2. Item Response Theory

1. Derive the log-likelihood and show the derivative of the log-likelihood with respect to separately

*Solution:*

*Part 1: format of log-likelihood*

Given response matrix C, whose rows are indexed by students and columns are indexed by questions. The entries of the marks are binary, 1’s and 0’s. If Cij equals to 1, it means that the student i gets a correct answer to question j.

Assume there are N students and M questions.

For a certain student, some of the questions he answers are correct and some are wrong.

Now we write the likelihood of a set of parameter values , given a certain pattern of outcomes :

Based on the assumption that all students are independent, the likelihood function of the entire test is the product of the likelihood function of each student

Thus, we have likelihood function:

is the indicator for student i correctly answered question j

Log-likelihood:

*Part 2: partial derivatives respect to*

*Part 3: partial derivatives respect to*

1. Implement missing functions that perform alternating gradient descent on to maximize the log-likelihood and report the hyperparameters selected.

With the chosen hyperparameters, report the training curves that shows the training and validation log-likelihoods as a function of iteration.

*Solution:*

*Part 1: Basic thoughts of code implementation*

To find so as to maximize log-likelihood equivalent to minimize negative log-likelihood

*Part 2: tune hyperparameters*

Since there are two hyperparameters, learning rate and number of iterations to tune, we did the following experiments:

1. Experiment on learning rate

Fixed number of iterations = 500, and initial values of theta and beta are all 1’s

* Learning rate = 0.0005

Chart, line chart

Description automatically generated

* Learning rate = 0.001

Chart, line chart

Description automatically generated

* Learning rate = 0.01

A picture containing chart

Description automatically generated

From all three plots, we can tell that the accuracy rate curves rise rapidly initially and become stable after some number of iterations. Validation accuracy is always lower than training accuracy. The accuracy of training set is at around 0.74, and the accuracy of validation set is about 0.71.

We can also tell that when learning rate equals to 0.01, curves are too steep; when learning rate equals to 0.0005, accuracy is still slightly rising after 500 iterations, thus it rises relatively too slow. Learning rate of 0.001 is neither too fast nor too slow.

1. Experiment on number of iterations

Fixed learning rate =0.001, and initial values of theta and beta are all 1’s

* Number of iterations = 100

Chart, line chart

Description automatically generated

* Number of iterations = 300

Chart, line chart

Description automatically generated

* Number of iterations = 1000

Chart

Description automatically generated

We can tell that after 300 iterations, both training and validation accuracy reach their maximum values at about 0.74 and 0.70 separately and stay stable. Thus, 300 iterations are acceptable in this case.

Based on the experiments above, we finally choose the following hyperparameters as the final parameters:

Initial values of theta and beta are 1

Learning rate = 0.001

Number of iterations = 300

Part 3: plots report

Fixed learning rate = 0.001, number of iterations = 300

A picture containing graphical user interface, chart

Description automatically generated

We can tell from the plot above that, both training and validation set log-likelihood are negative and rise as number of iterations increases. With number of iterations gets larger, the slopes of both training and validation sets become shallower. This trend is consistent with what we learned: as more iterations go, it converges to its extreme value.

1. With the implemented code, report the final validation and test accuracy

After running the code, we get the following accuracy output:

final accuracy of validation set is 0.7066045723962744

final accuracy of test set is 0.7053344623200677

We can tell that validation set’s accuracy is slightly higher than test set, but they are still quite similar.

1. Select five questions and use the trained to plot five curves on the same plot that shows the probability of the correct response as a function of given a question . Comment on the shape of the curve and briefly describe what these curves represent.

Solution:

Part 1: ideas to get the formula

Given

Now treat the difficulty of a certain question , , be a constant, then probability of the correct response is a function of

*Part 2: plot*

Chart

Description automatically generated

*Part 3: Interpretation*

We randomly selected 5 questions, which are Q818, Q94, Q670, Q481, Q988

From the plot we can tell that these 5 curves have similar shape – “s” curve

For all 5 curves, we can tell that, as gets larger, the probability of correct response gets larger as well. This is reasonable, since the greater the ability of the student has, the higher chance he or she has to give the correct answer. The probability is bounded by 0 and 1 obviously.

We can also tell that, for the same value, the probability to answer correctly for the 5 randomly chosen questions are:

Prob(correct) for question 988 > Prob(correct) for question 481 > Prob(correct) for question 670 > Prob(correct) for question 94 > Prob(correct) for question 818.

Since for the same student, his or her ability is fixed, then the probability of answer correctly is decided by the difficulty of given questions. The simpler the given question is, the higher probability the question is answered correctly. Thus, we would guess: question 818 is the most challenging one among the 5 chosen questions, then question94, then question 670, then question481, question 988 is the simplest one among the 5 question. To prove it, we printed the difficulty of these 5 questions, which are:

selected five questions has the following difficulty level:

question 670 has difficulty[0.84724177],

question 818 has difficulty[2.46655858],

question 94 has difficulty[1.92220531],

question 481 has difficulty[0.65751092],

question 988 has difficulty[0.30782227]

From the numerical representations of difficulty of questions, we can tell that the difficulty has:

Q818 > Q94 > Q670 > Q481 > Q988

This is consistent with what we observed in the plot