In [1]:
```
import numpy as np
import seaborn as sns
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import median_absolute_error
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import r2_score
from sklearn.ensemble import RandomForestRegressor
from sklearn.svm import SVR
```

In [2]:
```
data = pd.read_csv('advertising.csv')
data.head()
```

Out[2]:

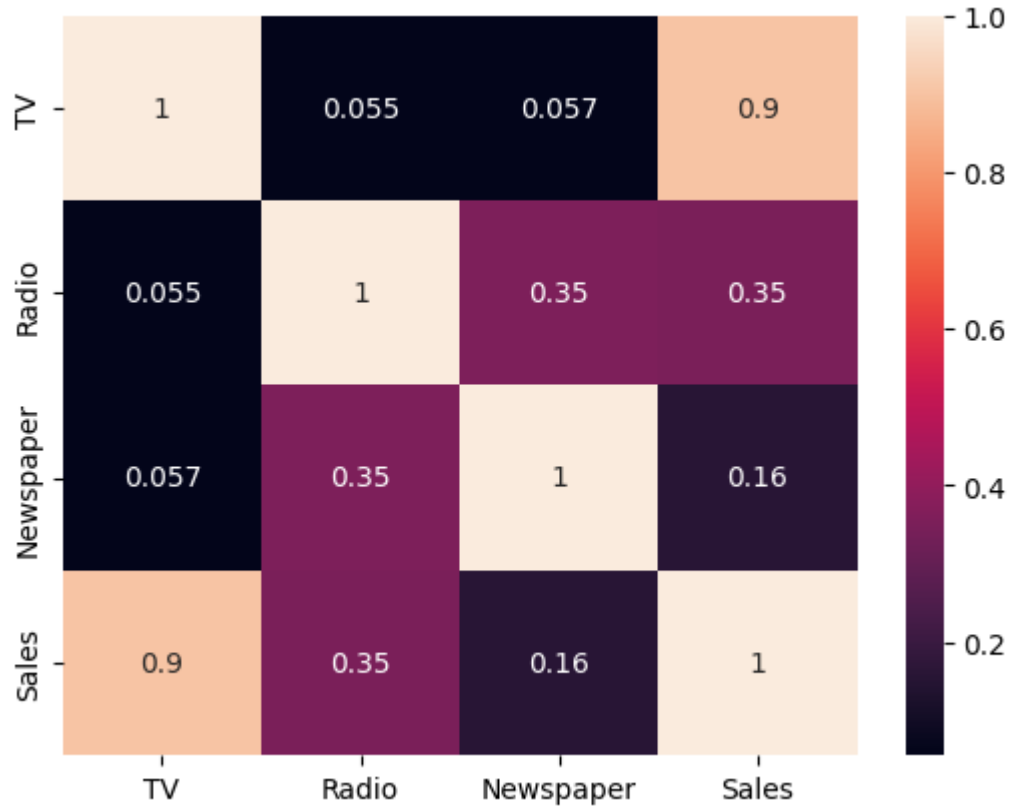
	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9

In [3]:
```
data.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
```

In [4]:
```
# finding correlation between features
sns.heatmap(data.corr(), annot=True)
```

Out[4]:
```
<Axes: >
```



In [5]:
```
data.isnull().sum()
```

Out[5]:
```
TV           0
Radio        0
Newspaper    0
Sales        0
dtype: int64
```

In [6]:
```
x = data.iloc[:, :-1].values
y = data.iloc[:, -1].values
```

In [9]:
```
x_train,x_test,y_train,y_test = train_test_split(x,y, test_size = 0.20, random_s
```

In [10]:
```
regressor = LinearRegression()
regressor.fit(x_train,y_train)
```

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

In [11]:
```
predicted = regressor.predict(x_test)
```

In [12]:
```
print(f"R2 score is {r2_score(y_test, predicted)*100:.2f}%")
print(f"Mean absolute error is {mean_absolute_error(y_test, predicted):.2f}")
print(f"Median absolute error is {median_absolute_error(y_test, predicted):.2f}"
```

```
R2 score is 86.45%
Mean absolute error is 1.42
Median absolute error is 0.58
```

In [13]:
```
rf_regressor = RandomForestRegressor(n_estimators=9000, random_state=0)
rf_regressor.fit(x, y)
```

```
y_pred = regressor.predict(x)
```

In [14]:
```
print(f"R2 score is {r2_score(y, y_pred)*100:.2f}%")
print(f"Mean absolute error is {mean_absolute_error(y, y_pred):.2f}")
print(f"Median absolute error is {median_absolute_error(y, y_pred):.2f}")
```

```
R2 score is 90.21%
Mean absolute error is 1.24
Median absolute error is 0.88
```

In [15]:
```
rf_regressor.feature_importances_
```

Out[15]:
```
array([0.85329491, 0.13453907, 0.01216602])
```

In [16]:
```
svr_regressor = SVR(kernel='rbf')
svr_regressor.fit(x,y)
```

Out[16]:
```
▾ SVR
SVR()
```

In [17]:
```
svr_predicted = svr_regressor.predict(x)
```

In [18]:
```
print(f"R2 score is {r2_score(y, svr_predicted)*100:.2f}%")
print(f"Mean absolute error is {mean_absolute_error(y, svr_predicted):.2f}")
print(f"Median absolute error is {median_absolute_error(y, svr_predicted):.2f}")
```

```
R2 score is 88.52%
Mean absolute error is 1.36
Median absolute error is 1.11
```