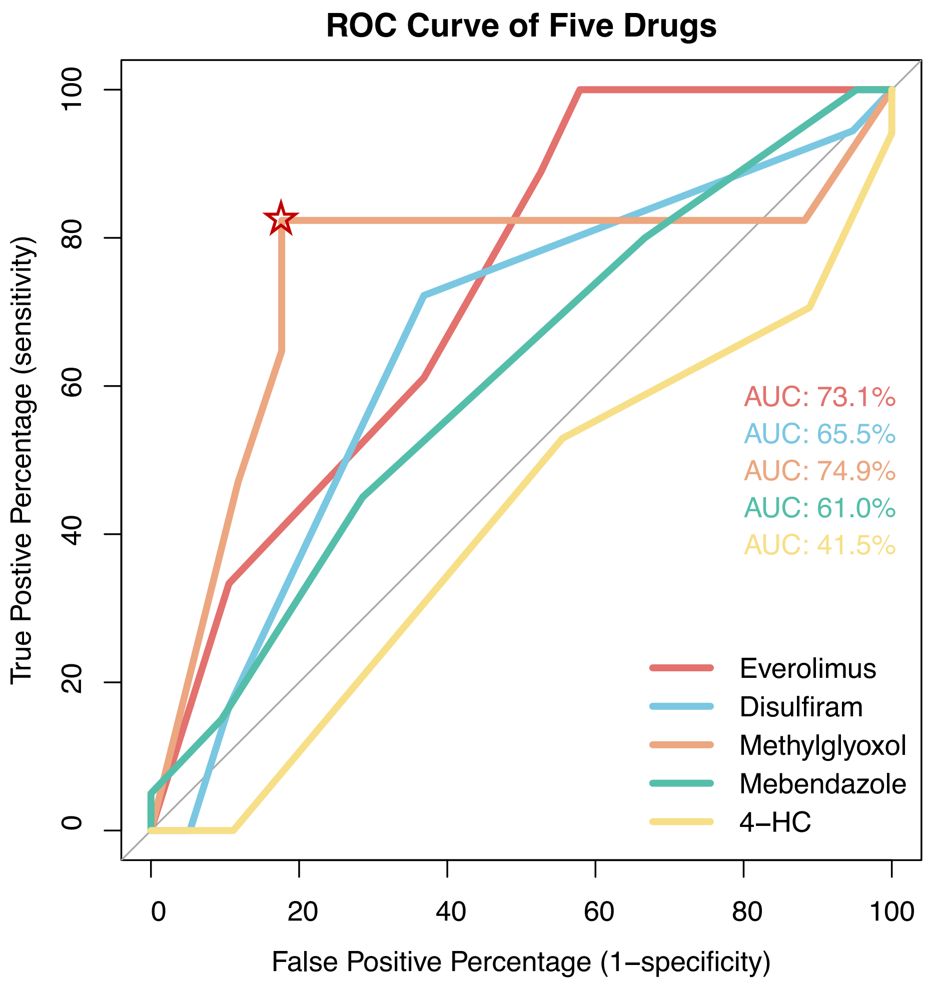
Homework 2

1. k-NN implementation.

I made a kNN function named kNN_classifier that can take an unknown gene expression profile data, the drug of interest, # of nearest neighbors (k) to predict the score which represents the fraction of the k-nearest neighbors that are sensitive to the corresponding drug.

2. kNN performance evaluation.

The ROC curves of five different drugs (classifiers) are shown in the following graph.



a. Does your classifier work better than a random classifier? For which drugs? Refer to specific evidence from your analysis to justify your answer.

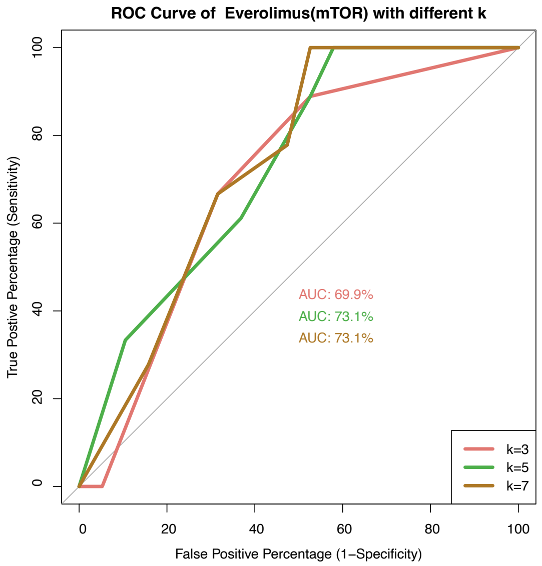
Among 5 different classifiers, 4 of them work better than a random classifier, which are Everolimus, Disulfiram, Methylglyoxol, and Mebendazole. These classifiers are all above the random classifier, which is shown as the straight line in black, and their AUCs are all larger than 50%. One classifier, derived from 4-HC, doesn’t work as well as the random classifier. The curve is below the random classifier and its AUC is less than 50%.

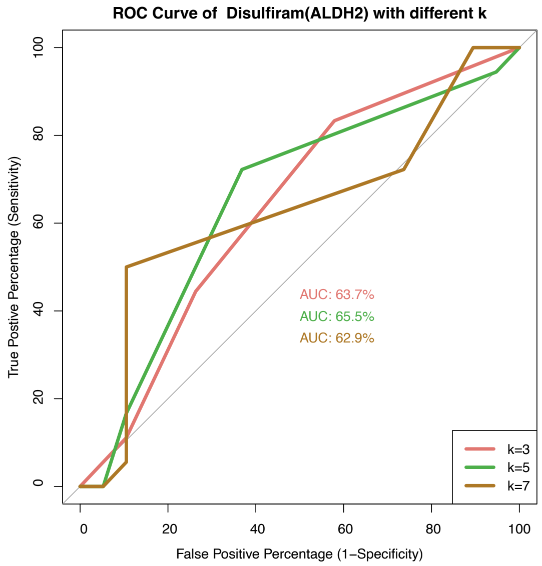
b. For the drug on which your approach appears to work the best, how good is the classification performance? Pick a point on the ROC curve, and report the relevant metrics (number of true positives, false positives).

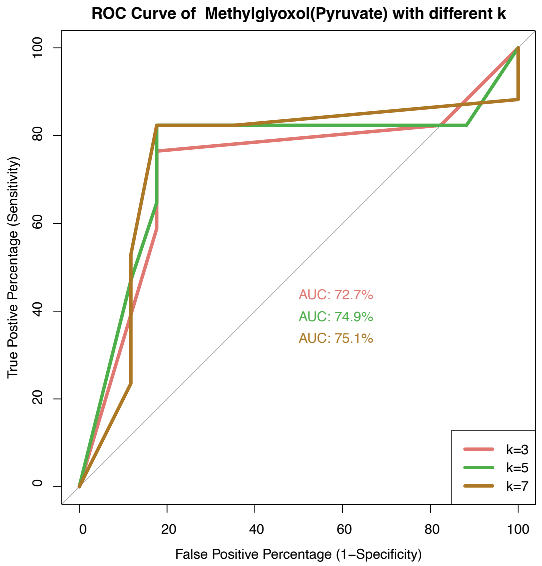
It seems that the classifier derived from Methylglyoxol (Pyruvate) works the best because this classifier has the largest AUC (74.9%). The higher the AUC, the better the classifier is at predicting true positives as true negatives. Specifically, the point indicated by the star has the highest true positives (28, 82.35%) and the lowest false positives (6, 17.65%),

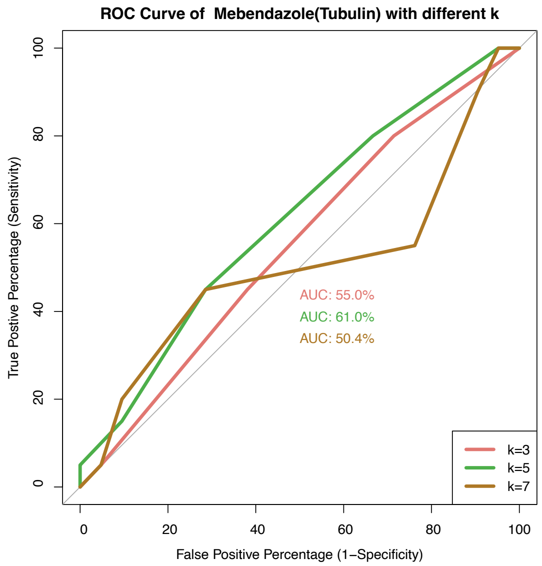
3. Exploration of parameters affecting kNN performance.

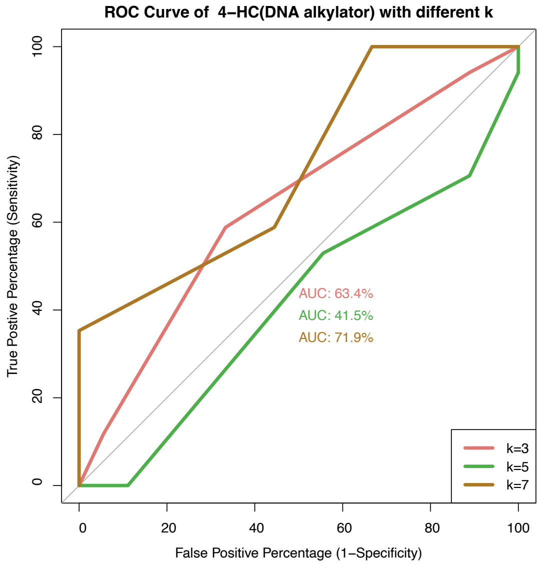
a. Does the choice of k affect the performance of the classifier?









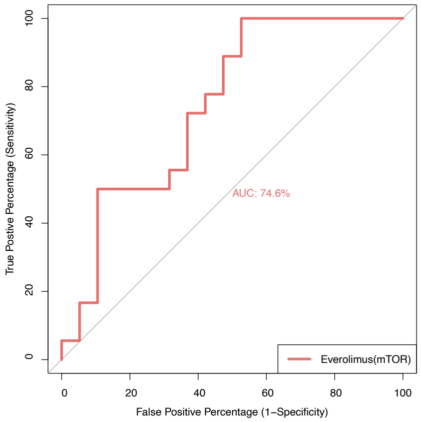


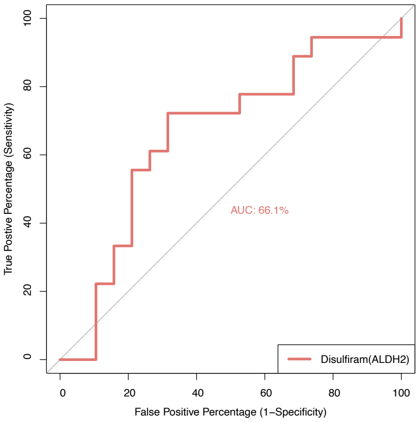
Yes, I think choosing different k parameters affects the performance of kNN classifier. The k value determines the number of neighbors considered when predicting a new/unknown sample. If the k value is too small, for example, 1, the classifier will be overfit; if the k value is large, it will lead to underfitting of the classifier. Therefore, finding the optimal k is important for kNN. This is the reason why we do cross-validation here—we can find the k that gives the best performance.

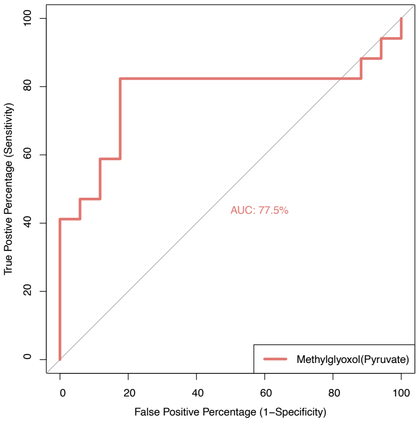
For the ROC curves of the classifiers based on five drugs, it seems that the relatively larger k (5 or 7) gives best performance based on the provided AUCs.

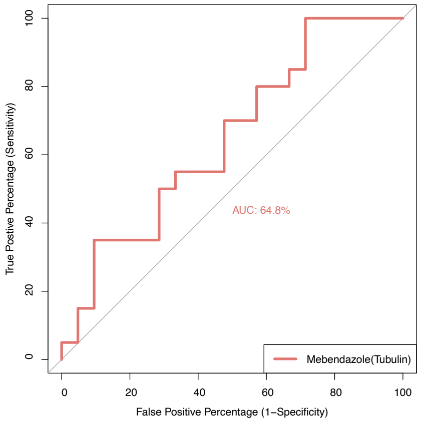
b. kNN with weighted scores.

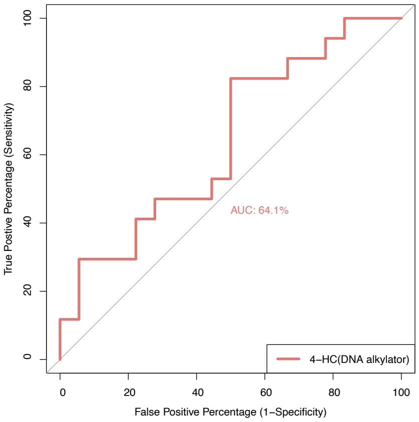
The following ROC curves were made with pROC package. Compared to the classifier made in Question 2, all the AUCs increased.



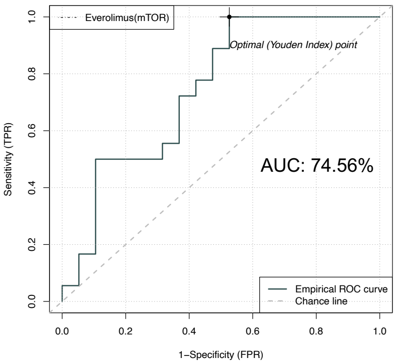


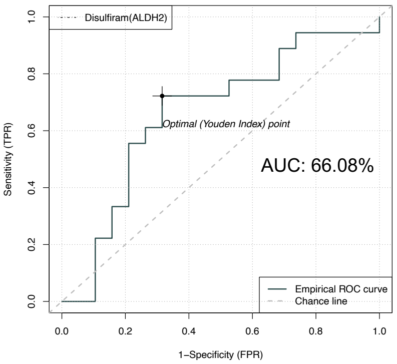


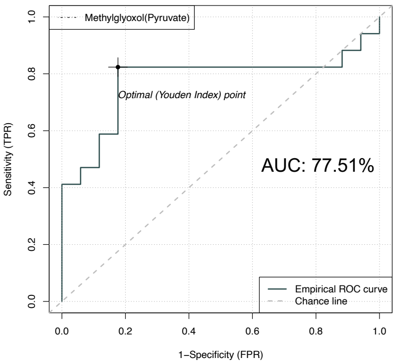


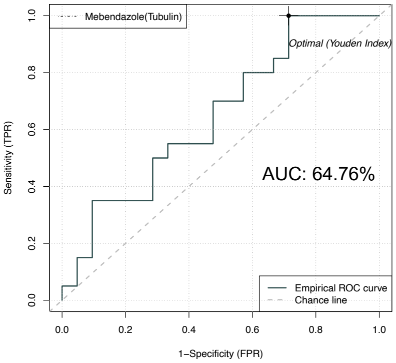


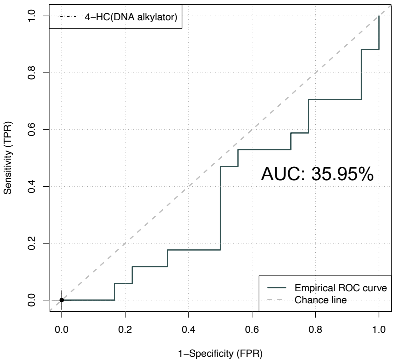
The following ROC curves were made with the ROCit package. Compared to the classifier made in Question 2, the AUCs of classifiers based on Drug 1–4 increased, but the classifier of Drug 5 provided an AUC to be 35.95%, which is lower than the classifier in Questions.











*** Interesting thing here. I used two packages ROCit and pROC. The comparison of AUCs for each classifier based on each drug is listed in the following table. For the ROC curves for Drug 1–4, both packages gave me identical graphs and AUCs. However, the ROC curve for Drug 5 is different between the two packages, and so are AUCs. I couldn't understand why it was happening.

	AUCs from Question2 (%)	AUCs from pROC package (%)	AUCs from ROCit package (%)
Drug1	73.1	74.6	74.56
Drug2	65.5	66.1	66.08
Drug3	74.9	77.5	77.51
Drug4	61.0	64.8	64.76
Drug5	41.5	64.1	35.95