Pattern Recognition: Homework 2

Due date: 2023.3.7

Problem 1 (10 pt)

Suppose there is a linear classifier

$$f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b,$$

where $\boldsymbol{w} \in \mathbb{R}^d$ and b are parameters. The decision boundary is hyperplane $H : \{\boldsymbol{x} : f(\boldsymbol{x}) = 0\}$. Give the distance of any point $\boldsymbol{v} \in \mathbb{R}^d$ to H. (Distance means $d(\boldsymbol{v}, H) = \min_{\boldsymbol{x} \in H} \|\boldsymbol{x} - \boldsymbol{v}\|_2$)

Problem 2 (20 pt)

In our class, we have learned the deduction for Fisher's criterion as maximum the ratio $J_F(\boldsymbol{w}) = \frac{S_b}{S_w}$. Actually, there is another way to deduce it. Suppose we have a set of points $\{(\boldsymbol{x}_i, y_i)\}, i = 1, \ldots, N$ where $\boldsymbol{x}_i \in \mathbb{R}^d$ and $y_i \in \{\frac{N}{N_1}, -\frac{N}{N_2}\}$. There are N_1 positive sample with positive label value N/N_1 and vice versa, $N_1 + N_2 = N$. We build a classification function as $f(\boldsymbol{x}; \boldsymbol{w}, b) = \boldsymbol{w}^\top \boldsymbol{x} + b$. The classification error for a certain data point is defined as $L(f(\boldsymbol{x}_i), y_i) = \frac{1}{2}(f(\boldsymbol{x}_i) - y_i)^2$. Prove that the Fisher's criterion for selecting $\boldsymbol{w}^*$ in our class as

$$\boldsymbol{w}^* = S_W^{-1}(\boldsymbol{m}_1 - \boldsymbol{m}_2),$$

is parallel to the $\boldsymbol{w}^\star$ in solution

$$\boldsymbol{w}^\star, b^\star = \mathrm{argmin}_{\boldsymbol{w}, b} \frac{1}{N} \sum_{i=1}^{N} L(f(\boldsymbol{x}_i; \boldsymbol{w}, b), y_i)$$

So we know Fisher's criterion is actually finding the optimal linear classifier under loss function $L(y, \hat{y}) = -\frac{1}{2}(y - \hat{y})^2$

Problem 3 (30 pt)

Denote Sigmoid function as $\sigma(x) = \frac{1}{1+e^{-x}}$. Prove the following statement holds

- $\sigma(x) + \sigma(-x) = 1$.
- $\sigma'(x) = \sigma(x)(1 - \sigma(x))$. (And this is important for back-propagation through sigmoid function.)
- $\tanh(x) = 2\sigma(x) - 1$.

Bonus (10 pt)

Suppose we have a classifier $f(\boldsymbol{x}) = \sigma(\boldsymbol{w}^\top \boldsymbol{x} + b)$, and a loss function $L(\hat{y}, y) = (y - \hat{y})^2$.
Compute the gradient $\frac{\partial L(f(\boldsymbol{x}), y)}{\partial \boldsymbol{w}}$, $\frac{\partial L(f(\boldsymbol{x}), y)}{\partial b}$.

Problem 4 (40 pt)

In this problem, you need to write a linear classifier in different ways to get a taste of the content in class. **Please notice that you should not use any package that solves the problem in very few lines like `scipy.stats.linregress`. You should only use package like `numpy` to build up the model on your own, otherwise you will not get any points.**

You will write a classifier for predicting whether a person is likely to have breast cancer. In the attachment is our data file `breast-cancer-wisconsin.txt`. The file consists of 699 lines, each line with 11 integer attributes (or features) as below

1 初始化调小，和x点积之后处于[-1, 1]
2 rho 调小

Attribute Domain

```
1.  Sample code number              id number
2.  Clump Thickness                 1 - 10
3.  Uniformity of Cell Size         1 - 10
4.  Uniformity of Cell Shape        1 - 10
5.  Marginal Adhesion               1 - 10
6.  Single Epithelial Cell Size     1 - 10
7.  Bare Nuclei                     1 - 10
8.  Bland Chromatin                 1 - 10
9.  Normal Nucleoli                 1 - 10
10. Mitoses                         1 - 10
11. Class:                          (0 for benign, 1 for malignant)
```

1 (10 pt)

Adopt Fisher's criterion to find the optimal linear classifier using attributes 2 to 10 to predict label 11. Give the 9-dimensional unit norm vector for $\boldsymbol{w}^*$, and the classification accuracy on the dataset.

2 (20 pt)

Using logistic regression, namely the classifier $f(\boldsymbol{x}) = \sigma(\boldsymbol{w}^\top \boldsymbol{x} + b)$ in problem3 to do the classification. You can follow the procedure below

- 1. Randomly sample the initial parameter $\boldsymbol{w}_0$ from i.i.d. Gaussian and choose $b_0 = 0$.
- 2. Use loss function $L(\boldsymbol{w}, b) = \sum_{i=1}^{N} \frac{1}{2}(\sigma(\boldsymbol{w}^\top \boldsymbol{x}_i + b) - y_i)^2$ to compute the loss value of the current classifier on all the 699 data.
- 3. Compute the gradient $\left.\frac{\partial L}{\partial \boldsymbol{w}}\right|_{\boldsymbol{w}_t}, \left.\frac{\partial L}{\partial b}\right|_{b_t}$.
- 4. Pick a proper (small) value ρ, update $\boldsymbol{w}_t = \boldsymbol{w}_{t-1} - \rho \nabla_{\boldsymbol{w}} L(\boldsymbol{w}, b)$, $b_t = b_{t-1} - \rho \nabla_b L(\boldsymbol{w}, b)$.
- 5. Go back to 2 until the loss is sufficiently low (or repeat for enough iterations).

You need to specify the ρ you use, plot the loss value against iterations, and report the final classification accuracy.

3 (10 pt)

Compare the cosine between two $\boldsymbol{w}^*$ you get in sections 1 and 2. How similar are they? Why? And try to figure out the most indicative feature that implies one gets breast cancer from $\boldsymbol{w}^*$.

注意归一化影响，
应该除上feature的std再进行比较