Machine Learning Assignment 4

SVM

1. [35pts] Support Vector Machine

(1) Recall that the soft margin support vector machine solves the problem:

$$min \quad \frac{1}{2}w^\top w + C\sum_i \varepsilon_i$$

$$\text{s.t.} \quad y_i(w^\top x_i + b) \geq 1 - \varepsilon_i, \quad \varepsilon_i \geq 0.$$

a) [10pts] Derive its dual problem using the method of Lagrange multipliers.

b) [10pts] Further simplify the dual problem when at its saddle point to prove

$$\max_{\alpha} \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j x_i^\top x_j$$

$$\text{s.t.} \; C \geq \alpha_i \geq 0, \quad \sum_i \alpha_i y_i = 0,$$

is equivalent to the primal problem.

Solution.

a) Rewrite the system of equations:

$$min \quad \frac{1}{2}w^\top w + C\sum_i \varepsilon_i$$

$$\text{s.t.} \; 1 - \varepsilon_i - y_i(w^\top x_i + b) \leq 0$$

$$-\varepsilon_i \leq 0.$$

We can write the Lagrange function where α and μ is the multipliers:

$$L(\boldsymbol{w}, b, \varepsilon, \alpha, \mu) = \frac{1}{2}\boldsymbol{w}^\top\boldsymbol{w} + C\sum_i \varepsilon_i + \sum_i \alpha_i[1 - \varepsilon_i - y_i(\boldsymbol{w}^\top\boldsymbol{x_i} + b)] - \sum_i \mu_i \varepsilon_i \quad (1)$$

Thus, the dual problem of the original problem is:

$$\max_{\alpha,\mu} \min_{w,b,\varepsilon} L(\boldsymbol{w}, b, \varepsilon, \alpha, \mu)$$

$$s.t. \quad \alpha_i \geq 0$$

$$\mu_i \geq 0$$

b) The partial derivative of w, b, ε for L is as follows:

$$\frac{\partial L}{\partial \boldsymbol{w}} = \boldsymbol{w} - \sum_i \alpha_i y_i \boldsymbol{x_i}$$

$$\frac{\partial L}{\partial b} = \sum_i \alpha_i\, y_i$$

$$\frac{\partial L}{\partial \varepsilon_i} = C - \alpha_i - \mu_i$$

Let them equal to zero:

$$\boldsymbol{w} = \sum_i \alpha_i y_i \boldsymbol{x_i}$$

$$\sum_i \alpha_i\, y_i = 0$$

$$C = \alpha_i + \mu_i$$

Bring them into (1) we can obtain that the dual function is:

$$\mathrm{g}(\alpha, \mu) = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i\, \alpha_j y_i y_j x_i^{\top} x_j$$

From:

$$C = \alpha_i + \mu_i$$

$$\alpha_i \geq 0$$

$$\mu_i \geq 0$$

We can have that:

$$0 \leq \alpha_i \leq C$$

Finally the following system of equations:

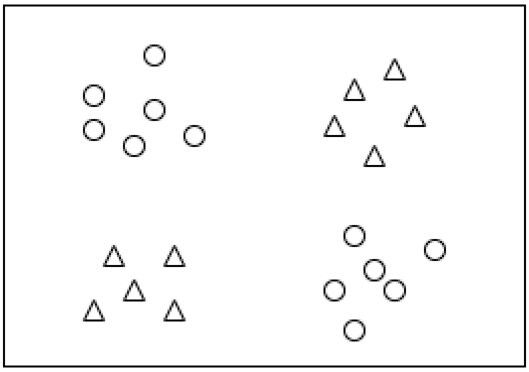
$$\max_{\alpha} \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i\, \alpha_j y_i y_j x_i^{\top} x_j$$

$$\text{s.t. } C \geq \alpha_i \geq 0, \quad \sum_i \alpha_i\, y_i = 0$$

is equivalent to the primal problem.

(2) [15pts] Given the XOR sample points as below, we train an SVM with a quadratic kernel, i.e. our kernel function is a polynomial kernel of degree 2: $\kappa(x_i, x_j) = (x_i^T x_j)^d, d = 2$.

(a) [5pts] what is the corresponding mapping function $\phi(x)$?



(b) [5pts] Use the following code to generate XOR data, and according to the answer of (a), map the data with $\phi(x)$ to see if it can be linearly separable.

(c) [5pts] Could we get a reasonable model with hard margin (after feature mapping)? If yes, draw the decision boundary in the figure (original feature space), otherwise state reasons.

```
import numpy as np
import matplotlib.pyplot as plt
#创建数据
X_xor = np.random.randn(40,2)
y_xor = np.logical_xor(X_xor[:,0]>0, X_xor[:,1]>0)
y_xor = np.where(y_xor, 1, -1)
#绘制散点图
plt.scatter(x=X_xor[y_xor==1,0]),  # 横坐标
            y=X_xor[y_xor==1,1]),  # 纵坐标
            color='g', marker='x', label='1')
plt.scatter(x=X_xor[y_xor==-1,0]),
            y=X_xor[y_xor==-1,1]),
            color='b', marker='o', label='-1')
plt.legend() #显示图例
plt.show()
```

Solution.

(a) For three-dimensional input vectors $\boldsymbol{x} = (x_1, x_2)^T$ and $\boldsymbol{y} = (y_1, y_2)^T$, the kernel function can be expanded as:

$$\boldsymbol{\kappa}(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x}^T \boldsymbol{y})^2 = (x_1 y_1 + x_2 y_2)^2$$

Expand the squared term:

$$\boldsymbol{\kappa}(\boldsymbol{x}, \boldsymbol{y}) = (x_1 y_1 + x_2 y_2)^2$$
$$= {x_1}^2 {y_1}^2 + {x_2}^2 {y_2}^2 + 2 x_1 y_1 x_2 y_2$$

The above expansion corresponds to the inner product in a higher-dimensional space. Thus, we can deduce the corresponding feature mapping $\phi(x)$ that would produce this inner product.

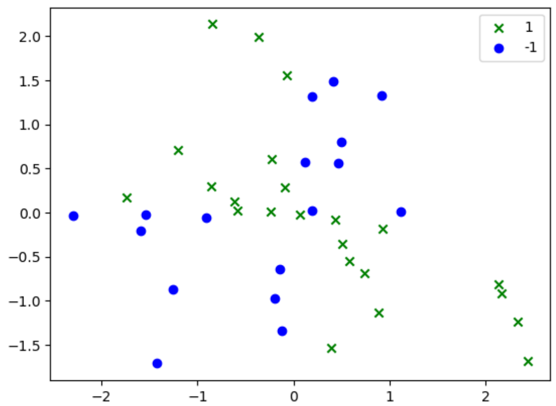
$$\boldsymbol{\kappa}(\boldsymbol{x}, \boldsymbol{y}) = (x_1 y_1 + x_2 y_2)^2$$
$$= {x_1}^2 {y_1}^2 + 2 x_1 y_1 x_2 y_2 + {x_2}^2 {y_2}^2$$
$$= \phi(\boldsymbol{x})^T \phi(\boldsymbol{y})$$

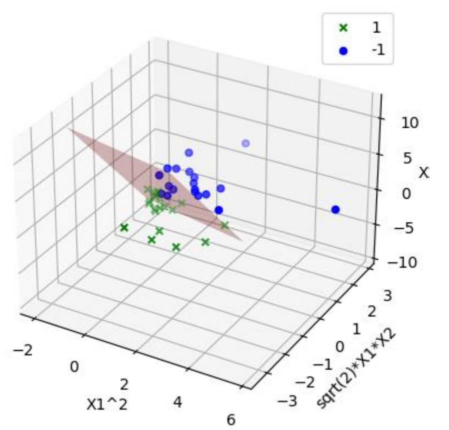
Consider the mapping function $\phi(x)$ that transforms the input vector $\boldsymbol{x} = (x_1, x_2)^T$ into a higher-dimensional space:

$$\phi(\boldsymbol{x}) = \left({x_1}^2, \sqrt{2} x_1 x_2, {x_2}^2\right)$$

This function maps the original 2-dimensional input vectors into a 3-dimensional feature space, enabling the SVM to find a linear separating hyperplane in this higher-dimensional space.

(b) Generated XOR data in 2-dimensional feature space is as follow (left) :



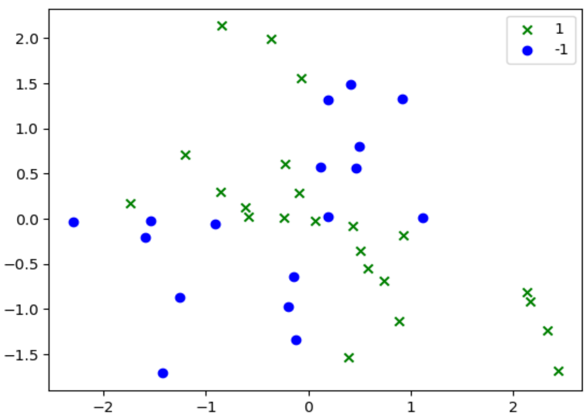


Apparently, it can't be linearly separable now.

After mapping by $\phi(\boldsymbol{x})$ into 3-dimensional feature space is as the right one.

Now, it can be linearly separable, with the Red plane.

(c) Yes. The hard margin is as the red one:



Bayesian Classifiers

1. [30pts] Naïve Bayes Classifier

Suppose you are given the following set of data with four Integer input variables A, B, C, and D, and a single binary label y.

In this task, the value of a variable means how many times it appears in text from the corresponding label (i.e., word frequency, which is a popular representation of text). For example, in the text of x_1 from label +1, word A appears twice, word B appears 4 times, word C appears 10 times and word D appears 3 times.

We are trying to fit a Naïve Bayes Classifier on this dataset.

	A	B	C	D	y
x_1	2	4	10	3	+1
x_2	3	1	4	2	+1
x_3	0	2	0	5	-1
x_4	2	0	4	0	+1
x_5	1	6	6	0	-1
x_6	0	2	1	7	-1
x_7	3	0	0	8	+1
x_8	6	1	2	7	-1

(1) [**20pts**] Calculate the empirical conditional probability of each variable for appearing in texts from each label. To illustrate, for variable A, calculate $p_{A,j} = P(word = A \mid y = j),\ j \in \{-1, +1\}$, and the same for the other variables. Remember to use Laplace smoothing to avoid zero probabilities.

(2) [**10pts**] Give a new sample where $A = 3, B = 2, C = 1, D = 2$. Predict its label. You should write down your calculation in detail. (It is enough to only give the form of a fraction, not necessarily calculated as a decimal, 仅给出分数形式即可，不一定需要计算为小数)

Solution.

(1) Laplacian correction formula:

$$\hat{\boldsymbol{P}}(\boldsymbol{x_i}|\boldsymbol{c}) = \frac{|\boldsymbol{D}_{\boldsymbol{x_i},\boldsymbol{c}}| + \mathbf{1}}{|\boldsymbol{D_c}| + \boldsymbol{N_i}}$$

Where N_i represents the number of possible values of feature x_i.

In this case, x_i represents the appeared word, $N_i = 4\ (A, B, C\ or\ D)$.

$$\hat{P}(A|+1) = \frac{|D_{A,+1}| + 1}{|D_{+1}| + 4} = \frac{10 + 1}{46 + 4} = \frac{11}{50}, \hat{P}(A|-1) = \frac{|D_{A,-1}| + 1}{|D_{-1}| + 4} = \frac{7 + 1}{46 + 4} = \frac{8}{50}$$

$$\hat{P}(B|+1)=\frac{|D_{B,+1}|+1}{|D_{+1}|+4}=\frac{5+1}{46+4}=\frac{6}{50},\hat{P}(B|-1)=\frac{|D_{B,-1}|+1}{|D_{-1}|+4}=\frac{11+1}{46+4}=\frac{12}{50}$$
$$\hat{P}(C|+1)=\frac{|D_{C,+1}|+1}{|D_{+1}|+4}=\frac{18+1}{46+4}=\frac{19}{50},\hat{P}(A|-1)=\frac{|D_{C,-1}|+1}{|D_{-1}|+4}=\frac{9+1}{46+4}=\frac{10}{50}$$
$$\hat{P}(D|+1)=\frac{|D_{D,+1}|+1}{|D_{+1}|+4}=\frac{13+1}{46+4}=\frac{14}{50},\hat{P}(A|-1)=\frac{|D_{D,-1}|+1}{|D_{-1}|+4}=\frac{19+1}{46+4}=\frac{20}{50}$$

(2) Laplacian correction formula:

$$\hat{\boldsymbol{P}}(\boldsymbol{c})=\frac{|\boldsymbol{D}_{\boldsymbol{c}}|+\mathbf{1}}{|\boldsymbol{D}|+\boldsymbol{N}}$$

Where N represents the number of categories.

In this case, $N=2\ (+1\ or\ -1)$.

We can obtain:

$$\hat{P}(y=+1)=\hat{P}(y=-1)=\frac{46+1}{46\times 2+2}=\frac{1}{2}$$

From Bayesian formula:

$$\begin{aligned}\hat{P}(y=+1|A=3,B=2,C=1,D=2)&=\frac{\hat{P}(y=+1,A=3,B=2,C=1,D=2)}{\hat{P}(A=3,B=2,C=1,D=2)}\\&=\frac{\hat{P}(A=3,B=2,C=1,D=2|y=+1)\hat{P}(y=+1)}{\hat{P}(A=3,B=2,C=1,D=2)}\\&\propto \hat{P}(A=3,B=2,C=1,D=2|y=+1)\hat{P}(y=+1)\\&=(\frac{11}{50})^3\times(\frac{6}{50})^2\times\frac{11}{50}\times(\frac{14}{50})^2\times\frac{1}{2}\\&=\frac{178439184}{50^8\times 2}\end{aligned}$$

Similarly, we have:

$$\begin{aligned}\hat{P}(y=+1|A=3,B=2,C=1,D=2)&\propto(\frac{8}{50})^3\times(\frac{12}{50})^2\times\frac{10}{50}\times(\frac{20}{50})^2\times\frac{1}{2}\\&=\frac{294912000}{50^8\times 2}\end{aligned}$$

Because $\hat{P}(y=+1|A=3,B=2,C=1,D=2)<\hat{P}(y=-1|A=3,B=2,C=1,D=2)$

The label is predicted to be -1.

2. [35pts] Gaussian Bayesian Classifiers

Given data set $D=\{(x_1,y_1),(x_2,y_2),\dots,(x_m,y_m)\}$, where $y\in Y=\{1,2,\dots,K\}$.

(1) [**5pts**] Please write down the Bayes optimal classifier that minimizes the misclassification error rate.

(2) [**15pts**] Suppose the samples in the k-th class are i.i.d. sampled form normal distribution $\mathcal{N}(\mu_k,\Sigma)$, ($k=1,2,\dots,\mathrm{K}$, all classes share the same covariance matrix). Let m_k denote the number of samples in the k-th class, and the prior probability $P(y=k)=\pi_k$. If $x\in R^d\sim\mathcal{N}(\mu,\Sigma)$, then the probability density function is:

$$p(x)=\frac{1}{(2\pi)^{d/2}\,det(\Sigma)^{1/2}}exp\left(-\frac{1}{2}(x-\mu)^{\top}\Sigma^{-1}(x-\mu)\right)$$

Please write down the corresponding Bayes optimal classifier.

(3) [**15pts**] For binary classification problem, please prove that when samples in each class are i.i.d. sampled from normal distributions which share the same covariance matrix and the two classes have equal prior probabilities $\pi_0=\pi_1$, LDA (Linear Discriminant Analysis) gives the Bayes optimal classifier.

Hint: The optimal solution of LDA is:

$$w=S_w^{-1}(\mu_0-\mu_1)$$

where S_w is within-class scatter matrix, $S_w=\Sigma_0+\Sigma_1$ (Σ_i is the covariance matrix of the i-th class).

Solution.

(1) Assume using the 0-1 loss, the conditional risk: $R(c|\boldsymbol{x})=1-P(c|\boldsymbol{x})$

The Bayes optimal classifier that minimize the misclassification error rate is:

$$h^*(x)=arg\min_{x\in D}R(c|x)=arg\max_{x\in D}P(c|x)$$

(2) $h^*(\boldsymbol{x})=arg\max_{y}P(y|\boldsymbol{x})$

$$=arg\max_{y}\ln P(y|\boldsymbol{x})$$

$$=arg\max_{y}\ln P(y)P(\boldsymbol{x}|y)$$

$$=arg\max_{y}\ln\pi_y-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_y)^{\boldsymbol{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_y)$$

$$=arg\max_{y}\ln\pi_y-\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{x}+\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_y-\frac{1}{2}\boldsymbol{\mu}_y{}^{T}\Sigma^{-1}\boldsymbol{\mu}_y$$

(3) the discriminant function:

$$\boldsymbol{g}(\boldsymbol{x})=\boldsymbol{P}(\boldsymbol{w}_1|\boldsymbol{x})-\boldsymbol{P}(\boldsymbol{w}_2|\boldsymbol{x})=\ln\frac{\boldsymbol{P}(\boldsymbol{x}|\boldsymbol{w}_1)}{\boldsymbol{P}(\boldsymbol{x}|\boldsymbol{w}_2)}+\ln\frac{\boldsymbol{P}(\boldsymbol{w}_1)}{\boldsymbol{P}(\boldsymbol{w}_2)}$$

$$=\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{\Sigma}^{-1}(\mu_0-\mu_1)-\frac{\boldsymbol{1}}{\boldsymbol{2}}(\mu_0-\mu_1)^{\boldsymbol{T}}\boldsymbol{\Sigma}^{-1}(\mu_0-\mu_1)+\ln(\frac{\pi_0}{\pi_1})$$

The optimal solution of LDA is:

$$\boldsymbol{w}=S_w^{-1}(\mu_0-\mu_1)=(2\boldsymbol{\Sigma})^{-1}(\mu_0-\mu_1)$$

The mid point:

$$\boldsymbol{c}=\frac{\boldsymbol{1}}{\boldsymbol{2}}(\mu_0-\mu_1)^{\boldsymbol{T}}\boldsymbol{w}=\frac{\boldsymbol{1}}{\boldsymbol{4}}(\mu_0-\mu_1)^{\boldsymbol{T}}\boldsymbol{\Sigma}^{-1}(\mu_0-\mu_1)$$

The decision boundary of LDA is:

$$\boldsymbol{f}(\boldsymbol{x})=\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}-\boldsymbol{c}=\frac{\boldsymbol{1}}{\boldsymbol{2}}\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{\Sigma}^{-1}(\mu_0-\mu_1)-\frac{\boldsymbol{1}}{\boldsymbol{4}}(\mu_0-\mu_1)^{\boldsymbol{T}}\boldsymbol{\Sigma}^{-1}(\mu_0-\mu_1)$$

3. [30pts] MLE and Linear Regression

Sample points come from an unknown distribution, $X_i \sim D$. Labels y_i are the sum of a deterministic function $f(\mathrm{X_i})$ plus random noise: $y_i = f(X_i) + \varepsilon_i$, where $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$.

For this problem, we will assume that $\varepsilon_i \sim \mathcal{N}(0, \sigma_i^2)$—that is, the variance σ_i^2 of the noise is different for each sample point and we will examine how our loss function changes as a result. We assume that we know the value of each σ_i^2. You are given an $n \times p$ design matrix X, an n-dimensional vector y of labels, such that the label y_i of sample point X_i is generated as described above, and a list of the noise variances σ_i^2.

(1) [**10pts**] Apply MLE to derive the optimization problem that will use the maximum likelihood estimate of the distribution parameter f. (Note: f is a function, but we can still treat it as the parameter of an optimization problem.) Express your Objective function as a summation of loss functions, one per sample point.

(2) [**10pts**] We decide to do linear regression, so we parameterize $f(X_i)$ as $f(X_i) = w \cdot X_i$, where w is a p-dimensional vector of weights. Write an equivalent optimization problem where your optimization variable is w and the cost function is a function of X, y, w, and the variances σ_i^2. Find a way to express your cost function in matrix notation. (Hint: You can define a new matrix.)

(3) [**10pts**] Write the solution to your optimization problem as the solution of a linear system of equations. (Again, in matrix notation.)

Solution.

(1) The Maximum likelihood function:

$$\ell L = \sum_i \ln \frac{1}{\sqrt{2\pi\sigma_i}} \exp(-\frac{1}{2}\left(\frac{y_i - f(X_i)}{\sigma_i}\right)^2) = \sum_i \ln \frac{1}{\sqrt{2\pi\sigma_i}} - \frac{1}{2}\sum_i \left(\frac{y_i - f(X_i)}{\sigma_i}\right)^2)$$

The loss function:

$$loss = \frac{1}{2}\sum_i \left(\frac{y_i - f(X_i)}{\sigma_i}\right)^2$$

(2) Expanding the loss function:

$$loss = \frac{1}{2}\sum_i \left(\frac{y_i - f(X_i)}{\sigma_i}\right)^2 = \frac{1}{2}\sum_i \frac{1}{\sigma_i{}^2}(y_i - w \cdot X_i)^2 = \frac{1}{2}(\boldsymbol{y} - \boldsymbol{Xw})^T \boldsymbol{\Sigma}(\boldsymbol{y} - \boldsymbol{Xw})$$

(3) Set the partial of w for $loss$ zero:

$$\frac{\partial loss}{\partial \boldsymbol{w}} = \boldsymbol{X}^T \boldsymbol{\Sigma}(\boldsymbol{y} - \boldsymbol{Xw}) = \boldsymbol{0}$$

We can obtain that:

$$\boldsymbol{w} = (\boldsymbol{X}^T \boldsymbol{\Sigma} \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{\Sigma} \boldsymbol{y}$$