BMEG3105 Data analytics for personalized genomics and precision medicine

Week 6 Lecture 10

Classification Performance Evaluation

By Cheung Ho Lun 1155174348 11/10/2024

1. Suggested reason that using gradient descent algorithm instead of differentiation

➔ Some functions are extremely complicated e.g. multivariable function, which is very difficult to perform differentiation to find the minimum

Exercise (Ureply): True: The initialization of Ws can affect the training time of the model

About last lecture:

The problems of KNN: need to store all the data, many calculations, slow predictions

Logistic function $\frac{1}{1+e^{-(w_h H+w_w W+w_0)}} \geq 0.5$

Loss function $(\gamma^{output} - \gamma)^2$

Gradient descent algorithm $L = \sum_{P1}^{P4}(\gamma^{output} - \gamma)^2$

Calculate the outpout γ^{output}

Update weights

$w_i = w_i + \Delta w_i$, where $\Delta w_i = 2 * \alpha(\gamma - \gamma^{output})\frac{\partial \gamma^{output}}{\partial w_i}$

α is a small constant to measure step

From logistic regression to neural network

2. <u>Purpose of Model Evaluation</u>

➔ To pinpoint the strong points and weak points of a method e.g. clustering, classification

3. <u>Classification Performance Evaluation</u>

Given:

Person	Height(m)	Weight(kg)	Male?	Prediction	
P1	1.79	75	Yes	Yes	TP
P2	1.64	54	No	No	TN
P3	1.70	63	Yes	No	FN
P4	1.88	78	Yes	Yes	TP
P5	175	70	Yes	No	FN
P6	1.65	52	No	Yes	FP

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

Confusion Matrix:

		Predicted class	
		Class=Yes	Class=No
Actual class	Class=Yes	2 (TP)	2 (FN)
	Class=No	1 (FP)	1 (TN)

TP: True Positive
TN: True Negative
FP: False Positive
FN: False Negative

Hence, Accuracy = $\frac{2+1}{6}$ **=** $\frac{3}{6}$ **= 0.5**

➔ **Limitation of accuracy:** It may be misleading for imbalanced data e.g. an imbalanced with 90% of actual class is yes while only 10% is no

How to solve the problem?

➔ Precision, Recall, F1 score, and Balanced accuracy

		Predicted class	
		Class=Yes	Class=No
Actual class	Class=Yes	4949(TP)	0(FN)
	Class=No	51(FP)	0(TN)

(Given)

Precision: To find how many samples are really correct among the predicted positive samples

Precision = $\frac{TP}{TP + FP}$ = 0.99

Recall: To find how many actual positive samples that can be predicated to be positive

Recall = $\frac{TP}{TP + FN}$ = 1

F1 score: The weighted average of precision and recall

F1 score = $\frac{2*precision*recall}{precision+recall}$ = 0.995

Balanced accuracy = 0.5 * ($\frac{TP}{TP + FN}$ + $\frac{TN}{TP + FN}$) = 0.5, where $\frac{TN}{TP + FN}$ is the actual negative of samples which are predicated to be negative

Exercise:

		Predicted class	
		Class=Yes	Class=No
Actual class	Class=Yes	2(TP)	0(FN)
	Class=No	50(FP)	50(TN)

Accuracy ~0.5, precision = 0, recall = 1, F1 score ~0.5, balanced accuracy ~0.5

➔ **To determine whether the classifier is good or not, it depends what kinds of problem you**

emphasize.

In this case, the recall is good while other parameters are bad. However, if we care about recall, this classifier is in fact good. Examples of cases that we care about recall is *cancer pre-screening*. Thus, **value is not absolute, Context matters.**

4. How to choose a good K for KNN?

A good K: It can give us good prediction accuracy

Problem: We do not have the label for testing data. Thus, we cannot calculate the accuracy for each K

Solution: Use part of the training data as the testing data, use each part one by one, then calculate the average over all the parts (Cross-fold validation)

Example:

Person	Height	Weight	Gender
P1	0.625	0.875	M
P2	0	0	F
P3	0.25	0.375	M
P4	1	1	M
P5	0.4583	0.6667	??

Distance matrix:

	P1	P2	P3	P4
P1	0	0.875	0.5	0.375
P2	0.875	0	0.375	1
P3	0.5	0.375	0	0.75
p4	0.375	1	0.75	0

Let us choose K =1,

In P1 (P1 as the testing data), the closest distance with P1 is P4, so P1 would be predicted as M

In P2 (P2 as the testing data), the closet distance with P2 is P3, so P2 would be predicted as M

Similarly, P3 would be predicted as F, and P4 would be predicted as M

Accuracy = 0.5

Let us choose K= 3,

In this case, the value of K is the same of the number of training data (3), thus, all testing data (P1, P2, P3 and P4) from each set would be predicted as M

Accuracy = 0.75

From the above, we should select K = 3

5. Cross-fold validation

➔ It is a technique for assessing how the results of a machine learning analysis will generalize to an independent data set. In simple words, it is a procedure to measure the performance of models

➔ One round of cross-validation involves partitioning a set of data into complementary subsets, performing the analysis on the training set, and validating on the testing set

n-fold validation

Name: the value of 'n' refers to how many groups of training data we have

Idea: we train multiple times, leaving out a disjoint subset of data each time for validation.

Then, average the validation set accuracies

Process:

A. Randomly partition data into n disjoint subsets
B. For i = 1 to n
   - Validation Data = i-th subset
   - H <- classifier trained on all data except for Validation Data
   - Accuracy(i) = accuracy of h on Validation Data
C. Final accuracy = mean of the n recorded accuracies

Note: Never touch the testing data

Examples:

Let's say we do a **5-fold cross-validation** and there are 10 data points (P1-P10)

For 5-fold (5 groups of training data), we have

➔ P1-P2, P3-P4, P5-P6, P7-P8, P9-P10 (can be random)

Procedure:

- P1-P2's results based on the model from P3-P10

Similarly, P9-P10's results based on the model from P1-P8

- Averaging

Leave-one-out cross-validation

➔ It's a special case of n-fold cross-validation, where the number of training data groups equal the number of training data (**n = N**)

➔ The process and procedure are the same with n-fold cross-validation

Exercise (Ureply):

What cannot we use cross-validation for?

A. Estimate the performance of a model on the testing data
B. Choose the K for KNN
C. Train the weights for logistic regression
D. Select a better classification method from KNN and logistic regression

6. Multi-class classification

Example: we have 6 classes instead of 2 classes for classification

We can choose the method of KNN or logistic regression

A. KNN: no need to change the algorithm. It is because it does not rely on the number of

candidates

B. Logistic regression: we need to change

➔ Build a logistic regression for each class

➔ When predicting, we assign class with highest value

➔ When training, we train 3 (parameters) * 6 (class) = 18 parameters

In this case, we convert 6 classes to 6 binary classifications

Multi-class evaluation

➔ Still using accuracy, precision, recall, F1 score……

➔ Considering each class as a binary classification problem

How to aggregate multiple values into one value?

A. *Macro – average* = $\frac{sum\ of\ accuracy\ from\ each\ class}{Total\ number\ of\ class}$

B. *Micro – average* = $\frac{\sum_{n=1}^{k} accuracy * cells}{Total\ number\ of\ cells}$, where k is the last number of the class

Example:

Class	Accuracy	Cells
1	0.9	150
2	0.95	50
3	0.85	100
4	0.8	40
5	0.7	20
6	0.2	10

Macro – average = $\frac{sum\ of\ accuracy\ from\ each\ class}{Total\ number\ of\ class}$ = $\frac{0.9+0.95+\cdots+0.7+0.2}{6}$ = 0.73

Micro – average = $\frac{\sum_{n=1}^{k} accuracy * cells}{Total\ number\ of\ cells}$ = $\frac{0.9*150+\cdots+0.2*10}{150+\cdots+10}$ = 0.85

From the above, the low-performance of small classes will show up in Macro-average

Exercise (Ureply):

We have 4 classes. Below are the prediction results of the 4 binary logistic regression classifiers for a data point. Which class should we assign?

1:0.9, 2:0.8, 3:0.2, 4:0.3

A. 1

B. 2

C. 3

D. 4