MLHW2

yz5946 Sky Zhou

March 2024

1 Problem 1

a

We iterate the linear model over all columns individually

```
#define best error and index of best feature
best_error = float('inf')
best_feature = -1
#iterate
for i in range(Xtr.shape[1]):
    model = LinearRegression()
    model.fit(Xtr[:,i:i+1]), ytr)
    #compute error
    mse = mean_squared_error(yts, model.predict(Xts[:,i:i+1]))
    #if less than best, update parameters
    if mse < best_error:
        best_error = mse
        best_feature = i
return best_feature
```

b

Similarly, we iterate the linear model with all pairs of features.

```
#define best error and index of best feature
best_error = float('inf')
best_feature = (-1,-1)
#iterate
for i in range(Xtr.shape[1]):
    for j in range(i+1, Xtr.shape[1]):
        model = LinearRegression()
        model.fit(Xtr[:,[i,j]]), ytr)
        #compute error
        mse = mean_squared_error(yts, model.predict(Xts[:,[i,j]]))
        #if less than best, update parameters
        if mse < best_error:
            best_error = mse
            best_feature = (i,j)
return best_feature
```

c

This requires to iterate over the number of different combinations of k features among p features. Therefore, we use combinatorics formula $\binom{p}{k} = \frac{p!}{(p-k)!k!}$
for $k = 10$ and $p = 1000$, we need

$$\frac{1000 * 999 * 998 * 997 * 996 * 995 * 994 * 993 * 992 * 991}{10 * 9 * 8 * 7 * 6 * 5 * 4 * 3 * 2 * 1} \approx 2.63 * 10^{20}$$

iterations.

2 Problem 2

a

Given the chart above, I estimate that the probability is

$$\frac{num_{spam}}{num_{emails}} = \frac{4}{6} \approx 0.667$$

b

In Bernoulli Naive Bayes, we compute conditional probability

$$P(\text{"}password\text{"}|spam) = \frac{P(\text{"}password\text{"} and spam)}{P(spam)} = \frac{\frac{2}{6}}{\frac{2}{3}} = \frac{1}{2}$$

c

In this problem there are 4 words in the email to be evaluated. We first compute the probability of the email being spam with Laplace smoothing. Then compute the probability of the email being ham, and finally normalize these two probabilities to get the normalized result.

c.1 This computation does not involve the absence of a word, and this follows the example in class

For the word "send":

$$P('send'|spam) = \frac{3+1}{4+6} = \frac{2}{5}$$
$$P('send'|ham) = \frac{1+1}{2+6} = \frac{1}{4}$$

For the word "us":

$$P('us'|spam) = \frac{3+1}{4+6} = \frac{2}{5}$$

$$P('us'|ham) = \frac{1+1}{2+6} = \frac{1}{4}$$

For the word "your":

$$P('your'|spam) = \frac{3+1}{4+6} = \frac{2}{5}$$

$$P('your'|ham) = \frac{2+1}{2+6} = \frac{3}{8}$$

For the word "account":

$$P('account'|spam) = \frac{1+1}{4+6} = \frac{1}{5}$$

$$P('account'|ham) = \frac{0+1}{2+6} = \frac{1}{8}$$

So, $P(spam|\text{'send us your account'}) = P('send'|spam) * P('us'|spam) * P('your'|spam) *$ $P('account'|spam) * P(spam) = \frac{2*2*2*1}{5*5*5*5} * \frac{2}{3} \approx 0.00853$

$P(ham|\text{'send us your account'}) = P('send'|ham) * P('us'|ham) * P('your'|ham) * P('account'|ham) *$ $P(ham) = \frac{1*1*3*1}{4*4*8*8} * \frac{1}{3} \approx 0.0009765625$

Since we need $P(spam|\text{'send us your account'}) + P(ham|\text{'send us your account'}) = 1$, we normalize these two values.

$P(spam|\text{'send us your account'}) = \frac{0.00853}{0.00853+0.0009765625} \approx 0.8974$

$P(ham|\text{'send us your account'}) = \frac{0.0009765625}{0.00853+0.0009765625} \approx 0.1026$

c.2 This version involve the absence of a word, which is standard Bernoulli Naive Bayes computation

For simplicity, I only compute the absence of the word "review" and "password"

For the word "review":

$$P(\text{'review' not in spam}|spam) = 1 - P('review'|spam) = 1 - \frac{1+1}{4+6} = \frac{4}{5}$$

$$P(\text{'review' not in ham}|ham) = 1 - P('review'|ham) = 1 - \frac{2+1}{2+6} = \frac{5}{8}$$

For the word "password":

$$P(\text{'password' not in spam}|spam) = 1 - P('password'|ham) = 1 - \frac{2+1}{4+6} = \frac{7}{10}$$

$$P(\text{'password' not in ham}|ham) = 1 - P('password'|ham) = 1 - \frac{1+1}{2+6} = \frac{3}{4}$$

So, $P(spam|\text{'send us your account'}) = P('send'|spam) * P('us'|spam) * P('your'|spam) * P('account'|spam) * P(\text{'review' not in spam}|spam) * P(\text{'password' not in spam}|spam) * P(spam) = \frac{2*2*2*1*4*7}{5*5*5*5*5*10} * \frac{2}{3} \approx 0.00477867$

$P(ham|\text{'send us your account'}) = P('send'|ham)*P('us'|ham)*P('your'|ham)*P('account'|ham)* P(\text{'review' not in ham}|ham)*P(\text{'password' not in ham}|ham)*P(ham) = \frac{1*1*3*1*5*3}{4*4*8*8*8*4}*\frac{1}{3} \approx 0.00045776$

Since we need $P(spam|\text{'send us your account'}) + P(ham|\text{'send us your account'}) = 1$, we normalize these two values.

$P(spam|\text{'send us your account'}) = \frac{0.00477867}{0.00477867+0.00045776} \approx 0.9125$

$P(ham|\text{'send us your account'}) = \frac{0.00045776}{0.00477867+0.00045776} \approx 0.0875$

d

In this problem there are 3 words in the email to be evaluated. We first compute the probability of the email being spam with Laplace smoothing. Then compute the probability of the email being ham, and finally normalize these two probabilities to get the normalized result.

For the word "review":

$$P('review'|spam) = \frac{1+1}{13+6} = \frac{2}{19}$$

$$P('review'|ham) = \frac{1+1}{7+6} = \frac{2}{13}$$

For the word "your":

$$P('your'|spam) = \frac{3+1}{13+6} = \frac{4}{19}$$

$$P('your'|ham) = \frac{2+1}{7+6} = \frac{3}{13}$$

For the word "account":

$$P('account'|spam) = \frac{1+1}{13+6} = \frac{2}{19}$$

$$P('account'|ham) = \frac{0+1}{7+6} = \frac{1}{13}$$

So, $P(spam|\text{'review your account'}) = P('review'|spam)^1 * P('your'|spam)^1 * P('account'|spam)^1 * P(spam) = \frac{2*4*2}{19^3} * \frac{2}{3} \approx 0.001555$

$P(ham|\text{'review your account'}) = P('review'|ham)^1 * P('your'|ham)^1 * P('account'|ham)^1 * P(ham) = \frac{2*3*1}{13^3} * \frac{1}{3} \approx 0.0009105$

Since we need $P(spam|\text{'review your account'}) + P(ham|\text{'review your account'}) = 1$, we normalize these two values.

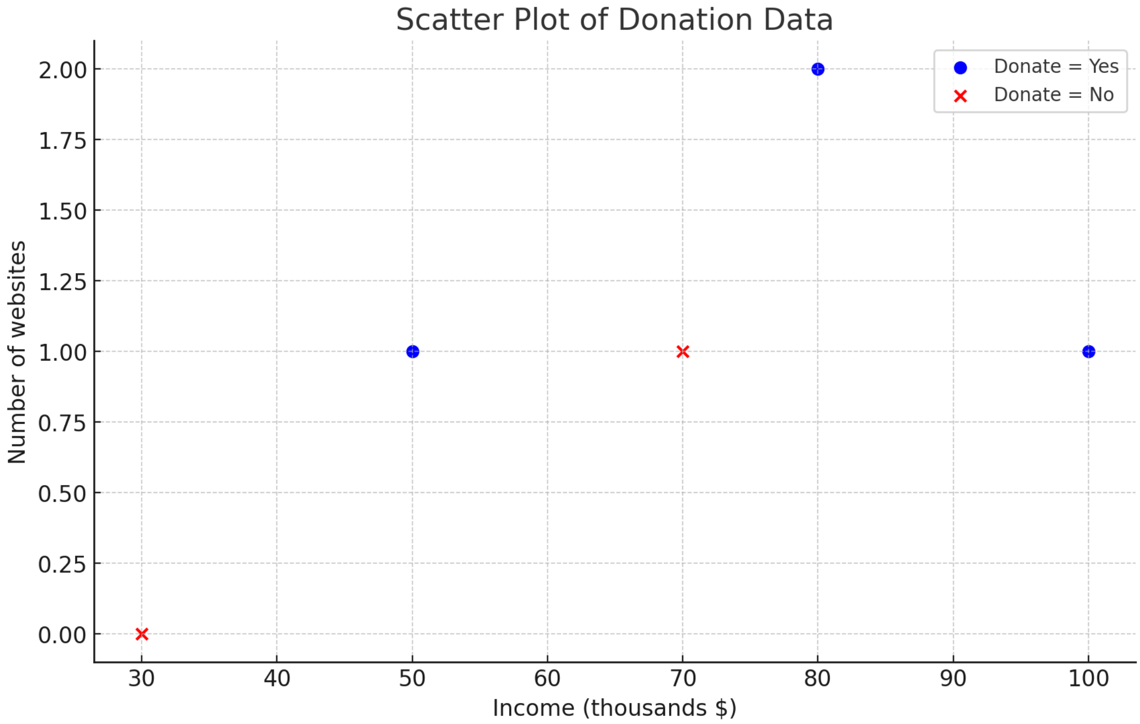
$P(spam|\text{'review your account'}) = \frac{0.001555}{0.001555+0.0009105} \approx 0.6308$

$P(ham|\text{'review your account'}) = \frac{0.0009105}{0.001555+0.0009105} \approx 0.3692$

3 Problem 3

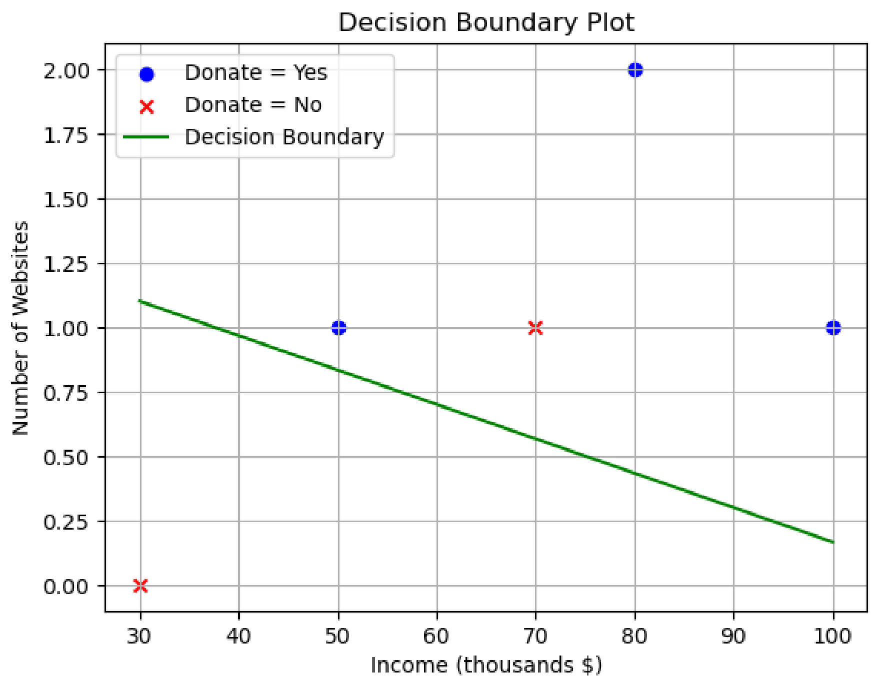
(See attached notebook HW2.ipynb for implementation)

a



b

I tried for different combinations of values for W and b. Intuitively, the coefficient for both income and websites should be positive since both of these factors are positively contributing to the result of donation or not.



For this decision boundary, the value of w is $[0.4, 30]^T$ and the value of b is -45

Above the decision line it would be classified as YES and vice versa. There is only one misclassified data point with regards to this decision line. Also, this line indicates that people with more income and more websites followed are more willing to donate.

c

The larger the value of z_i, the smaller the value of e^{-z_i}, then the larger the value of $P(y_i|x_i)$. Therefore, the least likely one should have the smallest value in z_i, which should be the smallest value of $w^T * x_i$.
We can compute the values of $W^T * x_i$ by hand for each data point.

$$z_1 = 0.4 * 30 + (30) * 0 + b \quad (1)$$
$$z_2 = 0.4 * 50 + (30) * 1 + b \quad (2)$$
$$z_3 = 0.4 * 70 + (30) * 0 + b \quad (3)$$
$$z_4 = 0.4 * 80 + (30) * 2 + b \quad (4)$$
$$z_5 = 0.4 * 100 + (30) * 1 + b \quad (5)$$

Therefore, the least likely one to donate is z_1

d

No, multiplying the same positive scalar to both w and b would not change the prediction result $\hat{y}$. This is because that $z_{i\alpha} = \alpha * (w^T x_i + b) = \alpha * z_i$. Since $\alpha > 0$, it will not change the sign of z_i. Therefore the decision threshold remains the same. Similarly, the actual value of likelihood will be affected but the classification result will not. Since in logistic model, the updated likelihood is

$$P(y_i = 1|x_i) = \frac{1}{1 + e^{-\alpha * z_i}}$$

But correspondingly, $P(y_i = 0|x_i)$ changes to the same proportion. And the change for both value will not change their ordering. Therefore the multiplication of α only changes the actual numerical value of likelihood but it won't affect the result of classification.