19	Young Adult
2	50	Middle_aged
3	21	Young Adult
4	45	Middle_aged
5	46	Middle_aged
6	63	Senior
7	27	Young Adult
8	26	Young Adult
9	57	Middle_aged

```
#creating a column of purchase_frequency_days

frequency_mapping = {
    'Fornightly': 14,
    'Weekly': 7,
    'Monthly': 30,
    'Quaterly': 90,
    'Bi-weekly': 14,
    'Annually': 365,
    'Every 3 Months': 90
}
df['purchase_frequency_days']=df['frequency_of_purchases'].map(frequency_mappir

df[['purchase_frequency_days','frequency_of_purchases']].head(10)
```

	purchase_frequency_days	frequency_of_purchases
0	NaN	Fortnightly
1	NaN	Fortnightly
2	7.0	Weekly
3	7.0	Weekly
4	365.0	Annually
5	7.0	Weekly
6	NaN	Quarterly
7	7.0	Weekly
8	365.0	Annually
9	NaN	Quarterly

```
df[['discount_applied','promo_code_used']].head(10)
```

	discount_applied	promo_code_used
0	Yes	Yes
1	Yes	Yes
2	Yes	Yes
3	Yes	Yes
4	Yes	Yes
5	Yes	Yes
6	Yes	Yes
7	Yes	Yes
8	Yes	Yes
9	Yes	Yes

```
(df['discount_applied'] == df['promo_code_used']).all()
```

```
np.True_
```

```
df=df.drop('promo_code_used',axis=1)
df.columns
```

```
Index(['customer_id', 'age', 'gender', 'item_purchased', 'category',
       'purchase_amount', 'location', 'size', 'color', 'season',
```

```
'review rating', 'subscription status', 'payment method',
```