

## EDS ASSIGNMENT = 6

Group number = 9

230 ( Aditya Babar )

231 (Janhavi Kawadkar)

236 (Rohan Agrawal)

```
In [22]: import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

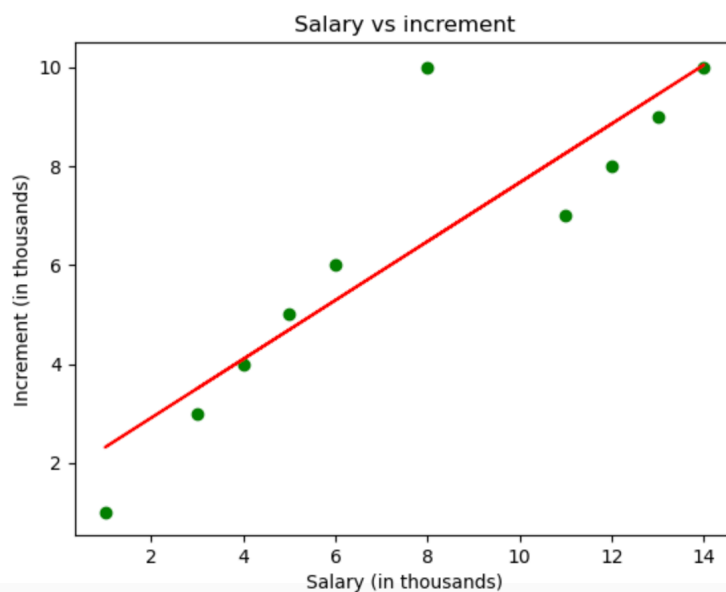
x = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]).reshape(-1, 1)
y = np.array([1, 2, 3, 4, 5, 6, 8, 10, 9, 7, 7, 8, 9, 10, 8])

# Splitting the dataset into training and test set.
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=1/3, random_state=0)

# Fitting the Simple Linear Regression model to the training dataset
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(x_train, y_train)

# Prediction of Test and Training set result
y_pred = regressor.predict(x_test)
x_pred = regressor.predict(x_train)

plt.scatter(x_train, y_train, color="green")
plt.plot(x_train, x_pred, color="red")
plt.title("Salary vs increment")
plt.xlabel("Salary (in thousands)")
plt.ylabel("Increment (in thousands)")
plt.show()
```



```
In [7]: from sklearn.neighbors import KNeighborsClassifier

# Sample dataset
X_train = [[100, 8], [150, 7], [200, 6], [250, 5], [300, 4], [350, 3]]
y_train = ['Circle', 'Circle', 'Triangle', 'Triangle', 'Triangle', 'Circle']

# Create a KNN classifier with k=3
knn = KNeighborsClassifier(n_neighbors=3)

# Train the classifier
knn.fit(X_train, y_train)

# New fruit to be classified
X_new = [[220, 7]]

# Predict the class of the new fruit
y_pred = knn.predict(X_new)

# Print the predicted class
print("Predicted class:", y_pred[0])

Predicted class: Triangle
```

---

```
In [2]: import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans

df = pd.read_csv("titanic.csv")
Data = {'x': df["Age"], 'y': df["Fare"]}
df = pd.DataFrame(Data, columns=['x', 'y'])

km = KMeans(n_clusters=3).fit(df)
centroids = km.cluster_centers_

plt.xlabel("Age")
plt.ylabel("Fare")
plt.scatter(df['x'], df['y'], c=km.labels_.astype(float), s=60, alpha=1)
plt.scatter(centroids[:, 0], centroids[:, 1], c='red', s=190)

plt.show()
```

