CS182 - Introduction to Machine Learning, 2021-22 Fall

Final Exam

9:00AM - 11:00AM, Saturday, Jan. 8th, 2022

12 pages, 6 problems, and 120 points (including 20 points for bonus) in total

(NOTE: your exam grade will be counted as min{100,"the grade on test paper"})

Problem 1. (20 points) *(Bayesian Decision Theory and Linear Discrimination)*

1) Consider a binary classification problem, with the class conditional density

$$p(\mathbf{x} \mid C_i)=\frac{1}{(2\pi)^{\frac{d}{2}}\det(\mathbf{\Sigma}_i)^{\frac{1}{2}}}\exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T\mathbf{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right],\ i=1,2.$$

Assume that $\mathbf{\Sigma}_i=\sigma^2\mathbf{I}$ for all i, and that the priors for the two classes are not equal, i.e., $P(C_1)\neq P(C_2)$. If our target is to minimize the expected loss, a.k.a. conditional risk, (as in the lecture notes, we denote by λ_{ij} the loss incurred for assigning an input $\mathbf{x}$ to class C_i when the actual state is C_j), derive the decision boundary, and explain how geometrically it differs from that when one minimizes the misclassification error, a.k.a. probability of error. (10 points)

2) Assume that $\mathbf{x}$ is 2-dimensional, and that the two dimensions are independent following the Laplace distribution:

$$p(x_j \mid C_i)=\frac{1}{2\sigma}\exp(-\frac{|x_j-\mu_{ij}|}{\sigma}),\ j=1,2.$$

By minimizing the misclassification error, obtain and draw the decision boundary when $\mu_{11}=1$, $\mu_{12}=1$, $\mu_{21}=3$, $\mu_{22}=5$, $\sigma=1$, and $P(C_1)=P(C_2)$. (10 points)

Solution:

1) When using the loss functions in the binary classification case, we will decide ω_1 if $(\lambda_{21}-\lambda_{11})p(\mathbf{x}|C_1)P(C_1)>(\lambda_{22}-\lambda_{12})p(\mathbf{x}|C_2)P(C_2)$.$(1')$ Taking the logarithm and the discriminant function can be obtained by

$$\begin{aligned}g_i(\mathbf{x})&=\ln c_i+\ln p\left(\mathbf{x}\mid C_i\right)+\ln P\left(C_i\right)\\&=-\frac{\left\|\mathbf{x}-\boldsymbol{\mu}_i\right\|^2}{2\sigma^2}+\left[\ln c_i+\ln P\left(C_i\right)+const.\right],(1')\end{aligned}$$

where $c_1=(\lambda_{21}-\lambda_{11})$, $c_2=(\lambda_{22}-\lambda_{12})$, $const.$ is a constant.

By applying the result in slides, the decision boundary is a line: $\mathbf{w}^T(\mathbf{x}-\mathbf{x}_0)=0$,$(1')$ where

$$\begin{aligned}\mathbf{w}&=\boldsymbol{\mu}_1-\boldsymbol{\mu}_2,\\ \mathbf{x}_0&=\frac{1}{2}\left(\boldsymbol{\mu}_1+\boldsymbol{\mu}_2\right)-\frac{\sigma^2}{\left\|\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right\|^2}\ln\frac{c_1P\left(C_1\right)}{c_2P\left(C_2\right)}\left(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right).(2')\end{aligned}$$

We know the decision boundary obtained by minimizing the error is the line $\mathbf{w}^T(\mathbf{x}-\mathbf{x}_{e0})=0$, where

$$\mathbf{x}_{e0}=\frac{1}{2}\left(\boldsymbol{\mu}_1+\boldsymbol{\mu}_2\right)-\frac{\sigma^2}{\left\|\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right\|^2}\ln\frac{P\left(C_1\right)}{P\left(C_2\right)}\left(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right).(1')$$

Since

$$\begin{aligned}\mathbf{x}_0 &= \frac{1}{2}\left(\boldsymbol{\mu}_1+\boldsymbol{\mu}_2\right)-\frac{\sigma^2}{\left\|\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right\|^2}\ln\frac{c_1P\left(C_1\right)}{c_2P\left(C_2\right)}\left(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right)\\ &= \frac{1}{2}\left(\boldsymbol{\mu}_1+\boldsymbol{\mu}_2\right)-\frac{\sigma^2}{\left\|\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right\|^2}\ln\frac{P\left(C_1\right)}{P\left(C_2\right)}\left(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right)-\frac{\sigma^2}{\left\|\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right\|^2}\ln\frac{c_1}{c_2}\left(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right)(2')\\ &= \mathbf{x}_{e0}-\frac{\sigma^2}{\left\|\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right\|^2}\ln\frac{c_1}{c_2}\left(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2\right),\end{aligned}$$

The decision boundary obtained by minimizing the loss is a translation of its error minimization counterpart.(2′)

2) When minimize error, the discriminant functions are

$$g_i(\mathbf{x}) = -\ln(2\sigma)-\frac{\|\mathbf{x}-\boldsymbol{\mu}_i\|_1}{\sigma}+\ln P(C_i), \quad i=1,2.(2')$$

The decision boundary satisfies the equation

$$g_1(\mathbf{x})-g_2(\mathbf{x})=0.$$

That is,

$$\left\|\mathbf{x}-\boldsymbol{\mu}_1\right\|_1-\left\|\mathbf{x}-\boldsymbol{\mu}_2\right\|_1-\sigma\ln\frac{P\left(C_1\right)}{P\left(C_2\right)}=0.(2')$$

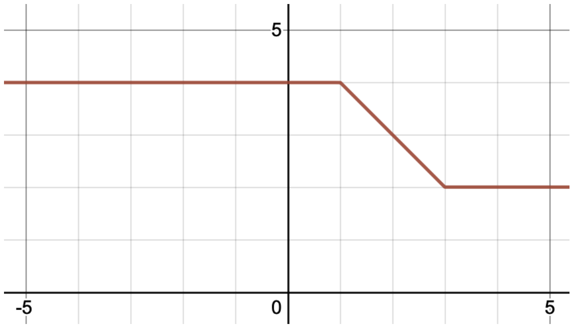
Using those values, the boundary satisfies

$$(|x_1-1|-|x_1-3|)+(|x_2-1|-|x_2-5|)=0.(2')$$

The decision boundary is

$$\begin{cases} x_1+x_2=5, & 1\le x_1\le 3,\\ x_2=2, & x_1>3, \qquad (2')\\ x_2=4, & x_1<1.\end{cases}$$

The following figure shows the decision boundary(2′)



Problem 2. (20 points) *(SVM and Decision Trees)*

A novel multi-class classification method is called Decision Trees SVM (DT-SVM), which is a binary decision tree where every binary split is trained by an SVM. Given the input feature $\mathbf{x}$, a classification result from the DT-SVM is given in the following figure where C_i denotes the region of class i.

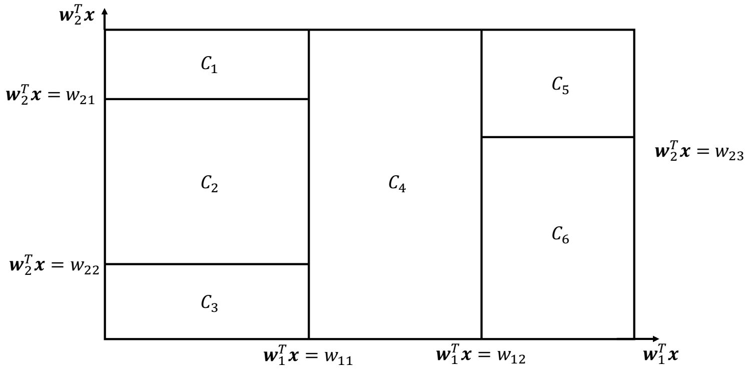
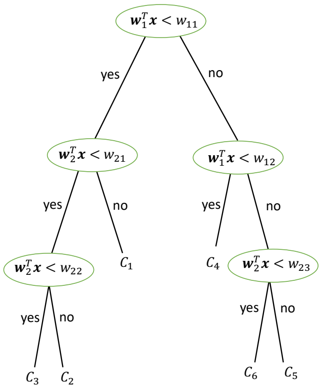

$\boldsymbol{w}_2^T\boldsymbol{x}$
C_1
$\boldsymbol{w}_2^T\boldsymbol{x} = w_{21}$
C_5
$\boldsymbol{w}_2^T\boldsymbol{x} = w_{23}$
C_2
C_4
C_6
$\boldsymbol{w}_2^T\boldsymbol{x} = w_{22}$
C_3
$\boldsymbol{w}_1^T\boldsymbol{x} = w_{11}$
$\boldsymbol{w}_1^T\boldsymbol{x} = w_{12}$
$\boldsymbol{w}_1^T\boldsymbol{x}$

Figure: A classification result from the DT-SVM.

1) Draw the corresponding binary decision tree according to the above figure. (6 points)
2) You have learned the hard-margin SVMs and the soft-margin SVMs. Discuss the differences between them. (7 points)
3) Suppose the decision boundary $\mathbf{w}_1^T\mathbf{x} = w_{11}$ is determined based on a soft-margin SVM, write down the corresponding primal optimization problem. (7 points)

Solution:

1) The decision tree is


$\boldsymbol{w}_1^T\boldsymbol{x} < w_{11}$
yes
no
$\boldsymbol{w}_2^T\boldsymbol{x} < w_{21}$
$\boldsymbol{w}_1^T\boldsymbol{x} < w_{12}$
yes
no
yes
no
$\boldsymbol{w}_2^T\boldsymbol{x} < w_{22}$
C_1
C_4
$\boldsymbol{w}_2^T\boldsymbol{x} < w_{23}$
yes
no
yes
no
C_3
C_2
C_6
C_5

(6')

2) Hard margin: We use the hard-margin SVM, when the data is linearly separable. It does not allow misclassifications. (3.5')

Soft margin: We use the soft-margin SVM in these two cases. (a) The separating hyperplane does not exist, because the data is not linearly separable or it has a high noise level which causes a large overlap of the classes. (b) The separating hyperplane exists, but it is not the best solution when there exist outliers in the data. (3.5')

3) Define $r^{(\ell)} = -1$ for $y^{(\ell)} \in \{C_1, C_2, C_3\}$, and $r^{(\ell)} = 1$ for $y^{(\ell)} \in \{C_4, C_5, C_6\}$. (3') The primal optimization problem is

$$\begin{aligned} \underset{\mathbf{w}_1, w_{11}, \xi_\ell}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}_1\|^2 + \eta \sum_{\ell} \xi_\ell \\ \text{subject to} \quad & r^{(\ell)} \left(\mathbf{w}_1^T \mathbf{x}^{(\ell)} + w_{11} \right) \geq 1 - \xi_\ell, \quad \forall \ell \end{aligned}$$

(A correct structure of soft-margin SVM. (1') Specification of $\mathbf{w}_1$ and w_{11} in the optimization problem. (3'))

Problem 3. (20 points) *(Multilayer Perceptron (MLP) and CNN)*

1) A single output MLP is considered with its output given by

$$y^{(\ell)} = \sum_{h=1}^{H} v_h z_h^{(\ell)} + v_0,$$

where

$$z_h^{(\ell)} = \mathsf{sigmoid}(\mathbf{w}_h^T \mathbf{x}^{(\ell)}) = \mathsf{sigmoid}(\sum_j w_{hj} x_j^{(\ell)}),$$

with $\mathsf{sigmoid}(a) = \frac{1}{1+\mathsf{exp}(-a)}$. Given sample $\{\mathbf{x}^{(\ell)}, r^{(\ell)}\}_\ell$, the loss function in learning is defined as

$$\mathcal{L} = \frac{1}{2}\sum_\ell (r^{(\ell)} - y^{(\ell)})^2.$$

Compute $\frac{\partial \mathcal{L}}{\partial w_{hj}}$ and explain why the learning algorithm is called "back-propagation". (10 points)

2) Given a 2-dimensional convolution layer in a CNN, the input matrix $\mathbf{X}$ and the convolution kernel $\mathbf{W}$ are defined by

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix},$$

and

$$\mathbf{W} = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix}.$$

   a) Write down the output of the convolution

$$\mathbf{Y} = \mathbf{X} * \mathbf{W}$$

   with no kernel flipping (you do not need to flip the kernel in computation) and no zero padding. (5 points)

   b) Discuss the properties of the convolution operation in comparison with the "fully-connected" weights in the conventional neural networks like the MLP discussed in 1). (5 points)

Solution:

1) Observed that

$$\frac{\partial \mathcal{L}}{\partial w_{hj}} = \sum_\ell \frac{\partial \mathcal{L}^{(\ell)}}{\partial y^{(\ell)}} \frac{\partial y^{(\ell)}}{\partial z_h^{(\ell)}} \frac{\partial z_h^{(\ell)}}{\partial w_{hj}} \quad (1')$$

$$= -\sum_\ell \left(r^{(\ell)} - y^{(\ell)}\right) \times v_h \times z_h^{(\ell)}\left(1 - z_h^{(\ell)}\right) x_j^{(\ell)} = -\sum_\ell \left(r^{(\ell)} - y^{(\ell)}\right) v_h z_h^{(\ell)}\left(1 - z_h^{(\ell)}\right) x_j^{(\ell)}. \quad (4')$$

The $(r^{(\ell)} - y^{(\ell)})v_h$ acts like the error term for hidden unit h. $(r^{(\ell)} - y^{(\ell)})$ is the error in the output which is backpropagated from the output to the hidden unit weighted by the "responsibility" of the hidden unit as given by its weight v_h. (5')

2) (a) $(\mathbf{y})_{mn} = (\mathbf{X} * \mathbf{W})_{mn} = \sum_{i=-1}^{1}\sum_{j=-1}^{1} x_{i+m,j+n} w_{ij}$. (5') (b) You answer should cover the following two points: sparse connectivity (2.5') and parameter sharing (2.5').

Problem 4. (15 + 5 points) *(Parameter Estimation, Clustering, and Nonparametric Methods)*

Given sample $\mathcal{X} = (\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)})$ generated from some unknown distribution $p(\mathbf{x})$.

1) Suppose $p(\mathbf{x})$ is a multivariate Gaussian distribution model $p(\mathbf{x}) = \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$, derive the estimates of parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ based on MLE. (5 points)

2) Suppose $p(\mathbf{x})$ is a multivariate Gaussian mixture model

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \quad \sum_{k=1}^{K} \pi_k = 1,\ \pi_k \geq 0,\ \forall k = 1, \ldots, K,$$

with parameters π_k, $\boldsymbol{\mu}_k$, and $\boldsymbol{\Sigma}_k$,

   a) discuss the advantages of this model over the multivariate Gaussian distribution model in 1) and why it can be used for clustering; (5 points)

   b) **(Bonus Question)** describe the idea of expectation-maximization (EM) algorithm for MLE of the parameters π_k, $\boldsymbol{\mu}_k$, and $\boldsymbol{\Sigma}_k$, and how the MLE results derived in 1) can be used in the EM algorithm. (5 points)

   (Hint: You are not required to write down the steps of the EM algorithm.)

3) Suppose dimensions of $\mathbf{x}$ are independent from each other, provide an approach to estimating $p(\mathbf{x})$ based on univariate nonparametric density estimation method. (5 points)

Solution:

1) Suppose $\mathbf{x}$ is a M-dimensional vector, then the log-likelihood is

$$\begin{aligned}
\mathcal{L} &= \log \prod_{n=1}^{N} \mathcal{N}\left(\mathbf{x}^{(n)} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \\
&= \sum_{n=1}^{N} \log \frac{1}{\sqrt{(2\pi)^M |\boldsymbol{\Sigma}|}} e^{-\frac{1}{2}\left(\mathbf{x}^{(n)} - \boldsymbol{\mu}\right)^T \boldsymbol{\Sigma}^{-1}\left(\mathbf{x}^{(n)} - \boldsymbol{\mu}\right)} \\
&= -\frac{1}{2}\sum_{n=1}^{N}\left(\log (2\pi)^M + \log |\boldsymbol{\Sigma}|\right) - \frac{1}{2}\sum_{n=1}^{N}\left(\mathbf{x}^{(n)} - \boldsymbol{\mu}\right)^T \boldsymbol{\Sigma}^{-1}\left(\mathbf{x}^{(n)} - \boldsymbol{\mu}\right) \\
&= \frac{1}{2}\sum_{n=1}^{N} \log \left|\boldsymbol{\Sigma}^{-1}\right| - \frac{1}{2}\sum_{n=1}^{N} \log (2\pi)^M - \frac{1}{2}\sum_{n=1}^{N} \operatorname{tr}\left(\left(\mathbf{x}^{(n)} - \boldsymbol{\mu}\right)\left(\mathbf{x}^{(n)} - \boldsymbol{\mu}\right)^T \boldsymbol{\Sigma}^{-1}\right). \quad \text{(1')}
\end{aligned}$$

By setting $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}} = \mathbf{0}$, we have

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}} = \sum_{n=1}^{N} \boldsymbol{\Sigma}^{-1}\left(\mathbf{x}^{(n)} - \boldsymbol{\mu}\right) = \mathbf{0}, \quad \text{(1')}$$

leading to the MLE estimator

$$\hat{\boldsymbol{\mu}} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}^{(n)}. \quad \text{(1')}$$

By setting $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\Sigma}^{-1}} = \mathbf{0}$, we have

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\Sigma}^{-1}} = \frac{1}{2}\sum_{n=1}^{N} \boldsymbol{\Sigma} - \frac{1}{2}\sum_{n=1}^{N}\left(\mathbf{x}^{(n)} - \boldsymbol{\mu}\right)\left(\mathbf{x}^{(n)} - \boldsymbol{\mu}\right)^T = \mathbf{0}, \quad \text{(1')}$$

leading to the MLE estimator

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{N} \sum_{n=1}^{N} \left(\mathbf{x}^{(n)} - \boldsymbol{\mu} \right) \left(\mathbf{x}^{(n)} - \boldsymbol{\mu} \right)^T . \quad \text{(1')}$$

2) (a) The multivariate Gaussian distribution model in 1) can only approximate Gaussian distributed data, while the multivariate Gaussian mixture model has the ability to approximate many naturally occurring real-world data thanks to the law of large numbers. (3') The clustering ability is natural obtained for mixture model as it classified the data into K components, i.e., clusters. (2')

(b) The EM algorithm is an iterative method to find (local) maximum likelihood or maximum a posteriori (MAP) estimates of parameters in statistical models, where the model depends on unobserved latent variables. The EM iteration alternates between performing an expectation (E) step, which creates a function for the expectation of the log-likelihood evaluated using the current estimate for the parameters, and a maximization (M) step, which computes parameters maximizing the expected log-likelihood found on the E step. These parameter-estimates are then used to determine the distribution of the latent variables in the next E step. The results in 1) can be used in the M step. (5')

3) Histogram estimator: First divide input space into equal-sized bins: $[x_0 + mh, x_0 + (m+1)h]$ with x_0 being the origin, h being the bin width, and m being an integer. Then the histogram estimator gives

$$\hat{p}(x) = \prod_{m=1}^{M} \frac{\sharp \left\{ x_m^{(n)} \text{ in the same bin as } x \right\}}{Nh} . \quad \text{(5')}$$

Other estimators like naive estimator, kernel estimator, and k-nearest neighbor estimator are also fine.

Problem 5. (15 + 5 points) *(Dimension Reduction and Matrix Factorization)*

Let $\mathbf{X}$ be a centered (i.e., zero-mean) sample matrix where each row is one observation and $\mathbf{\Sigma}$ be the sample covariance matrix. Assuming unit vector $\mathbf{w}_1$ is the eigenvector of $\mathbf{\Sigma}$ corresponding to the largest eigenvalue.

1) Prove that $\mathbf{w}_1$ is the principal component in principal component analysis (PCA) in that the projection onto direction $\mathbf{w}_1$ leads to the maximum variance for $\mathbf{X}$, i.e., $\mathbf{w}_1$ maximizes $\mathbf{w}^T\mathbf{\Sigma}\mathbf{w}$ for any $\mathbf{w}$ satisfying $\|\mathbf{w}\|_2 = 1$. (10 points)
2) Suppose the sample $\mathbf{X}$ is labeled into two classes, briefly describe the idea of linear discriminant analysis (LDA) and discuss the similarities and differences between PCA and LDA. (5 points)

   (Hint: You are not required to write down the detailed derivations of LDA.)
3) **(Bonus Question)** Show that $\mathbf{w}_1$ minimizes the reconstruction error $\|\mathbf{X} - \mathbf{X}\mathbf{w}\mathbf{w}^T\|_F^2$ for any $\mathbf{w}$ satisfying $\|\mathbf{w}\|_2 = 1$. (5 points)

   (Hint: You can directly use the result in 1).)

Solution:

1) The variance maximization problem is

$$\begin{aligned} \underset{\mathbf{w}}{\text{maximize}} \quad & \mathbf{w}^T\mathbf{\Sigma}\mathbf{w} \\ \text{subject to} \quad & \|\mathbf{w}\| = 1. \end{aligned}$$

   By setting the Lagrangian's derivative to zero as follows:

$$\frac{\partial}{\partial \mathbf{w}} - \mathbf{w}^T\mathbf{\Sigma}\mathbf{w} + \lambda\left(\mathbf{w}^T\mathbf{w} - 1\right) = 0,$$

   where λ is the Lagrange multiplier, leading to

$$\mathbf{\Sigma}\mathbf{w} = \lambda\mathbf{w}. \quad (5')$$

   Observe that

$$\mathbf{w}^T\mathbf{\Sigma}\mathbf{w} = \mathbf{w}^T\lambda\mathbf{w} = \lambda,$$

   as $\mathbf{w}_1$ corresponds to the largest λ, we can conclude that $\mathbf{w}_1$ maximizes the variance $\mathbf{w}^T\mathbf{\Sigma}\mathbf{w}$. (5')
2) LDA finds the vector $\mathbf{w}$ on which the data are projected such that the examples from the two classes are as well separated as possible. (2')

   Similarity of LDA and PCA: both are linear transformation techniques for dimension reduction. (1')

   Differences of LDA and PCA: LDA is a supervised technique that attempts to find a feature subspace that maximizes class separability, while PCA is unsupervised that focuses on capturing the direction of maximum variation in the data set. (2')
3) The reconstruction error minimization problem is

$$\begin{aligned} \underset{\mathbf{w}}{\text{maximize}} \quad & f(\mathbf{w}) = \|\mathbf{X} - \mathbf{X}\mathbf{w}\mathbf{w}^T\|_F^2 \\ \text{subject to} \quad & \|\mathbf{w}\| = 1. \end{aligned}$$

Observe that

$$\begin{aligned} f(\mathbf{w}) &= \left\|\mathbf{X} - \mathbf{X}\mathbf{w}\mathbf{w}^T\right\|_F^2 \\ &= \mathrm{tr}\left(\left(\mathbf{X} - \mathbf{X}\mathbf{w}\mathbf{w}^T\right)^T\left(\mathbf{X} - \mathbf{X}\mathbf{w}\mathbf{w}^T\right)\right) \\ &= \mathrm{tr}\left(\mathbf{X}^T\mathbf{X} - \mathbf{X}^T\mathbf{X}\mathbf{w}\mathbf{w}^T - \mathbf{w}_1\mathbf{w}^T\mathbf{X}^T\mathbf{X} + \mathbf{w}\mathbf{w}^T\mathbf{X}^T\mathbf{X}\mathbf{w}\mathbf{w}^T\right) \\ &= \mathrm{tr}\left(\mathbf{X}^T\mathbf{X} + \mathbf{w}\mathbf{w}^T\mathbf{w}\mathbf{w}^T\mathbf{X}^T\mathbf{X}\right) - 2\mathrm{tr}\left(\mathbf{w}\mathbf{w}^T\mathbf{X}^T\mathbf{X}\right) \\ &= \mathrm{tr}\left(\mathbf{X}^T\mathbf{X}\right) - \mathrm{tr}\left(\mathbf{w}^T\mathbf{\Sigma}\mathbf{w}\right). \end{aligned}$$

Therefore, it is obvious that the reconstruction error minimization problem is equivalent to the variance maximization problem, through which the proof is completed. (5')

Problem 6. (10 + 10 points) *(Ensemble Learning and Model Selection)*

1) In ensemble learning, suppose there are L independent two-class classifiers used for simple voting and the output of classifier j ($j = 1, \dots, L$) is denoted as d_j. From the point of view that the mean squared error of an estimator can be decomposed into the bias part and the variance part, explain why increasing L can lead to the increase of the classification accuracy. (10 points)
2) **(Bonus Question)** Suppose we carry out a K-fold cross-validation on a dataset and obtain the classification error rates $\{p_i\}_{i=1}^{K}$, describe the steps of a one-sided t test on testing the null hypothesis H_0 that the classifier has error percentage p_0 or less at a significance level α. (10 points)

Solution:

1) Let $\mathbb{E}\left[d_j\right]$ and $\operatorname{Var}\left(d_j\right)$ are the expected value and variance of d_j for classifier j, where d_j is the output of the jth classifier. Expected value and variance of output for independent classifiers:

$$\mathbb{E}[y] = \mathbb{E}\left[\sum_j \frac{1}{L} d_j\right] \geq \frac{1}{L} L \min_j \left\{\mathbb{E}\left[d_j\right]\right\} = \min_j \left\{\mathbb{E}\left[d_j\right]\right\} \ (2')$$

$$\operatorname{Var}(y) = \operatorname{Var}\left(\sum_j \frac{1}{L} d_j\right) = \frac{1}{L^2} \operatorname{Var}\left(\sum_j d_j\right) \leq \frac{1}{L^2} L \max_j \left\{\operatorname{Var}\left(d_j\right)\right\} = \frac{1}{L} \max_j \left\{\operatorname{Var}\left(d_j\right)\right\} \ (2')$$

As L increases, the expected value (and hence the bias) does not change $(3')$ but the variance decreases $(3')$, and hence the mean squared error of the estimator y decreases, leading to an increase in accuracy.

2) Let

$$m = \frac{\sum_{i=1}^{K} p_i}{K} \ (2'), \quad S^2 = \frac{\sum_{i=1}^{K} \left(p_i - m\right)^2}{K-1} \ (2'),$$

we have

$$\sqrt{K} \frac{(m - p_0)}{S} \sim \tau_{K-1} (2')$$

and hence we will fail to reject at significance level α if

$$\sqrt{K} \frac{m - p_0}{S} \in (-\infty, t_{\alpha, K-1}) . (4')$$