Homework 4 - Applied Stochastic Processes

kipngeno koech - bkoech

November 2, 2024

Estimation, Mixture Models and Random Processes with Python Simulations

1 MoM, MLE, Bias and Consistency (20 points)

Consider a normal distribution defined by the probability density function (PDF):

$$f(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \quad -\infty < x < \infty \tag{1}$$

where μ is the mean and σ^2 is the variance. Given a random sample $Y = \{y_1, y_2, \ldots, y_n\}$ drawn from this normal distribution, perform the following tasks:

1. (**5 points**) Use the method of moments to derive the estimators for μ and σ^2.
2. (**5 points**) Derive the Maximum Likelihood Estimators (MLE) for μ and σ^2.
3. (**3 points**) Calculate the bias of the MoM estimators μ_{MoM} and σ^2_{MoM}
4. (**3 points**) Calculate the bias of the MLE estimators μ_{MLE} and σ^2_{MLE}
5. (**4 points**) Show that both the MoM and MLE estimators are consistent, meaning that $n \to \infty$, then $\mu_{MoM} \to \mu$ and $\sigma^2_{MoM} \to \sigma^2$ in probability.

2 Spam-Ham Detection Using MLE and MAP (30 points)

In digital communication, distinguishing spam from ham (non-spam) is crucial for email security. Sta- tistical techniques such as Maximum Likelihood Estimation (MLE) and Maximum A Posteriori (MAP) estimation are effective for classification. This section aims to build a spam-ham classifier using both MLE and MAP methods.

2.1 MLE/MAP on Toy Dataset (10 points)

You are provided with a mini dataset containing six SMS messages labeled as either spam or ham. A single feature, "offer," indicates the presence (1) or absence (0) of the word "offer" in each message. The dataset is shown in Table 1. Calculate the following MLE estimates:

Message ID	**Message content**	**Offer (X)**	**class (Y)**
1	Special offer now!	1	1 (Spam)
2	Meeting at 10 AM	0	0 (Ham)
3	Claim your offer	1	1 (Spam)
4	Lunch tomorrow?	0	0 (Ham)
5	Free offer available	1	1 (Spam)
6	Hello, how are you?	0	0 (Ham)

Table 1: Mini Dataset for Spam-Ham Detection

1. (**1 point**) $\pi = P(Y = 1)$. Probability that a message is spam.
2. (**1 points**) $\theta_{spam} = P(X = 1|Y = 1)$.The probability of the word "offer" appearing in a spam message
3. (**1 points**) $\theta_{ham} = P(X = 1|Y = 0)$. The probability of the word "offer" appearing in a ham message
4. (**2 points**) Derive the likelyhood function and maximize it to find the parameter estimates. Assume Beta priors:
   (a) (**2 points**) $\pi \sim \beta(2, 2)$
   (b) (**2 points**) $\theta_{spam} \sim \beta(2, 1)$
   (c) (**2 points**) $\theta_{ham} \sim \beta(1, 2)$

   Calculate the MAP estimates for π, θ_{spam} and θ_{ham} using prior information.

2.2 Practical Implications (4 points)

Discuss the following:

1. (**2 points**) How do different prior distributions affect the MAP estimates?
2. (**1 points**) Why might MLE overfit with small datasets?
3. (**1 points**) In what scenarios would MLE or MAP perform better?
4. (**1 points**) What is the bias-variance trade-off between MLE and MAP?

2.3 Real-World Implementation (10 points)

In this exercise, you will classify messages as either "spam" or "ham" (not spam) using a Naive Bayes classifier. You will implement two different estimation methods. Use the "SMS Spam Collection" dataset, available at this link, to implement a spam-ham detection classifier using:

- **Maximum Likelihood Estimation (MLE)**: Estimates the parameters based solely on the training data without prior beliefs about the parameters.
- **Maximum A Posteriori (MAP):**: Incorporates prior beliefs about the parameters into the estimation, using Laplace smoothing to handle zero probabilities.

Tasks: Data Loading and Preprocessing:

1. **Load the dataset:**
   - Download the SMS Spam Collection dataset and load it into your environment using pandas.
   - Ensure the dataset is read correctly, with columns labeled "label" for spam/ham and "message" for the text content.
2. **Preprocess the Text Messages:**
   - Convert all text to lowercase.
   - Remove punctuation and special characters.
   - Tokenize the messages (split the messages into words).
3. **Split the Dataset:**
   - Divide the dataset into training and test sets (e.g., 80% training, 20% testing).
   - Ensure that both sets maintain the same class distribution.

Implement Maximum Likelihood Estimator (MLE):

1. **Calculate the Probabilities:**
   - For each class (spam and ham), calculate the probability of each word appearing in that class based on the training data.
2. **Implement Prediction Function:**
   - Create a function to classify messages using the calculated probabilities and the prior probabilities of each class.
3. **Evaluate the Classifier:**
   - Use metrics such as accuracy, precision, recall, and F1 score to evaluate the performance of the MLE classifier on the test set.

Implement Maximum A Posteriori (MAP):

1. **Implement MAP Estimator:**
   - Calculate the same probabilities as in MLE but include Laplace smoothing to avoid zero probabilities.
2. **Implement Prediction Function:**
   - Create a prediction function similar to MLE but using the MAP probabilities.
3. **Evaluate the Classifier:**
   - Again, use accuracy, precision, recall, and F1 score to evaluate the MAP classifier's perfor- mance on the test set.

Compare Results:

1. **Performance Comparison:**
   - Create a comparison table that summarizes the accuracy, precision, recall, and F1 score for both classifiers.
2. **Discuss the Results:**
   - Reflect on the differences in performance:
     - How did incorporating prior knowledge in MAP affect the predictions?
     - Were there any significant changes in the classification of messages between MLE and MAP?
     - What factors might account for any differences in the performance metrics?

Vary the Prior (MAP):

1. **Experiment with Different alpha values:**
   - Run the MAP classifier with varying values of the Laplace smoothing parameter (alpha) such as 0.1, 0.5, 1, and 5
   - Observe how these variations affect the results and the evaluation metrics.
2. **Discussion of Findings:**
   - Summarize your observations regarding the impact of varying the prior on the classification performance.

Deliverables: