Machine Learning

Instructors: Mohammadreza A. Dehaqani, Babak n. Arabi

Shahzad Momayez, Roshak Masoudi Nejad

Fall 2025

Homework 1

Question 1: Minimum Risk Classification with Rejection Option

In many pattern classification problems we have the option to either assign a pattern to one of c classes, or reject it as unrecognizable—if the cost to reject is not too high. Let the cost of classification be defined as:

$$\lambda(\omega_i|\omega_j) = \begin{cases} 0, & \text{if } \omega_i = \omega_j, \quad \text{(i.e., Correct Classification)} \\ \lambda_r, & \text{if } \omega_i = \omega_0, \quad \text{(i.e., Rejection)} \\ \lambda_s, & \text{Otherwise, (i.e., Substitution Error)} \end{cases}$$

Show that for the minimum risk classification, the decision rule should associate a test vector x with class ω_i, if

$$P(\omega_i|x) \geq P(\omega_j|x) \quad \forall j \quad \text{and} \quad P(\omega_i|x) \geq 1 - \frac{\lambda_r}{\lambda_s},$$

and reject otherwise.

Question 2: Naive Bayes

Consider a **Naive Bayes** classification problem with 3 classes and 2 features. One of these features follows a Bernoulli distribution, and the other follows a Gaussian distribution. The features are denoted by $X = [X_1, X_2]^T$, and the class by Y.

Prior distribution

$$P[Y = 0] = 0.5, \quad P[Y = 1] = 0.25, \quad P[Y = 2] = 0.25$$

Feature distributions

$$p_{X_1|Y}(x_1 \mid Y = c) = Ber(x_1; \theta_c)$$
$$p_{X_2|Y}(x_2 \mid Y = c) = Normal(x_2; \mu_c, \sigma_c^2)$$

Parameter assumptions

$$\sigma_c^2 = \begin{cases} 1 & \text{if } c = 0 \\ 1 & \text{if } c = 1 \\ 1 & \text{if } c = 2 \end{cases} \quad \mu_c = \begin{cases} -1 & \text{if } c = 0 \\ 0 & \text{if } c = 1 \\ 1 & \text{if } c = 2 \end{cases} \quad \theta_c = \begin{cases} 0.5 & \text{if } c = 0 \\ 0.75 & \text{if } c = 1 \\ 0.5 & \text{if } c = 2 \end{cases}$$

Questions (a) Compute $p_{Y|X_1,X_2}(y \mid x_1 = 0, x_2 = 0)$. (The answer should be a vector in $\mathbb{R}^3$ whose elements sum to 1.)

(b) Compute $p_{Y|X_1}(y \mid x_1 = 0)$.

(c) Compute $p_{Y|X_2}(y \mid x_2 = 1)$.

(d) Analyze the pattern observed in the results from the previous parts.

Question 3: Multinomial Naive Bayes

Consider a Multinomial Naive Bayes model for a two-class text classification problem. Assume that the total number of words in our dictionary (the total number of model features) is d. For a text input sample x, the values $c_1, c_2, \ldots, c_p$ form the feature vector. That is, each c_i represents the number of times the i-th word appears in the document. The parameters of this model are as follows: (y is the model output or the class label of the sample.)

$$P_y = P(y = 1)$$

$$P_{i|y=1} = P(\text{word } i \text{ appears in a specific document position} \mid y = 1)$$

$$P_{i|y=.} = P(\text{word } i \text{ appears in a specific document position} \mid y = .)$$

(a) Write an expression for the conditional probability $P(y = 1 \mid x)$ for a text sample x in terms of the model parameters.

(b) Show that the decision boundary of the trained model is linear.

(c) Show that the conditional probability written in part (a) is a logistic function:

$$P(y = 1 \mid x) = \frac{1}{1 + e^{-(\theta^T x + \theta_0)}}$$

Question 4: MLE

(a) Let $X_1, X_2, ..., X_N$ be data drawn independently from a uniform distribution over a diamond-shaped area with edge length $\sqrt{2}\theta$ in $\mathbb{R}^2$, where $\theta \in \mathbb{R}^+$ (see Figure 1). Thus, $X_i \in \mathbb{R}^2$ and the distribution is

$$p(x \mid \theta) = \begin{cases} \frac{1}{2\theta^2}, & \text{if } \|x\|_1 \leq \theta, \\ 0, & \text{otherwise,} \end{cases}$$

where $\|x\|_1 = |x_1| + |x_2|$ is the L_1 norm. Find the maximum likelihood estimate of θ.

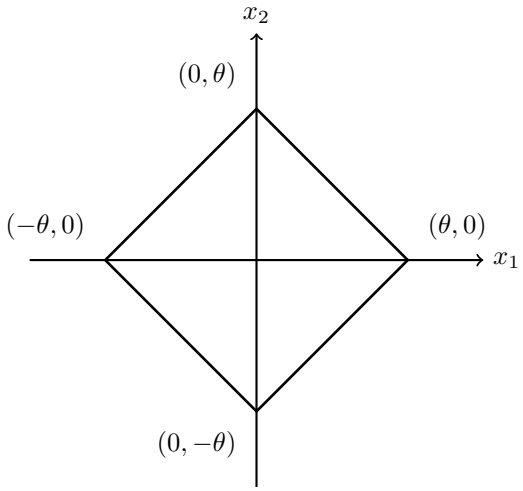

Figure 1: Area $\|x\|_1 \leq \theta$

(b) Let $X_1, X_2, ..., X_n$ be a random sample from a Gamma(α, β) population. Find the maximum likelihood estimate (MLE) of β, assuming α is known.

Question 5: MAP

MAP of Multinational Distribution

You have just got a loaded 6-sided dice from your statistician friend. Unfortunately, he does not remember its exact probability distribution $p_1, p_2, \ldots, p_6$. He remembers, however, that he generated the vector $(p_1, p_2, \ldots, p_6)$ from the following Dirichlet distribution:

$$\mathbb{P}(p_1, p_2, \ldots, p_6) = \frac{\Gamma\left(\sum_{i=1}^{6} u_i\right)}{\prod_{i=1}^{6} \Gamma(u_i)} \prod_{i=1}^{6} p_i^{u_i - 1} \, \delta\left(\sum_{i=1}^{6} p_i - 1\right),$$

where he chose $u_i = i$ for all $i = 1, \ldots, 6$. Here Γ denotes the gamma function, and δ is the Dirac delta function. To estimate the probabilities $p_1, p_2, \ldots, p_6$, you roll the dice 1000 times and then observe that side i occurred n_i times, where

$$\sum_{i=1}^{6} n_i = 1000.$$

(a) Conjugacy of the Dirichlet Prior:
Prove that the Dirichlet distribution is a conjugate prior for the multinomial distribution.
(b) Posterior Distribution:
Given the data $\{n_1, n_2, \ldots, n_6\}$, find the posterior distribution of the side probabilities:

$$\mathbb{P}(p_1, p_2, \ldots, p_6 \mid n_1, n_2, \ldots, n_6).$$

Question 6: Gaussian Discriminant Analysis

(Computer Assignment 1)

Use the following code to generate a dataset:

```
from sklearn import datasets
from sklearn.decomposition import PCA
iris = datasets.load_iris()
Xorig = iris.data
y = iris.target
pca = PCA(n_components=2)
X = pca.fit(Xorig).transform(Xorig)
plt.scatter(X[:, 0], X[:, 1], marker='o', c=y, edgecolor='k')
```

(a) Use the data for classes 1 and 2 to estimate the parameters of a bivariate Gaussian distribution for each class. Assume that the covariance matrix is the same for both classes. Write the parameters of the probability distribution functions for both classes.

(b) Write a Python function that calculates the discriminant function for each class.

(c) Draw a plot, where the regions corresponding to the different classes are shown with different colors. A region corresponding to a class is the set of points where the particular class discriminant function is maximum.

(d) The boundary between both class regions must be a line. Calculate the equation of this line clearly explaining the deduction process. Draw the line along with the regions.

(e) What happens with the boundary line if we change the prior probabilities of the classes? Illustrate with a graphical example.

(f) Repeat steps (a) to (c) from the previous item, but this time:

Use data from the three classes.

Estimate a different covariance matrix for each class.

(i) Repeat the previous item, but this time:

Use only a portion of the dataset (80% of the samples) to estimate the parameters of the probability distribution functions of each class.

Write a function that calculates the discriminant function for each class, taking into account the possibility of rejection with a cost λ and cost 1 for misclassification). Look for values of λ that produce a rejection region easily distinguishable from the other regions.

Question 7: Gaussian Naive Bayes

(Computer Assignment 2)

The data set for this question is in the guassianBayes.zip file.
In this question, you will implement the Gaussian Naive Bayes classification algorithm. As a reminder, in the Naive Bayes algorithm we calculate:

$$p(c \mid \mathbf{f}) \propto p(c) \prod_i p(f_i \mid c).$$

In Gaussian Naive Bayes, we learn a one-dimensional Gaussian for each feature in each class, i.e.

$$p(f_i \mid c) = \mathcal{N}(f_i; \mu_{i,c}, \sigma^2_{i,c}),$$

where $\mu_{i,c}$ is the mean of feature f_i for those instances in class c, and $\sigma^2_{i,c}$ is the variance of feature f_i for instances in class c. You can (and should) test your implementation locally using the provided `xTrain` and `yTrain` data.

(a) Training Model — Learning Class Priors

Complete the function `[p] = prior(yTrain)`. p is a $c \times 1$ vector where p_i is the prior probability of class i.

(b) Training Model — Learning Class-Conditional Feature Probabilities

Complete the function `[M, V] = likelihood(xTrain, yTrain)`. M is an $m \times c$ matrix where $M_{i,j}$ is the conditional mean of feature i given class j. V is an $m \times c$ matrix where $V_{i,j}$ is the conditional variance of feature i given class j.

(c) Naive Bayes Classifier

Complete the function `[t] = naiveBayesClassify(xTest, M, V, p)`. t is an $m \times 1$ vector of predicted class values, where t_i is the predicted class for the ith row of `xTest`.

(d) Evaluation

Let's analyze the accuracy of the classifier on the test data. Create a text file named `evaluation.txt`. On each line, report one of the following metrics in decimal format, rounded to three decimal places:

- Fraction of test samples classified correctly
- Precision for class 1
- Recall for class 1
- Precision for class 5
- Recall for class 5

Question 8: Bag of Words — Naive Bayes Classifier

(Computer Assignment 3)

In this problem, you will code up a Naive Bayes classifier to perform "Bag of Words"-style classification of text documents. Our goal is to train a classifier that can accurately predict whether newsgroup posts come from a newsgroup about baseball or (ice) hockey. Each post is represented as a vector of word counts, where each entry corresponds to how many times a given word (from a shared vocabulary) appears in the post.
The provided ZIP file (bagOfWords.zip) contains: `vocabulary.txt`, `baseball_train_set.csv`, `baseball_test_set.csv`, `hockey_train_set.csv`, `hockey_test_set.csv`. Each CSV file has 50 lines (posts) and 5822 columns (distinct words). Each value represents the number of times a word appears in that post.

(a) Naive Bayes Classifier — Maximum Likelihood Estimation

Code up a Bag of Words-style Naive Bayes classifier to handle the provided data. Your classifier should take as input:

- training data for baseball and hockey,
- test data for baseball and hockey,

and it should output predictions for each post in the test set (either "baseball" or "hockey").
Train your classifier using all of the training set (50 posts each from baseball and hockey) and have it classify all of the test set (50 posts each from baseball and hockey). Report, in writing, the answers to the following questions:

- What is your test set accuracy?
- What did your classifier end up predicting?
- Why is using the MLE estimate a bad idea in this situation?

Requirements:

- Use the Maximum Likelihood Estimate (MLE) for all parameters in your classifier.
- Define $0 \times \log(0) = 0$ wherever it occurs.
- Assume equal class priors: $P(Y = \text{“baseball”}) = P(Y = \text{“hockey”})$.
- In case of ties, predict “baseball.”
- Keep your code under 100 lines.

(b) Naive Bayes Classifier — MAP Estimation with Dirichlet Prior
Extend your classifier so that it can compute a Maximum A Posteriori (MAP) estimate using a symmetric Dirichlet prior. The prior assumes each word is “seen” β times in the training data (“hallucinated counts”). Formally, all Dirichlet parameters are equal to β. Modify your classifier to take β as an additional parameter and use it when estimating word likelihoods.

(c) Effect of β on Classification Accuracy
Train your classifier using all available training data (50 posts per class) and test it on all available test data (50 posts per class). Run your classifier for the following β values:

$$\beta \in \{0, 10^{-8}, 10^{-6}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 2, 5, 10\}.$$

Plot β (x-axis) vs. classification accuracy (y-axis). Briefly describe how accuracy changes with β and discuss whether large β values are beneficial or harmful.

(d) Effect of Training Set Size
Fix $\beta = 0.1$. Train your classifier using the first n posts from each training set (for a total of $2n$ training examples) for:

$$n \in \{1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50\}.$$

Test your classifier on the entire test set (50 baseball + 50 hockey posts). Plot n (x-axis) vs. classification accuracy (y-axis).

Implementation notes / suggestions:

- Preprocess each CSV row as an integer vector of length 5822 (word counts).
- For MLE: estimate word likelihood for class c by $\hat{p}_{w|c} = \frac{\text{count}_{w,c}}{\sum_{w'} \text{count}_{w',c}}$.
- For MAP with Dirichlet(β): use $\hat{p}_{w|c} = \frac{\text{count}_{w,c}+\beta}{\sum_{w'}(\text{count}_{w',c}+\beta)}$.
- Compute log-probabilities for numerical stability and sum log-likelihoods across words: $\log P(\mathbf{x} \mid c) = \sum_w x_w \log \hat{p}_{w|c}$.
- Use equal log-priors; break ties by selecting “baseball.”
- When plotting, use a log-scale for β on the x-axis for clarity (especially when values span many orders of magnitude).