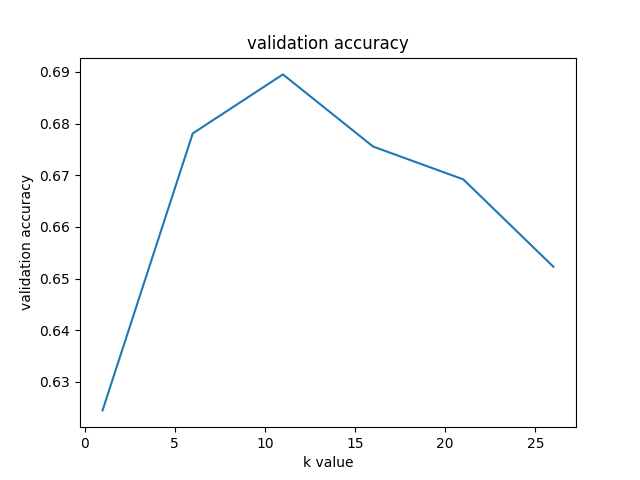
**CSC311 project report**

Part A

1.

(a)

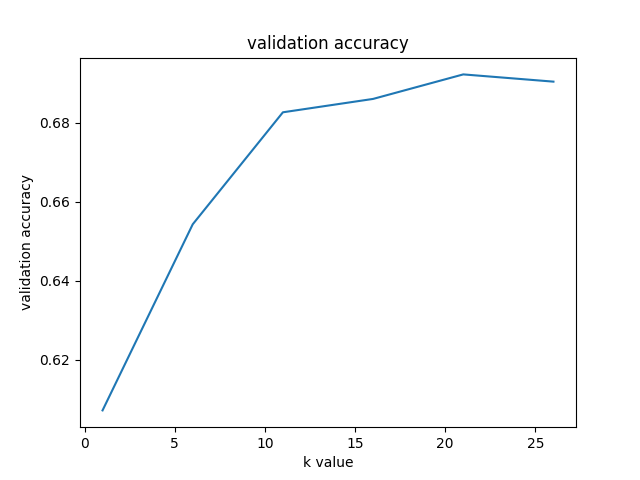


(b)

k\* = 11 has the best validation accuracy of 0.6895, and its test accuracy of 0.6842.

(c)

The underlying assumption of this scenario is that if question A has the same correct and incorrect answers with other students as question B, A’s correctness on specific students matches that of question B.



k\* = 21 has the best validation accuracy of 0.6922, and its test accuracy of 0.6816.

(d)

Imputing by user has accuracy of 0.6842 on the testing set.

Imputing by item has accuracy of 0.6816 on the testing set.

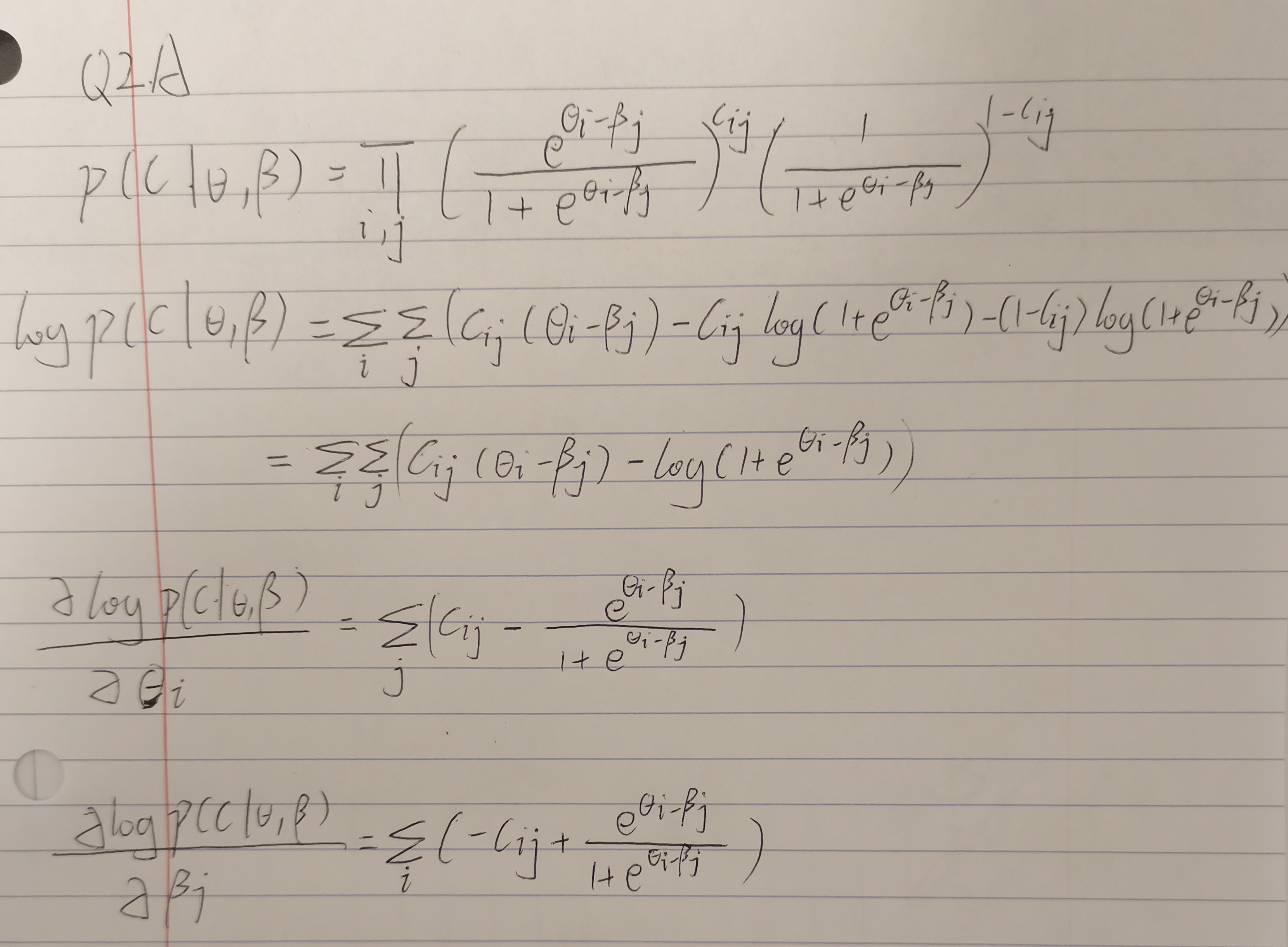
Imputing by user has better performance.

(e)

1. The underlying assumptions of either cases are invalid.

2. There is no strong relevance between students and questions, which means there could be many noisy samples.

2.

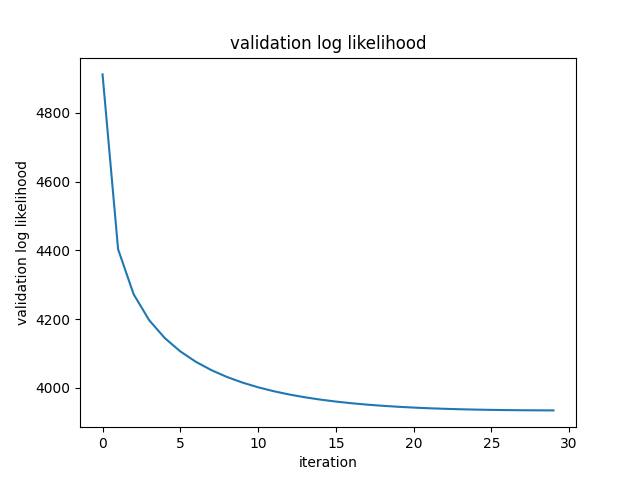
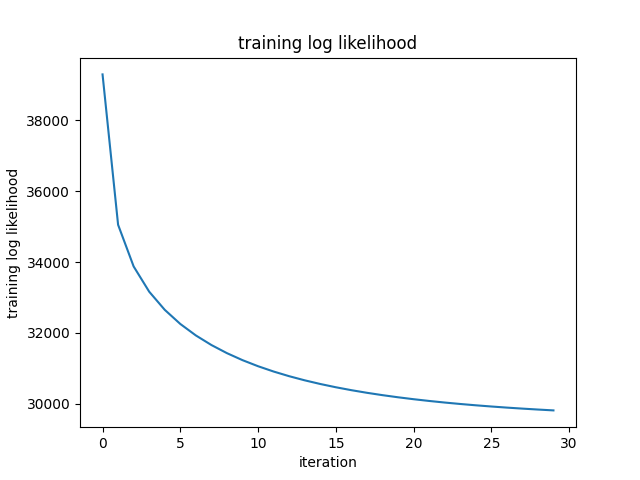


(b)

Chosen parameters:

Learning rate = 0.01

Iteration = 30



(c)

Output:

NLLK: 29814.94707407383 Train Score: 0.7388159751622918 Validation Score: 0.7067456957380751

Test Score: 0.7053344623200677

Validation accuracy: 0.7067

Test accuracy: 0.7053

(d)



All three curves in the figure above show a trend of increasing. Theta represents a student’s ability, and the probability shows the possibility of a student to answer a question correctly. These increasing curves show that students with better ability have higher possibility to answer questions correctly.

3. Option 1 Matrix Factorization

(a)

Output:

k = 3, validation accuracy = 0.6583403895004234

k = 5, validation accuracy = 0.659046006209427

k = 7, validation accuracy = 0.6591871295512278

k = 9, validation accuracy = 0.6613039796782387

k = 11, validation accuracy = 0.656929156082416

k = 13, validation accuracy = 0.6559412926898109

k = 15, validation accuracy = 0.6565057860570138

final k = 9, validation accuracy = 0.6613039796782387, test accuracy = 0.6587637595258256

Chosen k = 9 with final validation accuracy = 0.6613, and test accuracy = 0.6588

(b)

A missing value in the sparse matrix is filled with the mean value of the existing values in that column. The underlying assumption is that if a student gives correct answers to half of questions he attempts to do, he will answer all questions correctly, which is not true and could lead to lower accuracy.

(c)

Coding

(d)

Hyperparameters:

Learning rate: 0.05

Iterations: 100000

Output:

k = 3, validation accuracy = 0.6953147050522156

k = 5, validation accuracy = 0.6922099915325995

k = 7, validation accuracy = 0.698842788597234

k = 9, validation accuracy = 0.6958791984194186

k = 11, validation accuracy = 0.6965848151284222

k = 13, validation accuracy = 0.6867061812023709

k = 15, validation accuracy = 0.6994072819644369

k = 17, validation accuracy = 0.6930567315834039

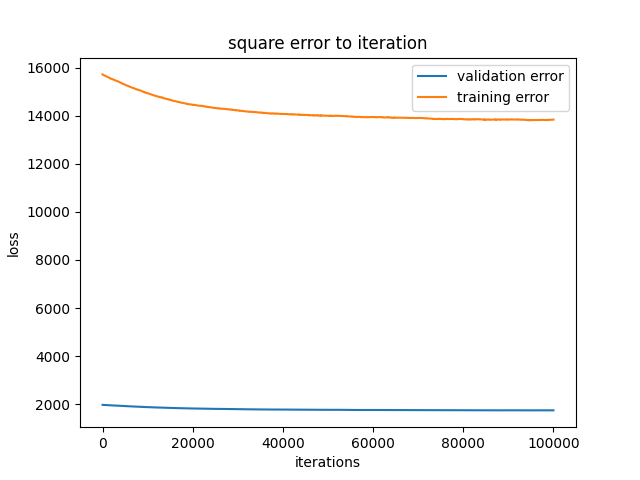
k = 19, validation accuracy = 0.6915043748235958

final k = 15, validation accuracy = 0.7002540220152413, test accuracy = 0.6917866215071973

k\* = 15 achieves highest validation accuracy = 0.7003.

Here the best k would sometimes be 7 or 9 due to the randomness introduced by stochastic gradient descend.

(e)



Both the training loss and the validation loss decrease as the gradient decent iterates.

final validation accuracy = 0.7003

final test accuracy = 0.6918

4.

Output:

irt

Ensemble Train Score: 0.7168889359300028 Validation Score: 0.6951735817104149 Test Score: 0.6931978549252046

Original Train Score: 0.7388688964154672 Validation Score: 0.7068868190798758 Test Score: 0.7053344623200677

Final validation accuracy: 0.6952

Final test accuracy: 0.6932

Firstly, we resample the training set to three smaller sets with replacement, and we run IRT algorithm with each resampled set. Next, we generate the binary prediction with each IRT result and make the final decision according to the majority vote of each models’ prediction. Finally, we calculate the accuracy on training set, validation set and test set.

We did not receive better performance. This could be explained by the nature of Item Response Theory (IRT). The IRT is actually trying to learn the ability of students (theta) and the difficulty of questions (beta). Bagging in this algorithm would reduce the information we can learn on a student or a question, which might cause the algorithm to underestimate or overestimate theta and beta. In this way, the accuracy of each model could decrease, so that the overall accuracy would become lower.