In [3]:
```
import pandas as pd
import numpy as np
```

In [4]:
```
df=pd.read_csv("Advertising.csv")
```

In [5]:
```
df
```

Out[5]:

	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9
...	...	...	...	...
195	38.2	3.7	13.8	7.6
196	94.2	4.9	8.1	14.0
197	177.0	9.3	6.4	14.8
198	283.6	42.0	66.2	25.5
199	232.1	8.6	8.7	18.4

200 rows × 4 columns

In [6]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 4 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   TV         200 non-null    float64
 1   Radio      200 non-null    float64
 2   Newspaper  200 non-null    float64
 3   Sales      200 non-null    float64
dtypes: float64(4)
memory usage: 6.4 KB
```

Linear Regression

In [7]:
```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score, mean_squared_error
```

In [8]:
```
X = df['TV'].values.reshape(-1, 1)
y = df['Sales']
```

In [9]:
```
model = LinearRegression()
```

In [10]:
```
model.fit(X, y)
```

Out[10]:
```
LinearRegression()
```

In [16]:
```
y_pred = model.predict(X)
```

In [12]:
```
r_squared = r2_score(y, y_pred)
r_squared
```

Out[12]: 0.8121757029987414

Multi linear regression

In [13]:
```
X = df[['TV', 'Radio', 'Newspaper']]
y = df['Sales']
```

In [14]:
```
model = LinearRegression()
model.fit(X, y)
```

Out[14]:
```
LinearRegression()
```

In [15]:
```
y_pred = model.predict(X)
```

Knn

In [19]:
```
from sklearn.neighbors import KNeighborsRegressor
from sklearn.model_selection import train_test_split
```

In [20]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42
model = KNeighborsRegressor(n_neighbors=5)
```

In [21]:
```
model.fit(X_train, y_train)
```

Out[21]:
```
KNeighborsRegressor()
```

In [22]:
```
y_pred = model.predict(X_test)
```

Naive Bayes

In [26]:
```
from sklearn.naive_bayes import GaussianNB
```

In [27]:
```
X = df[['TV', 'Radio', 'Newspaper']]
y = df['Sales']
```

In [31]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42
```

In [32]:
```
model = GaussianNB()
```

In [ ]:
```
model.fit(X_train, y_train)
```

In [ ]:
```
y_pred = model.predict(X_test)
```

Gradient Descent

In [35]:
```
from sklearn.linear_model import SGDRegressor
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import mean_squared_error
```

In [36]:
```
X = df[['TV', 'Radio', 'Newspaper']]
y = df['Sales']
```

In [37]:
```
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

In [40]:
```
model = SGDRegressor(loss='squared_error', max_iter=1000, learning_rate='constant', eta0
```

In [41]:
```
model.fit(X_scaled, y)
```

Out[41]:
```
SGDRegressor
SGDRegressor(learning_rate='constant')
```

In [42]:
```
y_pred = model.predict(X_scaled)
```

In [ ]:
```
```