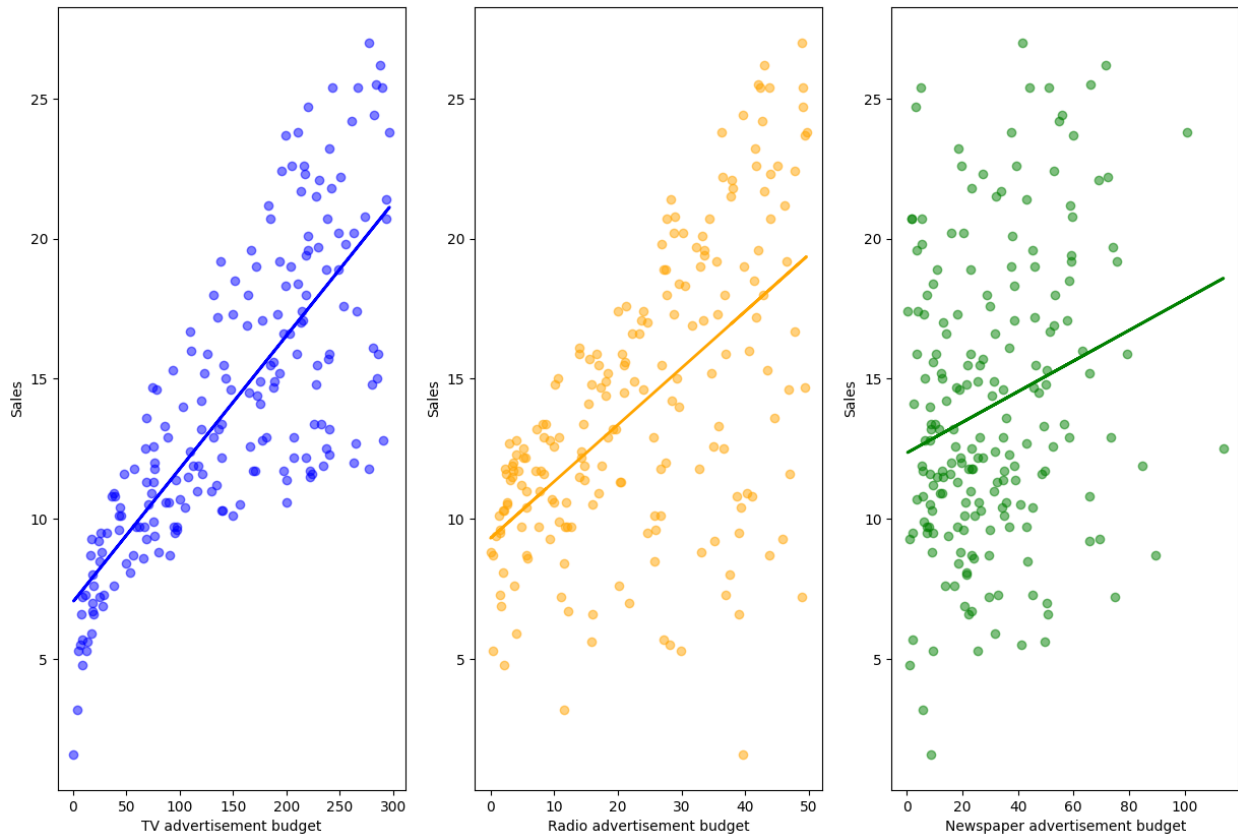


Assignment1

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

data = pd.read_csv('Advertising.csv')

X = data[['TV', 'radio', 'newspaper']]
y = data['sales']
fig = plt.figure(1, figsize=(15., 10.))
ax = plt.subplot(131)
plt.scatter(X['TV'], y, alpha = 0.5, color = 'blue')
plt.xlabel('TV advertisement budget')
plt.ylabel('Sales')
regr = LinearRegression()
regr.fit(X[['TV']], y)
plt.plot(X['TV'], regr.predict(X[['TV']]), color='blue', linewidth=2)
plt.subplot(132)
plt.scatter(X['radio'], y, alpha = 0.5, color = 'orange')
plt.xlabel('Radio advertisement budget')
plt.ylabel('Sales')
regr = LinearRegression()
regr.fit(X[['radio']], y)
plt.plot(X['radio'], regr.predict(X[['radio']]), color='orange',
linewidth=2)
plt.subplot(133)
plt.scatter(X['newspaper'], y, alpha = 0.5, color = 'green')
plt.xlabel('Newspaper advertisement budget')
plt.ylabel('Sales')
regr = LinearRegression()
regr.fit(X[['newspaper']], y)
plt.plot(X['newspaper'], regr.predict(X[['newspaper']]),
color='green', linewidth=2)
plt.savefig('Advertisement budget plots.png')
plt.show()
```



Assignment2

```
##Linear regression with all features
regr = LinearRegression()
regr.fit(X, y)
print('Intercept: ', regr.intercept_)
print('Coefficients: ', regr.coef_)
print('R-squared: ', regr.score(X, y))
print('Mean squared error: ', np.mean((regr.predict(X) - y) ** 2))
print('Root mean squared error: ', np.sqrt(np.mean((regr.predict(X) -
y) ** 2)))

Intercept:  2.938889369459412
Coefficients:  [ 0.04576465  0.18853002 -0.00103749]
R-squared:  0.8972106381789522
Mean squared error:  2.784126314510936
Root mean squared error:  1.6685701407225697

##Draw the Data Points (Y in x-axis and Predicted Y in y-axis) and the
y=x line
plt.figure(2,figsize=(15., 10.))
plt.scatter(y, regr.predict(X), alpha = 0.5)
plt.plot(y, y, color='red', linewidth=2)
plt.xlabel('Y')
plt.ylabel('Predicted Y')
```

```
plt.title('Data Points and y=x line')  
plt.savefig('Predicted Y vs Y_Y=X.png')  
plt.show()
```

