NTU CSIE FAI 2024 Homework 3

Hand-written Part

For the hand-written part, I discussed with 金京(B10204022), she inspired me a lot when I was struggling with mathematical problems.

1、

$$\nabla_w \, err(w^T x, y) = -2xy\,(\max(1-yw^Tx, 0))^2$$

$$\nabla E_{in}(w) = -\frac{2}{N}\sum_{n=1}^{N} x_n y_n \,(\max(1-y_n w^T x_n, 0))$$

2、

$$P_u(x_n) = \frac{1}{\sqrt{2\pi I}} e^{-\frac{1}{2}\frac{(x_n-u)^2}{I}}$$

$$u^* = \underset{u\in R^d}{\operatorname{argmax}} \prod_{n=1}^{N} P_u(x_n), \quad \text{argmax} \Rightarrow \text{derivative} = 0.$$

Since we want to find "u", and log won't affect their scale relationship

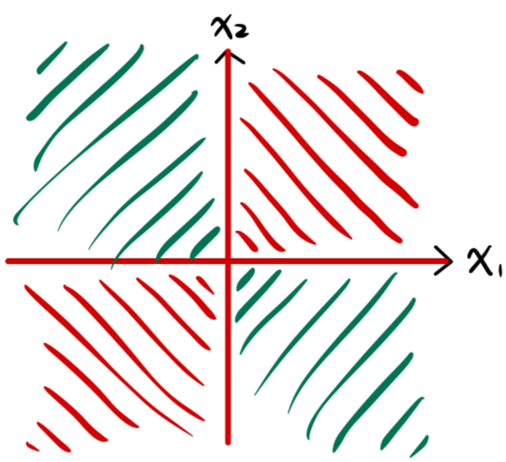
⇒ we can use ln to make it easier to calculate.

$$\ln\left(\prod_{n=1}^{N} P_u(x_n)\right) = -\sum_{n=1}^{N}\left(\frac{1}{2}\frac{(x_n-u)^2}{I} + \ln(\sqrt{2\pi I})\right)$$

$$\nabla_u \ln\left(\prod_{n=1}^{N} P_u(x_n)\right) = \sum_{n=1}^{N}\frac{x_n-u}{I} = 0 \Rightarrow u^* = \frac{1}{N}\sum_{n=1}^{N} x_n$$

3、

$$\tilde{W} = [0, 0, 0, 0, 1, 0]$$



4、

$$\frac{\sum_{n=1}^{N} W_n^{t+1}\,\delta(g_t(x_n), y_n)}{\sum_{n=1}^{N} W_n^{t+1}} = \frac{\overbrace{\sum_{wrong} W_n^t\sqrt{(1-\epsilon_t)/\epsilon_t}}^{\epsilon_t\cdot\sum_{n=1}^{N} W_n^t}\;\overbrace{\delta(g_t(x_n), y_n)}^{1}}{\underbrace{\sum_{wrong} W_n^t\sqrt{(1-\epsilon_t)/\epsilon_t}}_{\epsilon_t\cdot\sum_{n=1}^{N} W_n^t} + \underbrace{\sum_{correct} W_n^t/\sqrt{(1-\epsilon_t)/\epsilon_t}}_{(1-\epsilon_t)\sum_{n=1}^{N} W_n^t}} + \frac{\sum_{correct} W_n^t/\sqrt{(1-\epsilon_t)/\epsilon_t}\;\overbrace{\delta(g_t(x_n), y_n)}^{0}}{\sum_{wrong} W_n^t\sqrt{(1-\epsilon_t)/\epsilon_t} + \sum_{correct} W_n^t/\sqrt{(1-\epsilon_t)/\epsilon_t}}$$

$$= \frac{\sum_{n=1}^{N} W_n^t\sqrt{\epsilon_t(1-\epsilon_t)}}{\sum_{n=1}^{N} W_n^t\sqrt{\epsilon_t(1-\epsilon_t)} + \sum_{n=1}^{N} W_n^t\sqrt{\epsilon_t(1-\epsilon_t)}} = 0.5$$

Programming Part

For the programming part, I used ChatGPT and Copilot as my assistant.

1. Source code (Python)

see hw3.py.

2. Report

(a)

```
Logistic Regression Accuracy: 0.6444444444444445
Decision Tree Classifier Accuracy: 0.8888888888888888
Random Forest Classifier Accuracy: 0.9111111111111111
Linear Regression MSE: 43.55517372516372
Decision Tree Regressor MSE: 28.341570306709183
Random Forest Regressor MSE: 23.4852529856091
```

In both classification and regression, the performance of random forest is the best, which is same as my prior speculation. The rationale behind my speculation is

1. Linear models may not be powerful enough for the task due to their inability to capture non-linear relationships.
2. Decision tree is prone to overfitting.

In most cases, random forest performs the best among these models. This is because random forest leverages bagging and majority voting to prevent overfitting and enhance both performance and robustness.

(b)

Normalization:
Accuracy: 0.6444444444444445
Rationale: Use the minimum and maximum values of the features to rescale the data evenly.

Advantages:

1. Rescale the data to a fixed range, which is friendly to algorithms that are sensitive to the scale of the data.
2. Maintains the original shape of the data distribution better.

Disadvantages:

1. Sensitive to outliers.
2. May not perform well with algorithms that assume data in normal distribution.

Standardization
Accuracy: 0.8666666666666667
Rationale: Rescales the features to have a mean of 0 and a standard deviation of 1.

Advantages:

1. Less sensitive to outliers.
2. Perform well with algorithms that assume data in normal distribution.

Disadvantages:

1. Transformed data does not have a fixed range.
2. May not perform well if the data is not approximately normally distributed.

Which strategy is better depends on the dataset.

(c)

Config 1: learning rate: 0.01, iterations: 1000

```
Logistic Regression Accuracy: 0.6444444444444445
Linear Regression MSE: 43.55517372516372
```

Config 2: learning rate: 0.03, iterations: 1000

```
Logistic Regression Accuracy: 0.7555555555555555
Linear Regression MSE: 41.20507163976007
```

Config 3: learning rate: 0.01, iterations: 3000

```
Logistic Regression Accuracy: 0.7555555555555555
Linear Regression MSE: 41.20274641952791
```

Learning Rate
The performances of Config 2 are better than Config 1 in both classification and regression.
Higher learning rate will let the model converge faster, and therefore provide a better performance.
Iteration
The performances of Config 3 are better than Config 1 in both classification and regression.
More Iterations will give the model a higher chance to converge, and achieve better performance.

After the experiments, I consider the model of Config 1 is still underfitting.

(d)

Config 1: depth: 5, size: 100

```
Random Forest Classifier Accuracy: 0.9111111111111111
Random Forest Regressor MSE: 23.4852529856091
```

Config 2: depth: 10, size: 100

```
Random Forest Classifier Accuracy: 0.9111111111111111
Random Forest Regressor MSE: 23.704580444810183
```

Config 3: depth: 5, size: 200

```
Random Forest Classifier Accuracy: 0.9111111111111111
Random Forest Regressor MSE: 23.398899871131306
```

Discussion
Larger max-depth leads to higher complexity, lower generalization, and higher chance of overfitting. Larger forest size leads to higher complexity, higher generalization, and less chance of overfitting. Finally, I choose max-depth as 5 and forest size as 100 considering model complexity, generalization, and overfitting.

(e)

Linear Model

Strengths:

1. Extremely low model complexity
2. Less prone to overfitting
3. Works well with linear relationships

Weaknesses:

1. Limited expressiveness

Decision Tree

Strengths:

1. Non-linearity
2. Low model complexity
3. Feature importance
4. Better expressiveness

Weaknesses:

1. Overfitting

Random Forest

Strengths:

1. Non-linearity
2. Robustness
3. Feature importance
4. Better expressiveness

Weaknesses:

1. High model complexity
2. The decision of the forest is less interpretable due to majority voting.

In my opinion, if the task is easy, I would choose linear model for the sake of model complexity. Otherwise, I would choose random forest.
I think under no circumstances will I choose decision tree because it is vulnerable to noises and overfitting.