Introduction to Machine Learning

Fall 2025

University of Science and Technology of China

Lecturer: Jie Wang	Homework 7
Posted: Jan 4, 2026	Due: Jan 14, 2026

Notice, to get the full credits, please present your solutions step by step.

Exercise 1: Principal Component Analysis

Suppose that we have a set of data instances $\{\mathbf{x}_i\}_{i=1}^n \subset \mathbb{R}^d$. Let $\mathbf{g}_k \in \mathbb{R}^d$, $k = 1, \dots, K$, with $K \leq d$, be a set of orthonormal vectors such that

$$\langle \mathbf{g}_i, \mathbf{g}_j \rangle = \begin{cases} 1, & i = j; \\ 0, & \text{otherwise}, \end{cases}$$

and

$$G = (\mathbf{g}_1, \dots, \mathbf{g}_K).$$

From the lecture, we know that the main purpose of PCA is to find a set of orthonormal vectors $\{\mathbf{g}_1, \mathbf{g}_2, \dots \mathbf{g}_K\}$ that maximizes the **sample variance**, and finally, the problem becomes:

$$\begin{aligned} \max_{G \in \mathbb{R}^{d \times K}} \ & \operatorname{tr}(G^\top S G), \\ \text{s.t.}\, & G^\top G = I, \end{aligned} \tag{1}$$

where

$$S = \frac{1}{n-1} \sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top. \tag{2}$$

1. The projection of the $\mathbf{x}_i$ into the subspace spanned by $\{\mathbf{g}_1, \dots, \mathbf{g}_K\}$, that is, the column space of G, is

   $$\mathbf{z}_i = P_G(\mathbf{x}_i) = G G^\top \mathbf{x}_i.$$

   Show that if $\mathbf{x}_i$ is in the subspace spanned by $\{\mathbf{g}_1, \dots, \mathbf{g}_K\}$, we have:

   $$\mathbf{z}_i = G G^\top \mathbf{x}_i = \mathbf{x}_i$$

2. Show that the problem (1) always admits a solution but it may not be unique.

We use a different method from that given in lecture to solve the problem. Notice it is hard to optimize the problem directly since the feasible set is a matrix space. We could find the vectors $\mathbf{g}_k, k = 1, \dots, K$ step by step. Assume that $\lambda_1 > \lambda_2 > \cdots \geq 0$, where $\lambda_i, i = 1, 2, \dots, K$ are the eigenvalues of S.

4. Please find $\mathbf{g}_1$ defined as follows by the Lagrange multiplier method.

$$\mathbf{g}_1 := \underset{\mathbf{g}\in\mathbb{R}^d}{\mathbf{argmax}}\{\mathbf{g}^\top S\mathbf{g} : \|\mathbf{g}\|_2 = 1\}. \tag{3}$$

   Notice that, the vector $\mathbf{g}_1$ is the first principal component vector of the data.

5. Please find $\mathbf{g}_2$ defined as follows by the Lagrange multiplier method.

$$\mathbf{g}_2 := \underset{\mathbf{g}\in\mathbb{R}^d}{\mathbf{argmax}}\{\mathbf{g}^\top S\mathbf{g} : \|\mathbf{g}\|_2 = 1, \langle \mathbf{g}, \mathbf{g}_1\rangle = 0\},$$

   where $\mathbf{g}_1$ is given by (3). Similar to $\mathbf{g}_1$, the vector $\mathbf{g}_2$ is the second principal component vector of the data.

6. Please derive the first K principal component vectors by repeating the above process.

Solution:

■

Exercise 2: An Alternative Approach to Principal Component Analysis via Reconstruction Error

In the class, we derive PCA from the perspective of **projecting the data instances into a subspace such that the set of projected data instances preserves as much information**—**measured by the projection variance**—**as possible.** Thus, we call this approach to PCA as **maximization of projection variance**.

Indeed, there is another equivalent approach to PCA, which is called **minimization of reconstruction error**. The idea of this approach is to **look for a subspace such that the projections of the data instances into this subspace best approximate the data instances**. The metric to measure the approximation is the sum of the squared differences between the data instances and the corresponding projections.

1. Please derive PCA based on the idea of **minimization of reconstruction error** as introduced above. Please show your derivation step by step as what we did in the class. You can use the notations we introduced in the class.

2. Please solve the PCA you derived in the first part, and show that it is equivalent to the one we introduced in the class.

Solution: ■

Exercise 3: Principle Component Analysis. (Programming Exercise)

You are given 180 hand-drawn sketches from the TU-Berlin Sketch dataset, divided equally into three categories: guitar, tree, and tomato (60 images per category). Each image is grayscale and of size 64×64, which can be represented as a data point in $\mathbb{R}^d$ with $d = 4096$ (by flattening the 64×64 pixel grid into a vector). For each category separately, let $\{\mathbf{x}_i\}_{i=1}^{60}$ denote the dataset, and define the centered data matrix $\widetilde{X} \in \mathbb{R}^{d\times 60}$ whose i-th column is $\mathbf{x}_i - \bar{\mathbf{x}}$, where $\bar{\mathbf{x}}$ is the sample mean. Please complete the following tasks:

1. **Data Preprocessing**: The image has been flattened into a vector $\mathbf{x}_i \in \mathbb{R}^{4096}$ ($i = 1, \dots, 60$). Please first compute the sample mean $\bar{\mathbf{x}} = \frac{1}{60}\sum_{i=1}^{60} \mathbf{x}_i$, and then create the centered data matrix $\widetilde{X} \in \mathbb{R}^{4096\times 60}$ whose i-th column is $\mathbf{x}_i - \bar{\mathbf{x}}$.

2. **Principal component computation**: Compute the singular value decomposition (SVD) of X:

$$\widetilde{X} = U\Sigma V^\top.$$

Let $\mathbf{u}_1$ and $\mathbf{u}_2$ be the first two columns of U(corresponding to the two largest singular values). Reshape $\bar{\mathbf{x}}$, $\mathbf{u}_1$, and $\mathbf{u}_2$ into 64×64 images and display them. Additionally, plot the singular values: create a line or stem plot with the horizontal axis showing the component index $(1, 2, 3, \dots)$ and the vertical axis showing the corresponding singular values $\sigma_1, \sigma_2, \sigma_3, \dots$ (i.e., the diagonal entries of Σ).

3. **Analysis**: For each category, interpret the visual patterns in the displayed mean image and the first two principal components:

   - Describe what visual structure the mean image captures.
   - Explain what type of shape variation is represented by the first principal component (PC1), and why it corresponds to the direction of maximum variance.
   - Explain what kind of secondary variation is captured by the second principal component (PC2), and how its orthogonality to PC1 influences its interpretation.

You may use any programming language (Python recommended).

Solution: ■

Exercise 4: Properties of Transition Matrix

A transition matrix (also called a stochastic matrix, probability matrix) is a square matrix used to describe the transitions of a Markov chain. Each of its entries is a nonnegative real number representing a probability. A right (left) transition matrix is a square matrix with each row (column) summing to one. Without loss of generality, we study the right transition matrix in this exercise. Suppose that $\mathbf{T} \in \mathbb{R}^{n\times n}$ is a right transition matrix.

1. Show that 1 is an eigenvalue of $\mathbf{T}$.
2. Let λ be an eigenvalue of $\mathbf{T}$. Show that $|\lambda| \leq 1$.
3. Show that $\mathbf{I} - \gamma\mathbf{T}$ is invertible, where $\mathbf{I} \in \mathbb{R}^{n\times n}$ is the identity matrix and $\gamma \in (0, 1)$.

Solution:

■

Exercise 5: Planning with a Two-Armed Bandit

Consider a two-armed bandit with two states as shown in Figure 1. A player can either pull the Bandit 1 or Bandit 2 trigger, and the bandit will dispense coins and transit its state according to the following rules.

- **At State 1**, only Bandit 1 dispenses 1 coin. Pulling Bandit 1 does not cause a state transition, and pulling Bandit 2 has a $p_1 = 0.5$ probability of transitioning to State 2.
- **At State 2**, Bandit 1 dispenses 2 coins, and Bandit 2 dispenses 3 coins. Pulling Bandit 1 does not cause a state transition, and pulling Bandit 2 has a $p_2 = 0.7$ probability of transiting to State 1.

Now assume that the reward equals the number of coins dispensed, and the player can play the bandit infinite times.

1. Please find the state space $\mathcal{S}$, the action space $\mathcal{A}$, and the transition function $P(s'|s, a)$ of the two-armed bandit, and draw the Markov process diagram.
2. Let $\gamma = 0.9$. Please find the state value functions $V^{\pi}(s)$ for the given policy $\pi(a|s)$:
   (a) π_1: Always pull the Bandit 2.
   (b) π_2: Pull Bandit 2 at State 1, and pull Bandit 1 at State 2.
3. For the cases where $\gamma = 0.1$ and $\gamma = 0.99$, please find the optimal policy π^* and its state value function $V^{\pi^*}(s)$. Please explain the effect of the value of γ based on the results.

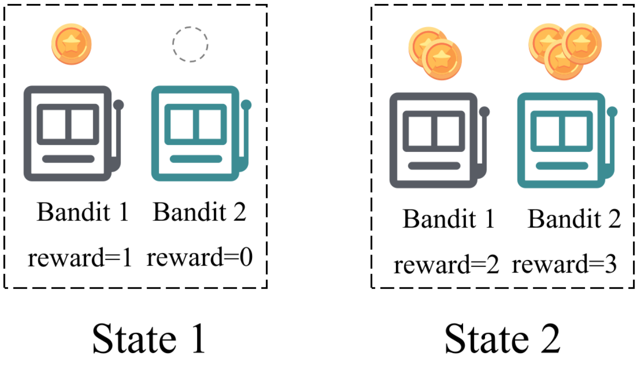


Figure 1: Illustration of the two armed-bandit.

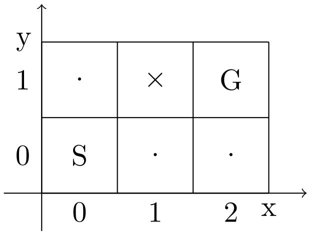
Solution:

■

Exercise 6: Value Iteration and Policy Iteration in Grid World

Consider a simplified 3×2 deterministic grid world where an agent needs to find the optimal path to a goal. The agent starts at $(0,0)$ and the goal is at $(2,1)$. There is an obstacle at $(1,1)$. The agent can take actions $\mathcal{A} = \{\text{up}, \text{down}, \text{left}, \text{right}\}$. If an action would move the agent out of the grid or into the obstacle, the agent stays in its current position. The immediate reward is -1 for all non-goal states and $+10$ for reaching the goal. The discount factor is $\gamma = 0.9$.

The grid layout is shown below (G: goal, ×: obstacle, S: start):



1. Please find the state space $\mathcal{S}$, the action space $\mathcal{A}$, the reward function $r(s,a)$, and the deterministic transition function.

2. **Value Iteration:** Starting from initial value function $V_0(s) = 0$ for all $s \in \mathcal{S}$, manually compute the value functions $V_1(s)$ and $V_2(s)$ after the first two iterations. For each iteration, show the Q-value calculations for all states.

3. **Policy Iteration:** Let the initial policy π_0 be: at every state, choose Right if legal (the agent will not move out of the grid or into the obstacle); otherwise choose UP if legal; otherwise choose DOWN. Perform **one complete iteration** of policy iteration:

   (a) **Policy Evaluation:** Solve the system of linear equations to compute $V^{\pi_0}(s)$ for all states.

   (b) **Policy Improvement:** Using V^{π_0}, compute the improved policy $\pi_1(s)$ for all states.

4. Compare the results after two iterations of value iteration and one iteration of policy iteration.

   (a) Discuss whether the observed performance difference is solely due to the choice of the initial policy π_0.

   (b) Beyond the initial policy, what are the fundamental algorithmic reasons that policy iteration typically requires fewer iterations to converge than value iteration

Solution:

■