CSCI 57800 ML Fall 2023 Homework 3

October 30, 2023

Instructions

We will be using Canvas to collect your assignments. Please read the following instructions to prepare your submission.

1. Submit your solution in a pdf file and a zip file (<yourLastName_FirstName>.pdf/zip). Your write-up must be in pdf. Your code must be in the zip file.
2. In your pdf file, the solution to each problem should start on a new page.
3. Latex is strongly encouraged to write your solutions, e.g., using Overleaf (https://www.overleaf.com/). Neither scanned handwritten copies nor hard copies are acceptable.
4. You need to add screen captures of your code and the output in your write-up.
5. You may discuss the problems and potential directions for solving them with another student. However, you need to write your own solutions and code separately, and not as a group activity. Please list the students you collaborated with on your submission.

Problem 1 (10 points)

Suppose we have a dataset of the form $[(x_1, y_1)...(x_n, y_n)]$ where $x_i \in \mathbb{R}$ and $y \in \mathbb{R}$. We will learn a linear function of x of the form $\hat{y} = \theta x$ where $\theta \in \mathbb{R}^d$. If the loss function is the absolute difference $|y - \hat{y}|$, write an expression for $F(\theta)$, the total loss over the entire dataset.

Problem 2 (20 points)

Consider the following loss function on vectors $w \in \mathbb{R}^4$:
$L(w) = w_1^2 + 2w_2^2 + w_3^2 - 2w_3w_4 + w_4^2 + 2w_1 - 4w_2 + 4$

(a) (10 pts) What is $\nabla L(w)$?

(b) (10 pts) Suppose we use gradient descent to minimize this function and that the current estimate is (0, 0, 0, 0). If the step size is η, what is the next estimate?

Problem 3 (10 points)

Compute $\frac{dJ}{dx}$ for our 2-hidden layer neural network (slide 23 in Lecture 14) using backpropagation. Show the forward pass and backward pass as in slide 21 in Lecture 14.

Problem 4 (60 points, Programming Involved)

You will implement a Naive Bayes classifier. Note that, using the Bayes rule, your class prediction is $\hat{y} = \text{argmax}_y P(X|y)P(y)$, where the class y is the Label "Y" or "N".

Since we are using the Naive Bayes model, the following holds: $P(X|y) = \prod_{x_i \in X} P(x_i|y)$.

We will ignore smoothing for thie problem.

The table below shows your training data.

W1	W2	Label
blue	gloves	Y
blue	hat	Y
cautious	cat	Y
new	hat	Y
cautious	cat	Y
cautious	cat	N
cautious	hat	N
new	gloves	N
new	cat	N

(a) (10 pts) Write code for reading the data. Attach the screenshot of your code here.

(b) (10 pts) Write a function `estimate_cond_probs(feature_values, labels)` that takes `feature_values` which contains all training data, and `labels` for that feature, the array of corresponding labels.

The function should return a pair of tuples, $(P_{i|Y}, P_{i|N})$, where, for feature i, $P_{i|Y}$ provides the estimated values of the parameters for class Y, and $P_{i|N}$ provides the estimated parameter values for the distribution of class N.

You must use Maximum Likelihood Estimation to estimate the parameters. Your probability estimation will be presented as a tuple of size n such that $P_{i|y}[j] = P(x_i = j|y)$ where j is the j-th possible value of the feature.

Attach the screenshot of your code here.

(c) (5 pts) Run the function `estimate_cond_probs(feature_values, labels)` on the attribute $W1$ in the table above. What is the log-probability that $W1 = blue$? In other words, what are the values for the following two log-probabilities? Attach the screenshot of your output as well.

$logP_{W1|Y}[\text{"}blue\text{"}] =?$
$logP_{W1|N}[\text{"}blue\text{"}] =?$

(d) (5 pts) Run the function `estimate_cond_probs(feature_values, labels)` on the attribute $W2$ in the table above. What is the log-probability that $W2 = cat$? In other words, what are the values for the following two log-probabilities? Attach the screenshot of your output as well.

$logP_{W2|Y}[\text{"}cat\text{"}] =?$
$logP_{W2|N}[\text{"}cat\text{"}] =?$

(e) (10 pts) For the case of the class prior probability distribution, write a function `estimate_prob_Y(labels)` that takes in input the array of labels from the dataset and uses it to estimate the prior probability that any text would be Y. Attach the screenshot of your code here.

(f) (5 pts) Run your function `estimate_prob_Y(labels)` on the provided training data. What is the estimated log-probability that $P(y = Y)$?

(g) (10 pts) Write a function `classify(trained_model, X)` that takes in input:

- `trained_model`: a tuple of 3 elements, where first 2 are the attribute probability estimations computed using `estimate_cond_probs`, and the last one is the value returned from `estimate_prob_Y`.
- `X`: an unlabeled new feature vector represented as an array of 2 attribute values.

`classify(trained_model, X)` should return a tuple of three values `class`, `log_prob_Y`, and `log_prob_N`:

- `class`: a string representing the predicted class "Y" or "N".
- `log_prob_Y`: the conditional log-probability of the feature vector `X` given that the label is "Y", $P(X|y = Y)$.
- `log_prob_N`: the conditional log-probability of the feature vector `X` given that the label is "N", $P(X|y = N)$.

Attached the screenshot of your code here.

(h) (5 pts) Run your functions on the following values of `X` (after mapping them to the correct representation) and report the classification and the log-probabilities returned.

W1="cautious", W2="gloves"

Attached the screenshot of the program output.

Three bonus points will be given if your homework is easy to review.