ENPM808A – Introduction to Machine Learning

Homework – 1

Author – Syed Mohammad Ibrahim

UMD ID: iamibi

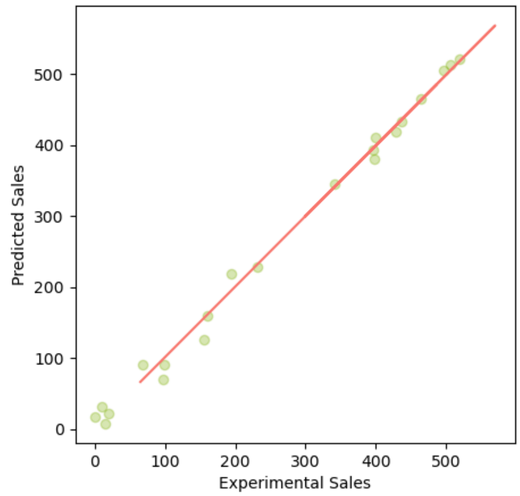
UID: 118428369

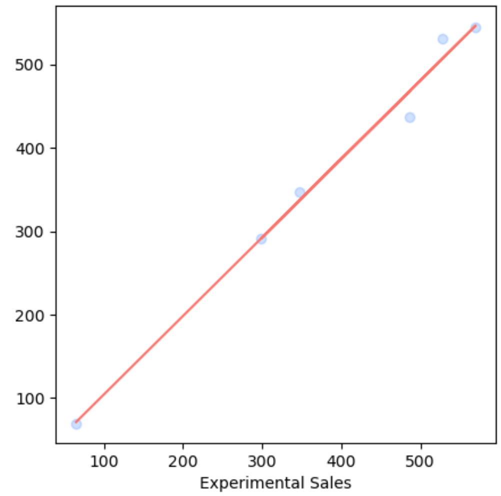
Email: iamibi@umd.edu

1.

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
from sklearn import linear_model
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.model_selection import train_test_split


df = pd.read_excel('mlr05.xls')

X = df.drop('X1', axis =1)
Y = df.X1

X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2)


model = linear_model.LinearRegression()
model.fit(X_train, Y_train)

Y_pred_train = model.predict(X_train)
Y_pred_test = model.predict(X_test)

full = linear_model.LinearRegression()
full.fit(X, Y)
full_pred = model.predict(X)

plt.figure(figsize=(5,11))

# 2 row, 1 column, plot 1
plt.subplot(2, 1, 1)
plt.scatter(x=Y_train, y=Y_pred_train, c="#7CAE00", alpha=0.3)

z = np.polyfit(Y_train, Y_pred_train, 1)
p = np.poly1d(z)
plt.plot(Y_test,p(Y_test),"#F8766D")

plt.ylabel('Predicted Sales')


# 2 row, 1 column, plot 2
```

```
plt.subplot(2, 1, 2)
plt.scatter(x=Y_test, y=Y_pred_test, c="#619CFF", alpha=0.3)

z = np.polyfit(Y_test, Y_pred_test, 1)
p = np.poly1d(z)
plt.plot(Y_test,p(Y_test),"#F8766D")

plt.ylabel('Predicted Sales')
plt.xlabel('Experimental Sales')

plt.savefig('plot_vertical_Sales.png')
plt.savefig('plot_vertical_Sales.pdf')
plt.show()

plt.figure(figsize=(11,5))

# 1 row, 2 column, plot 1
plt.subplot(1, 2, 1)
plt.scatter(x=Y_train, y=Y_pred_train, c="#7CAE00", alpha=0.3)

z = np.polyfit(Y_train, Y_pred_train, 1)
p = np.poly1d(z)
plt.plot(Y_test,p(Y_test),"#F8766D")

plt.ylabel('Predicted Sales')
plt.xlabel('Experimental Sales')

# 1 row, 2 column, plot 2
plt.subplot(1, 2, 2)
plt.scatter(x=Y_test, y=Y_pred_test, c="#619CFF", alpha=0.3)

z = np.polyfit(Y_test, Y_pred_test, 1)
p = np.poly1d(z)
plt.plot(Y_test,p(Y_test),"#F8766D")

plt.xlabel('Experimental Sales')

plt.savefig('plot_horizontal_Sales.png')
plt.savefig('plot_horizontal_Sales.pdf')
plt.show()
```





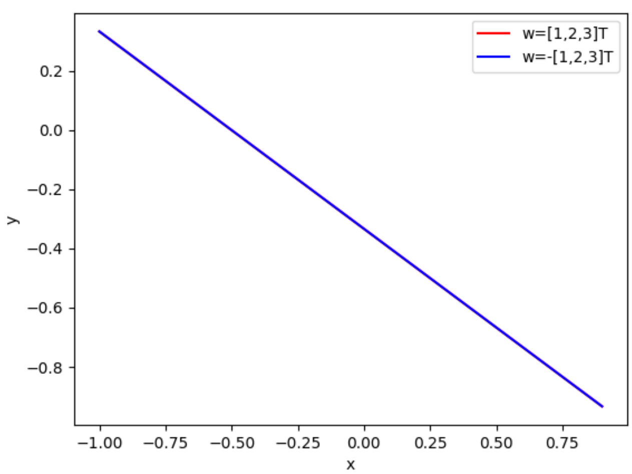
2.a. Using analytics geo., $w^T x = 0$. h(x) = +1 and h(x) = -1 are separated by a straight line because there are points on one side of the line $w^T x > 0$ and same for $w^T x < 0$. The slope *a* and intercept *b* can be expressed in w_0, w_1, w_2 as:

$$wTx \ = \ w0 \ + \ w1x1 \ + \ w2x2 \ = \ 0$$

when $w_2 \neq 0$

$$x2 = -\frac{w0}{w2} - \frac{w1}{w2}\,x$$

2.b.



3.

Q.3 → Let $w^* = \arg\min\{\|w\| : \langle w, x_i\rangle y_i \geqslant 1, \ \forall i \in [1:n]\}$

The idea of the proof is to show that after performing T iterations, the cosine of the angle between w^* and $w^{(T+1)}$ is at least $\frac{\sqrt{T}}{RB}$, i.e.,

$$\cos(\angle(w^*, w^{(T+1)})) = \frac{\langle w^*, w^{(T+1)}\rangle}{\|w^*\| \, \|w^{(T+1)}\|} \geqslant \frac{\sqrt{T}}{RB}$$

hence, $T \leqslant (RB)^2$

To show this, we first show that $\langle w^*, w^{(T+1)}\rangle \geqslant T$. Initially, we have $\underbrace{\langle w^*, w^{(1)}\rangle}_{=0}$

at iteration t, after updating $w^{(t+1)}$ based on example i we have

$$\begin{aligned}
\langle w^*, w^{(t+1)}\rangle - \langle w^*, w^{(t)}\rangle &= \langle w^*, w^{(t)} + y_i x_i\rangle - \langle w^*, w^{(t)}\rangle \\
&= \langle w^*, y_i x_i\rangle \\
&\geqslant 1
\end{aligned}$$

Therefore, after T iterations, we have

$$\langle w^*, w^{(T+1)} \rangle \geq T$$

Now we upper bound $\|w^{(T+1)}\|$. At each iteration t, we have

$$\|w^{(t+1)}\|^2 = \|w^{(t)} + y_i x_i\|^2$$
$$= \|w^{(t)}\|^2 + \|y_i x_i\|^2 + 2y_i \langle w^{(t)}, x_i \rangle$$
$$\leq \|w^{(t)}\|^2 + \|y_i x_i\|^2$$
$$\leq \|w^{(t)}\|^2 + R^2$$

Since $\|w^{(1)}\|^2 = 0$, after T iterations we have $\|w^{(T+1)}\|^2 \leq TR^2$

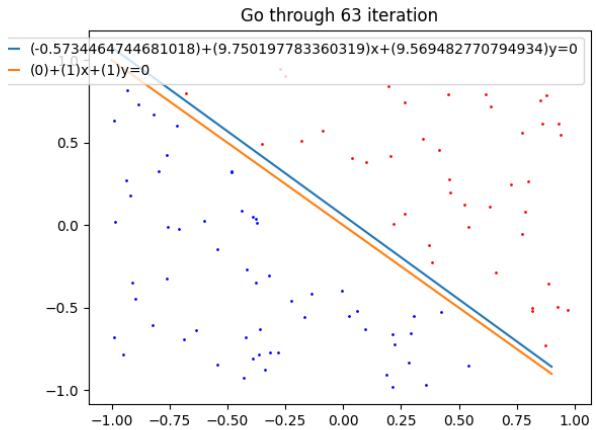
i.e., $\|w^{(T+1)}\| \leq \sqrt{T}R$

Combining the above we have

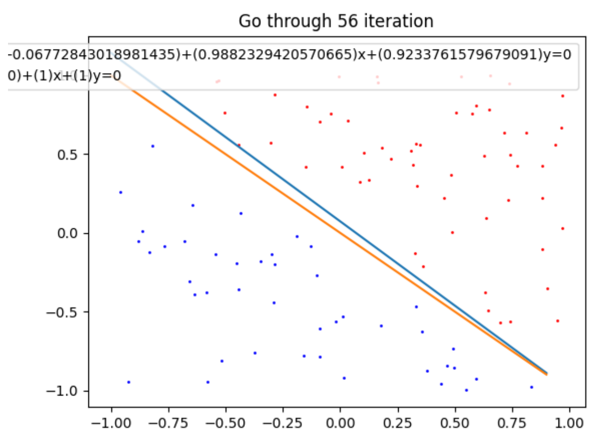
$$\frac{\langle w^*, w^{(T+1)} \rangle}{\|w^*\| \|w^{(T+1)}\|} \geq \frac{T}{B\sqrt{T}R}$$
$$= \frac{\sqrt{T}}{RB}$$

4. The question requires a consideration of the "deviation distance" when updating. In the process of trying, I found that taking 100 for η would cause the number to be too large, so here we take 1, 0.1, 0.01, 0.0001

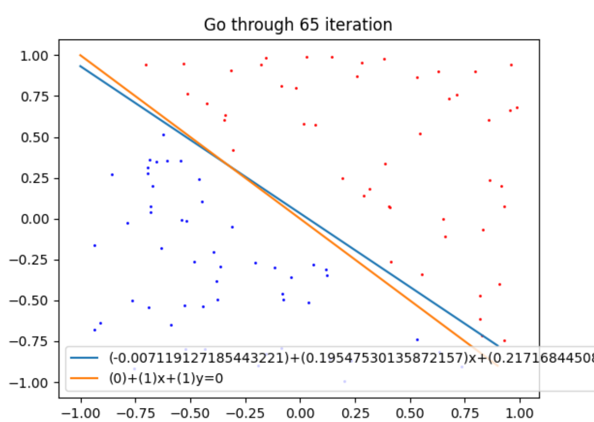
n = 1, time error rate = 0.0



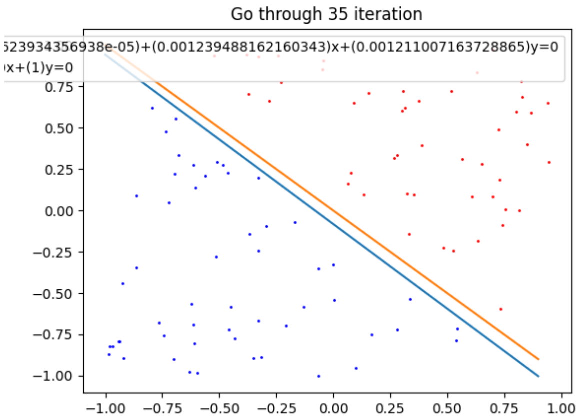
n = 0.1, time error rate = 0.0022



n = 0.01, time error rate = 0.002



n = 0.0001, time error rate = 0.0001



5.1. Using binomial distribution, there are two kinds of marbles in the box, 90% are red marbles, 10% are green marbles, now we take 10, and ask the probability that the number of red marbles is less than or equal to 1.

$$P = 0.1^{10} + C_{10}^{1} \times 0.1^{9} \times 0.9 \approx 9 \times 10^{-9} \qquad \text{where } P \text{ is the Probability}$$

5.2. Using Hoeffding Inequality:

$$\mathbb{P}\left[|\nu - \mu| > \epsilon\right] \leq 2e^{-2\epsilon^2 N} \qquad \text{for any } \epsilon > 0.$$

Substituting the values:

$$\mathbb{P}[v \leq 0.1] = \mathbb{P}[0.9 - v \geq 0.8] = \mathbb{P}[\mu - v \geq 0.8] \leq \mathbb{P}[|\mu - v| \geq 0.8] \leq 2e^{-2\times 0.8^2 \times 10} \approx 5.5215451440744015 \times 10^{-6}$$

This is a loose upper bound than the 5.1 (Exercise 1.8) as Hoeffding inequality is universal. Thus, the upper bound must be loose.