Introduction to Machine Learning

Spring 2021

University of Science and Technology of China

Lecturer: Jie Wang | Homework 4

Posted: Apr. 25, 2020 | Due: May. 7, 2020

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

Exercise 1: Entropy and Uncertainty

[1]Consider a random variable X that can take n values. $x_1 \ldots, x_n$, with corresponding probabilities $p_1, \ldots, p_n$. The **entropy** of X is defined to be

$$H(X) = -\sum_{i=1}^{n} p_i \log p_i.$$

(All logarithms in this problem are with respect to base two.)

(a) Show that if $q_1 \ldots q_n$ are nonnegative numbers such that $\sum_{i=1}^{n} q_i = 1$, then

$$H(X) \leq -\sum_{i=1}^{n} p_i \log q_i.$$

with equality if and only if $p_i = q_i$ for all i. As a special case, show that $H(X) \leq \log n$, with equality if and only if $p_i = 1/n$ for all i.

(b) Let X and Y be random variables taking a finite number of values, and having joint PMF $p_{X,Y}(x, y)$. Define

$$I(X, Y) = \sum_{x} \sum_{y} p_{X,Y}(x, y) \log \left(\frac{p_{X,Y}(x, y)}{p_X(x) p_Y(y)} \right).$$

Show that $I(X, Y) \geq 0$, and that $I(X, Y) = 0$ if and only if X and Y are independent.

(c) Show that

$$I(X, Y) = H(X) + H(Y) - H(X, Y),$$

where

$$H(X, Y) = -\textstyle\sum_x \sum_y p_{X,Y}(x, y) \log p_{X,Y}(x, y),$$
$$H(X) = -\textstyle\sum_x p_X(x) \log p_X(x), \quad H(Y) = -\sum_y p_Y(y) \log p_Y(y).$$

(d) Show that

$$I(X, Y) = H(X) - H(X \mid Y),$$

where

$$H(X \mid Y) = -\sum_{y} p_Y(y) \sum_{x} p_{X|Y}(x \mid y) \log p_{X|Y}(x \mid y).$$

Solution: ■

Exercise 2: Decision Tree

Please build a decision tree based on the information gain to classify the following dataset (you need to show the calculation steps in detail).

Sample	A_1	A_2	A_3	Response
x_1	0	0	0	0
x_2	1	0	1	0
x_3	0	1	0	0
x_4	0	1	1	1
x_5	1	1	0	1
x_6	0	1	1	1

Table 1: Dataset

The dataset consists of six samples $x_1, x_2, x_3, x_4, x_5, x_6$. For each sample, we can observe the features A_1, A_2, A_3 and the corresponding response.

Solution: ■

Exercise 3: Programming Exercise: Naive Bayes

We provide you with a data set that contains spam and non-spam emails ("hw4_nb.zip"). Please use the Naive Bayes Classifier to detect the spam emails. Finish the following exercises by programming. You can use your favorite programming language.

1. Remove all the tokens that contain non-alphabetic characters.
2. Train the Naive Bayes Classifier on the training set according to Algorithm 1.
3. Test the Naive Bayes Classifier on the test set according to Algorithm 2.
4. Compute the confusion matrix, precision, recall, and F1 score. Please report your result.

Algorithm 1 Training Naive Bayes Classifier

Input: The training set with the labels $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}$.

$\mathcal{V} \leftarrow$ the set of distinct words and other tokens found in $\mathcal{D}$
for each target value c in the labels set $\mathcal{C}$ **do**
  $\mathcal{D}_c \leftarrow$ the training samples whose labels are c
  $P(c) \leftarrow \frac{|\mathcal{D}_c|}{|\mathcal{D}|}$
  $T_c \leftarrow$ a single document by concatenating all training samples in $\mathcal{D}_c$
  $n_c \leftarrow |T_c|$
  for each word w_k in the vocabulary $\mathcal{V}$ **do**
    $n_{c,k} \leftarrow$ the number of times the word w_k occurs in T_c
    $P(w_k|c) = \frac{n_{c,k}+1}{n_c+|\mathcal{V}|}$
  end for
end for

Algorithm 2 Testing Naive Bayes Classifier

Input: An email $\mathbf{x}$. Let x_i be the i^{th} token in $\mathbf{x}$. $\mathcal{I} = \emptyset$.

for $i = 1, \ldots, |\mathbf{x}|$ **do**
  if $\exists\, w_{k_i} \in \mathcal{V}$ such that $w_{k_i} = x_i$ **then**
    $\mathcal{I} \leftarrow \mathcal{I} \cup k_i$
  end if
end for
predict the label of $\mathbf{x}$ by

$$\hat{y} = \arg\max_{c \in \mathcal{C}} P(c) \prod_{i \in \mathcal{I}} P(w_{k_i}|c)$$

Solution: ■

Exercise 4: Logistic Regression

Given the training data $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, where $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \{0, 1\}$. Let

$$\mathcal{I}^+ = \{i : i \in [n], y_i = 1\},$$
$$\mathcal{I}^- = \{i : i \in [n], y_i = 0\},$$

where $[n] = \{1, 2, \ldots, n\}$. We assume that $\mathcal{I}^+$ and $\mathcal{I}^-$ are not empty.

Then, we can formulate the logistic regression as:

$$\min_{\mathbf{w}} \; L(\mathbf{w}) = -\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log\left(\frac{\exp(\langle \mathbf{w}, \overline{\mathbf{x}}_i\rangle)}{1+\exp(\langle \mathbf{w}, \overline{\mathbf{x}}_i\rangle)}\right) + (1-y_i)\log\left(\frac{1}{1+\exp(\langle \mathbf{w}, \overline{\mathbf{x}}_i\rangle)}\right)\right), \quad (1)$$

where $\mathbf{w} \in \mathbb{R}^{d+1}$ is the model parameter to be estimated and $\overline{\mathbf{x}}_i^\top = (1, \mathbf{x}_i^\top)$.

1. Find the gradient and the Hessian of $L(\mathbf{w})$.
2. Suppose that $\overline{\mathbf{X}} = (\overline{\mathbf{x}}_1, \overline{\mathbf{x}}_2, \ldots, \overline{\mathbf{x}}_n)^\top \in \mathbb{R}^{n\times(d+1)}$ and $\mathbf{rank}(\overline{\mathbf{X}}) = d + 1$. Show that $L(\mathbf{w})$ is strictly convex, i.e., for all $\mathbf{w}_1 \neq \mathbf{w}_2$,

$$L(t\mathbf{w}_1 + (1-t)\mathbf{w}_2) < tL(\mathbf{w}_1) + (1-t)L(\mathbf{w}_2), \forall\, t \in (0, 1).$$

3. Suppose that the training data is strictly linearly separable, that is, there exists $\hat{\mathbf{w}} \in \mathbb{R}^{d+1}$ such that

$$\langle \hat{\mathbf{w}}, \overline{\mathbf{x}}_i\rangle > 0, \forall\, i \in \mathcal{I}^+,$$
$$\langle \hat{\mathbf{w}}, \overline{\mathbf{x}}_i\rangle < 0, \forall\, i \in \mathcal{I}^-.$$

   Show that problem (1) has no solution.

Let $\overline{\mathbf{z}}_i = (2y_i - 1)\overline{\mathbf{x}}_i$.

4. Show that

$$L(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n}\log(1+\exp(-\langle \mathbf{w}, \overline{\mathbf{z}}_i\rangle)).$$

5. Suppose that the training data is NOT linearly separable. That is, for all $\mathbf{w} \in \mathbb{R}^{d+1}$, there exists $i \in [n]$ such that

$$\langle \mathbf{w}, \overline{\mathbf{z}}_i\rangle < 0.$$

   Show that problem (1) always admits a solution.

Solution: ■

Exercise 5: Example in Stochastic Gradient Descent

Consider a simple linear regression model $f(x; w) = wx$ with samples $\{(x_i, y_i)\}_{i=1}^2$, where $x_i, y_i \in \mathbb{R}$ for $i = 1, 2$. We use SGD algorithm to minimize the average fitting error

$$L(w) = \frac{1}{2}\sum_{i=1}^{2}(y_i - wx_i)^2.$$

We uniformly sample a data instance $\xi_k = (x_{i_k}, y_{i_k})$ from $\{(x_i, y_i)\}_{i=1}^2$ at k^{th} iteration. The sequence (w_k) is generated by the stochastic gradient descent algorithm.

1. Please write down the derivative $L'(w_k)$ and stochastic derivative g_k.
2. Please write down the variance of the stochastic derivative $\mathbb{V}_{\xi_k}[g_k]$.
3. We assume the upper bound of $\mathbb{V}_{\xi_k}[g_k]$ takes the form of

   $$\mathbb{V}_{\xi_k}[g_k] \leq M + M_V|L'(w_k)|^2.$$

   Please find the corresponding M and M_V in this problem. Specifically, when will M become zero, and when will M_V become zero?
4. Can you explain why we cannot expect $\mathbb{V}_{\xi_k}[g_k]$ is bounded or equal to 0?

Solution: ■

Exercise 6: Law of Total Variance

Let X, Y, and Z be random variables.

1. Show that the tower property holds, i.e.,

$$\mathbb{E}[X|Y] = \mathbb{E}[\mathbb{E}[X|Y, Z]|Y].$$

2. The law of total variance holds, i.e.,

$$\mathbb{V}[X] = \mathbb{E}[\mathbb{V}[X|Y]] + \mathbb{V}[\mathbb{E}[X|Y]].$$

Hint: if you do not know measure theory well, you can assume that X, Y, and Z are continuous random variables.

Solution: ■

Exercise 7: Convergence of SGD for Convex Function

Suppose that $f : \mathbb{R}^n \to \mathbb{R}$ is convex and continuously differentiable, and it attains its minimum at $\mathbf{x}^*$. Suppose the second moment of stochastic gradient $\mathbf{g}$ is bounded, i.e.,

$$\mathbb{E}_\xi[\|\mathbf{g}(\mathbf{x}, \xi)\|_2^2] \leq G^2, \forall \mathbf{x} \in \mathbb{R}^n.$$

Suppose that $(\mathbf{x}_k)$ is a sequence generated by SGD algorithm with a fixed stepsize α. Define $\tilde{\mathbf{x}}_K = \frac{1}{K}\sum_{k=1}^K \mathbf{x}_k$ and $f^* = f(\mathbf{x}^*)$.

1. If X is a random variable and $h : \mathbb{R} \to \mathbb{R}$ is a convex function, please show that

$$h(\mathbb{E}[X]) \leq \mathbb{E}[h(X)].$$

2. Suppose that the stochastic gradient at k^{th} iteration is $\mathbf{g}_k$. Please show that

$$\mathbb{E}_{\xi_1:\xi_k}[f(\mathbf{x}_k) - f^*] \leq \mathbb{E}_{\xi_1:\xi_k}[\langle \mathbf{g}_k, \mathbf{x}_k - \mathbf{x}^* \rangle].$$

3. Please show that

$$\mathbb{E}_{\xi_1:\xi_k}[f(\mathbf{x}_k) - f^*] \leq \frac{1}{2\alpha}\mathbb{E}_{\xi_1:\xi_k}[\|\mathbf{x}_k - \mathbf{x}^*\|_2^2 - \|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 + \alpha^2\|\mathbf{g}_k\|_2^2].$$

4. Please show that

$$\begin{aligned}\mathbb{E}_{\xi_1:\xi_K}[f(\tilde{x}_K) - f^*] &\leq \frac{\|x_1 - x^*\|_2^2 + \alpha^2 G^2 K}{2\alpha K} \\ &\xrightarrow{O(1/K)} \frac{\alpha G^2}{2}.\end{aligned}$$

Solution: ■

Exercise 8: Programming Exercise: Logistic Regression

We provide you with a dataset of handwritten digits[1] that contains a training set of 60000 examples and a test set of 2018 examples ("hw4_lr.zip"). Each image in this dataset has 28×28 pixels and the associated label is the handwritten digit—that is, an integer from the set $\{0, 1, \cdots, 9\}$—in the image. In this exercise, you need to build a logistic regression classifier to predict if a given image has the handwritten digit 9 in it or not. You can use your favorite programming language to finish this exercise.

1. (a) Choose a proper normalization method to process the data matrix. Please report the normalization method you use.
   (b) Find a Lipschitz constant of $\nabla L(\mathbf{w})$, where $L(\mathbf{w})$ is the objective function of the logistic regression after normalizing and $\mathbf{w}$ is the model parameter to be estimated. Please report your result.
2. (a) Use GD and SGD to train the logistic regression classifier on the training set, respectively. Evaluate the classification accuracy on the training set after each iteration. Stop the iteration when Accuracy $\geq 90\%$ or total steps are more than 5000. Please plot the accuracy of these two classifiers (the one trained by GD and the other trained by SGD) versus the iteration step on one graph.
   (b) Compare the total iteration counts and the total time cost of the two methods (GD and SGD), respectively. Please report your result.
   (c) Compare the confusion matrix, precision, recall and F1 score of the two classifiers (the one trained by GD and the other trained by SGD). Please report your result.
3. (a) The training set is imbalanced as the majority class has roughly ten times more images than the minority class. Imbalanced data can hurt the performance of the classifiers badly. Thus, please undersample the majority class such that the numbers of images in the two classes are roughly the same.
   (b) Use GD to train the logistic regression classifier on the new training set after undersampling. Stop the iteration when Accuracy $\geq 90\%$ or total steps are more than 5000.
   (c) Evaluate the two classifiers (the one trained with GD on the original training set and the other trained on the new training set after undersampling) on the test set. Compare the confusion matrix, precision, recall and F1 score of the two classifiers. Please report your result.

Solution: ■

[1] This dataset is modified from the MNITS dataset: http://yann.lecun.com/exdb/mnist/

References

[1] D. P. Bertsekas and J. N. Tsitsiklis. *Introduction to probability*, volume 1. Athena Scientific Belmont, MA, 2002.