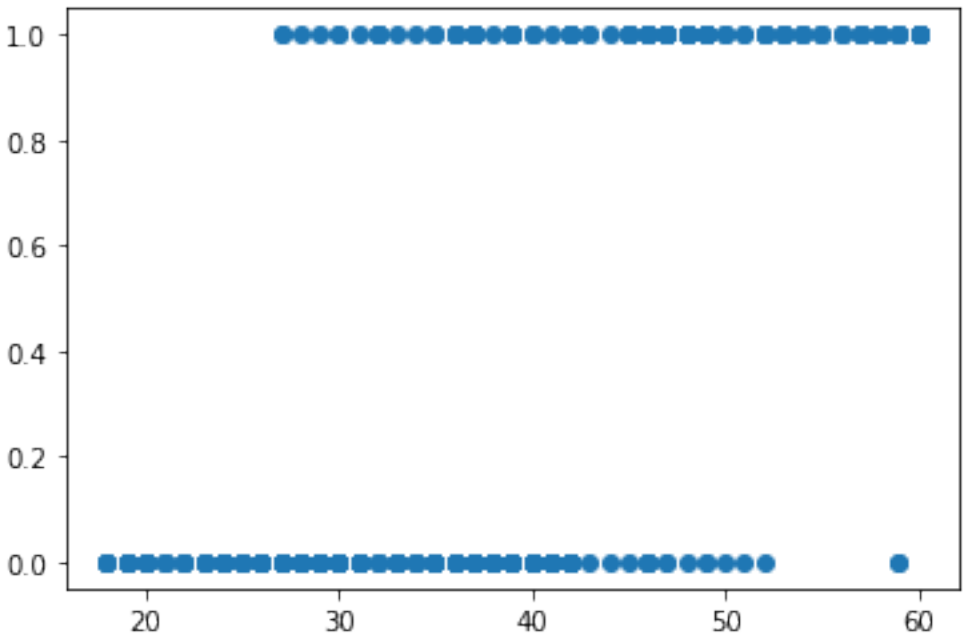
Lab 3 - Logistic regression 2

July 21, 2022

[1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from math import exp
plt.rcParams["figure.figsize"] = (10, 6)
```

[4]:
```
# Load the data
data = pd.read_csv("Social_Network_Ads.csv")
data.head()
#Visualizing the dataset
plt.scatter(data['Age'], data['Purchased'])
plt.show()
```



[5]:
```
# Divide the data to training set and test set
```

```
X_train, X_test, y_train, y_test = train_test_split(data['Age'],␣
↪data['Purchased'], test_size=0.20)
```

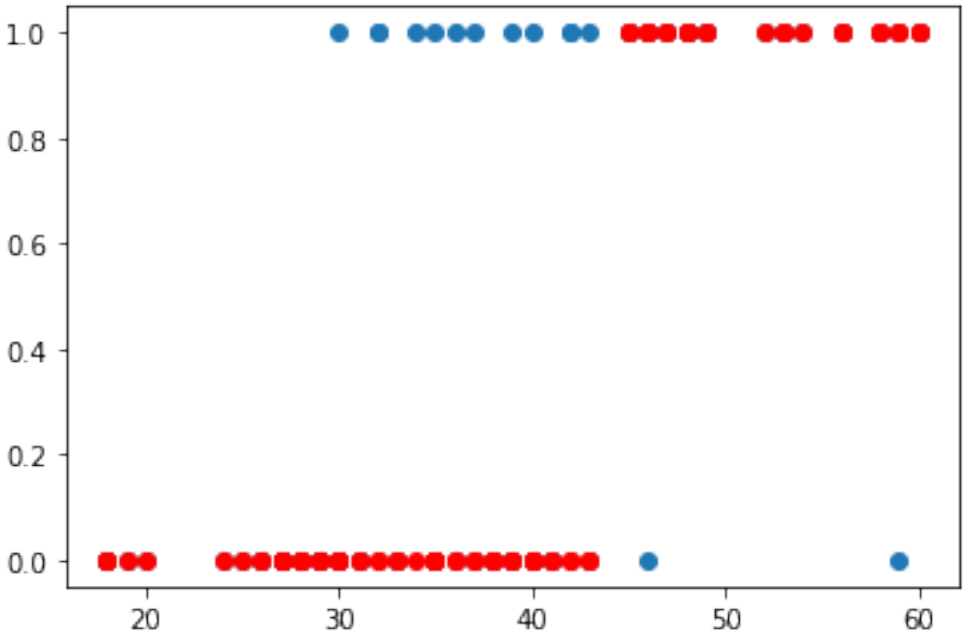
[6]:
```
# Helper function to normalize data
def normalize(X):
    return X - X.mean()

# Method to make predictions
def predict(X, b0, b1):
    return np.array([1 / (1 + exp(-1*b0 + -1*b1*x)) for x in X])

# Method to train the model
def logistic_regression(X, Y):

    X = normalize(X)

    # Initializing variables
    b0 = 0
    b1 = 0
    L = 0.001
    epochs = 300

    for epoch in range(epochs):
        y_pred = predict(X, b0, b1)
        D_b0 = -2 * sum((Y - y_pred) * y_pred * (1 - y_pred))  # Derivative of␣
↪loss wrt b0
        D_b1 = -2 * sum(X * (Y - y_pred) * y_pred * (1 - y_pred))  # Derivative␣
↪of loss wrt b1
        # Update b0 and b1
        b0 = b0 - L * D_b0
        b1 = b1 - L * D_b1

    return b0, b1
```

[7]:
```
# Training the model
b0, b1 = logistic_regression(X_train, y_train)
```

[8]:
```
# Making predictions
X_test_norm = normalize(X_test)
y_pred = predict(X_test_norm, b0, b1)
y_pred = [1 if p >= 0.5 else 0 for p in y_pred]
```

[9]:
```
plt.clf()
plt.scatter(X_test, y_test)
plt.scatter(X_test, y_pred, c="red")
plt.show()
```

```
# The accuracy
accuracy = 0
for i in range(len(y_pred)):
    if y_pred[i] == y_test.iloc[i]:
        accuracy += 1
print(f"Accuracy = {accuracy / len(y_pred)}")
```




```
Accuracy = 0.825
```

[ ]: