Jiayuan Zhu   1067734

Q1

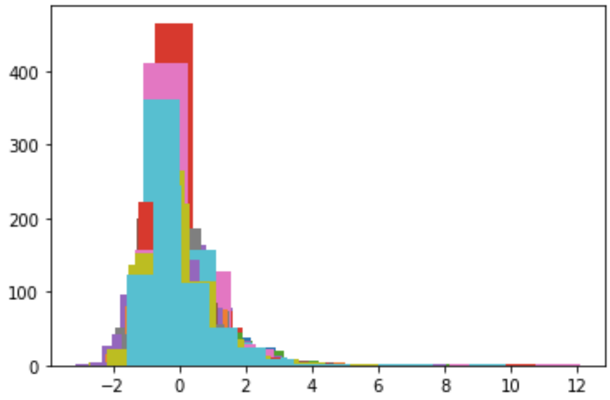
Overall speaking, discretizing the variables improves classification performance compared to GNB, but there’s exception as well. The basic strategy is to apply both GNB and classic NB after three different ways of discretizing, then comparing average accuracy. Discretization methods seem to give better accuracy because GNB always simply assumes that the data is normally distributed. But actually, the real data may have other distributions or outliers or skewed. From the following plots, it’s clear that only wine dataset looks like normal distribution. Consequently, GNB works worse for other datasets. Thus, for most cases, discretizing the variables seem to be a better choice because GNB is too naive to assume all dataset follows normal distribution. The exception appears generally for two reasons and the first reason is the scenario mentioned above that the data has a really approximate normal distribution. The second is discretizing methods perform really awful. It’s commonly used to discretizing numeric dataset into three to five bins and often works well, but the best way of discretizing varies from different dataset. Though using Elbow Method could help discretization, it still depends on human justification. Sometimes wrong choice of discretizing bins would miss some groups which would further effect the whole NB performance. To conclude, in most cases, discretization method performs better than GNB.

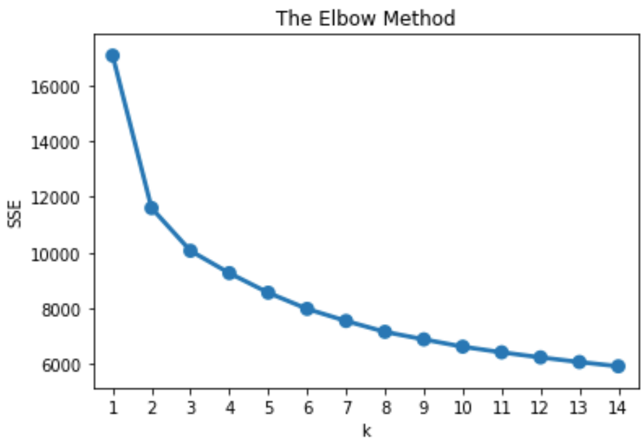
Dataset / Method	GNV	Kmeans	Equal width	Equal frequency
Wdbc	Accuracy:0.92982456 F-score:0.92556000	Accuracy:0.93684211 F-score:0.93338953	Accuracy:0.93859649 F-score:0.93453669	Accuracy:0.93771930 F-score:0.93421030
Wine	Accuracy:0.96944444 F-score:0.97098263	Accuracy:0.95555556 F-score:0.95689127	Accuracy:0.94166667 F-score:0.94100212	Accuracy:0.94444444 F-score:0.94467707
University	Accuracy:0.43030303 F-score:0.29541236	Accuracy:0.46666667 F-score:0.36004241	Accuracy:0.45757576 F-score:0.30605174	Accuracy:0.46969697 F-score:0.36795478
Adult	Accuracy:0.82625000 F-score:0.75368922	Accuracy:0.82675000 F-score:0.78801871	Accuracy:0.80475000 F-score:0.75513439	Accuracy:0.81650000 F-score:0.77502092
Bank	Accuracy:0.86375000 F-score:0.67160609	Accuracy:0.87525000 F-score:0.66129730	Accuracy:0.87600000 F-score:0.65436235	Accuracy:0.86175000 F-score:0.63812951

Distribution

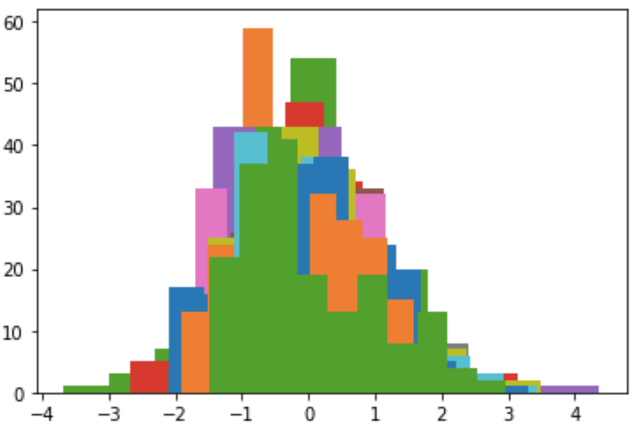
Elbow Method

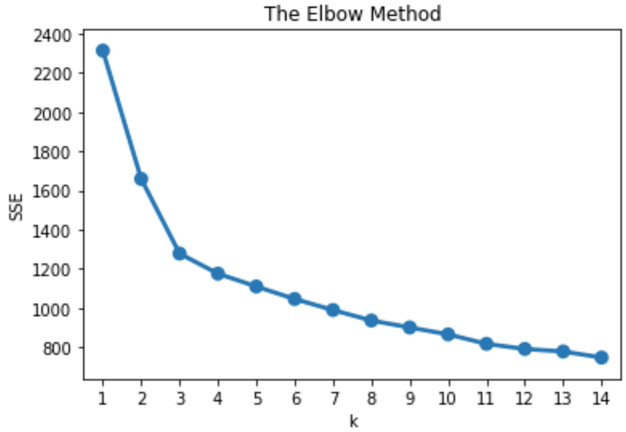
Wdbc



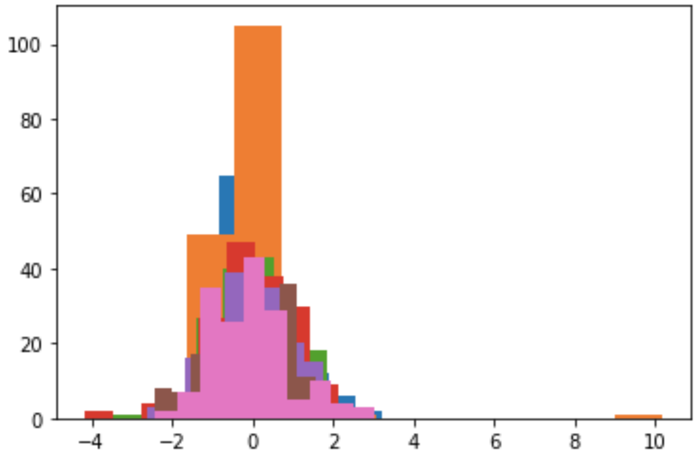


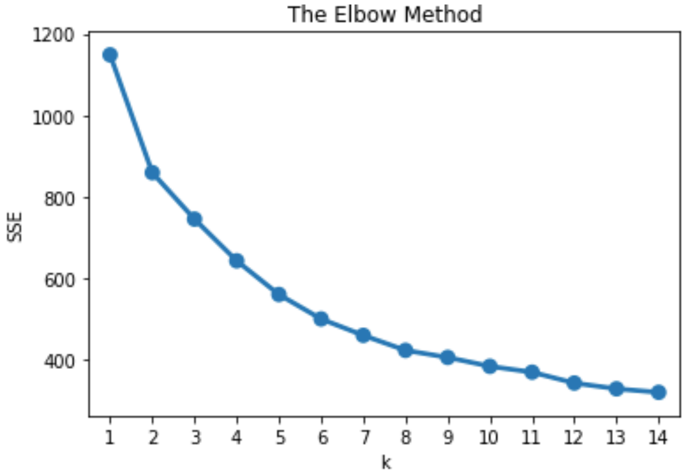
Wine



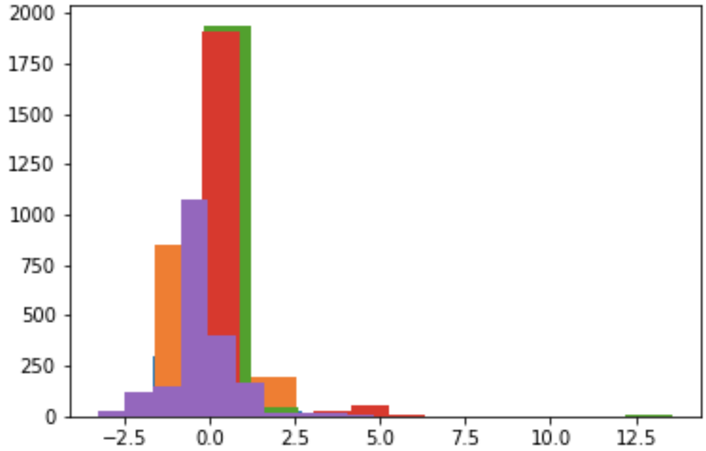


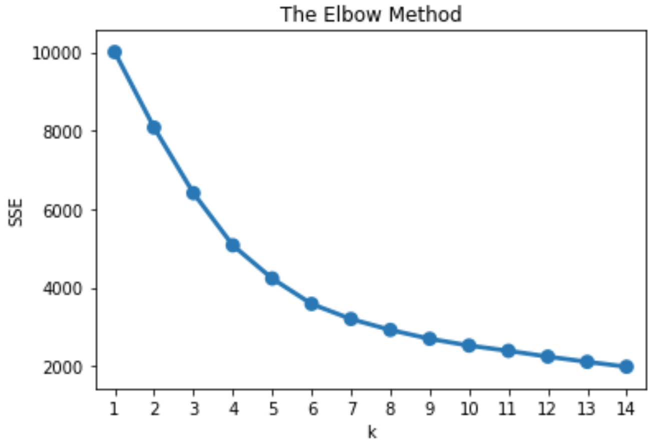
Unive
rsity



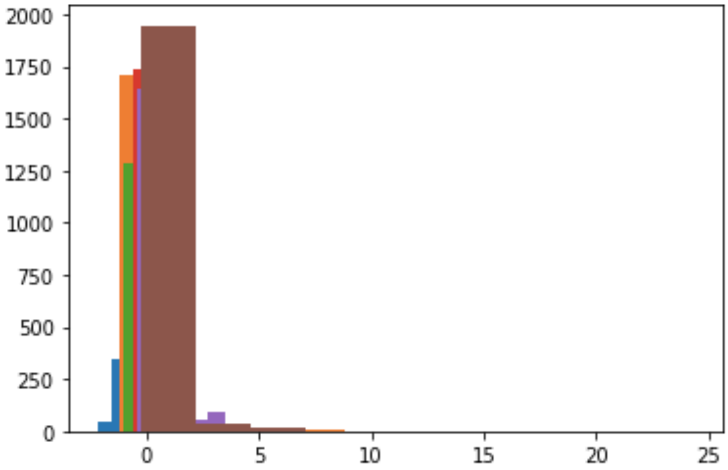


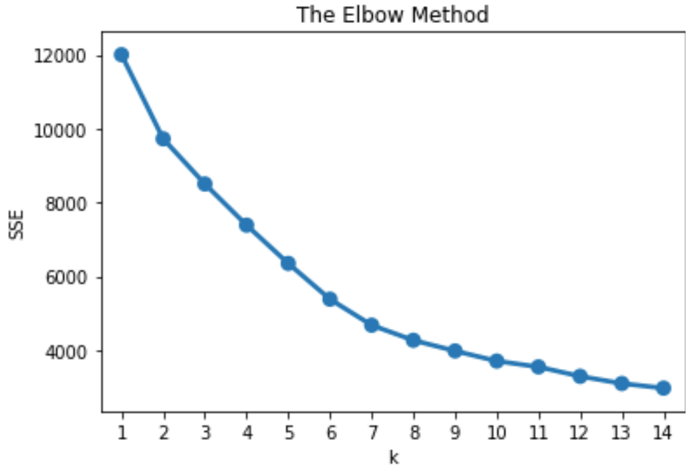
Adult





Bank





Q5

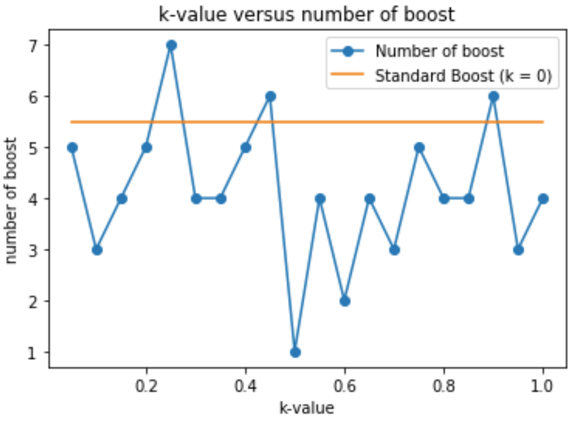
Changing the smoothing regime does affect the effectiveness of the NB classifier. Basic approach is to assign k different values between zero to one (inclusive), and for each k, run the model for ten times to calculate number of boost among this ten trials and average accuracy then compare to without smoothing. For some datasets (wine, breast, lymphography, university, adult), after add-k smoothing, the performance of NB improves (trading off between boost number and average accuracy). This is due to the avoid of overfitting, also known as avoid of zero-frequency problems. This prediction of testing data is only based on training data, without any background information. In the condition of overfitting, the testing data can only generate probability for the categories that have appeared in training data. But there's possibility that some rare category haven't appeared in training data, but would truly appeared in testing data. The add-k regime allows prediction for this kind scenario which would enhance the performance. To the contrast, some datasets' (wdbc, mushroom, car, nursery, somerville, bank) performance decrease due to add-k. The reason is that in these datasets, nearly all categories in testing data have appeared in training data, so there's no extra need to apply add-k regime. When applying add-k, the influence of rare cases increases (actually there's no rare cases) which decreases the influence of common cases (those appeared in training data) so that affects the effectiveness of Naive Bayes classifier as well.

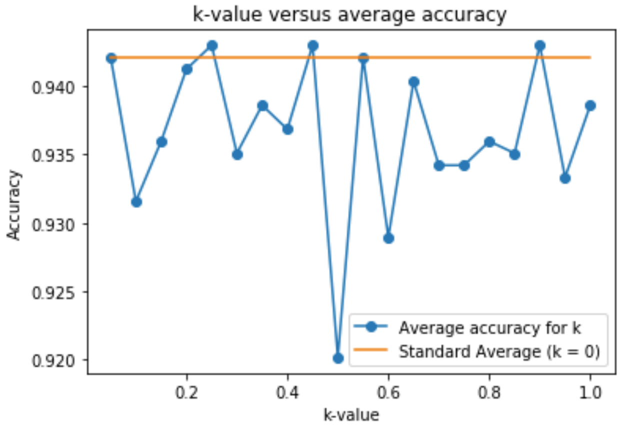
Dataset / Number	Number of category counts appeared in testing data but haven't appeared in training data per testing trial
Wdbc	1.17619048
Wine	1.77142857
Breast	4.20952381
Mushroom	0.01904762
Lymphography	9.05714286
Car	0
Nursery	1.32380952
Somerville	1.45714286
University	27.23809524
Adult	8.27619048
Bank	0.67619048

Number of boost

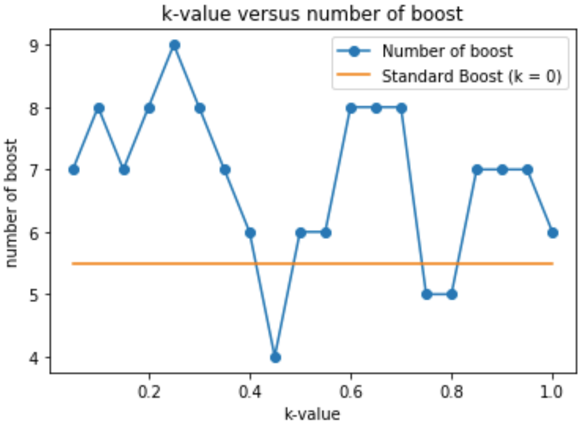
Average Accuracy

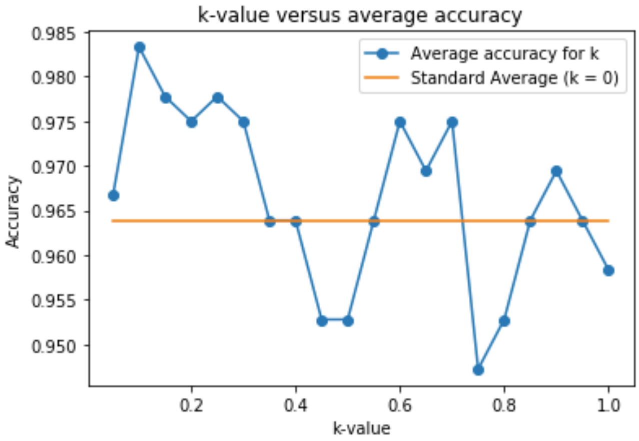
Wdbc



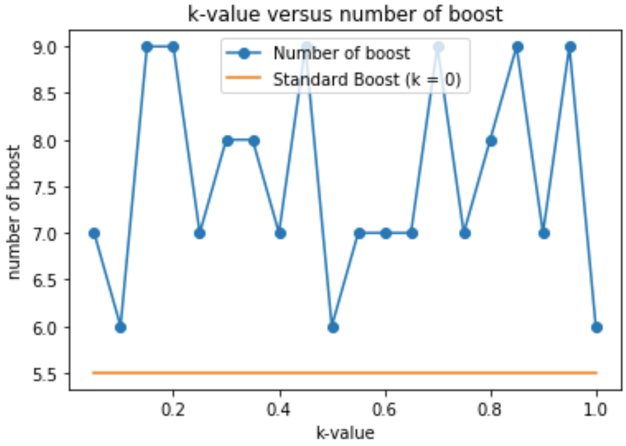


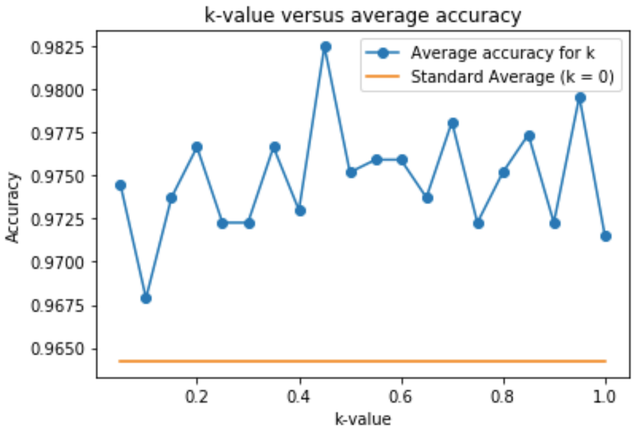
Wine



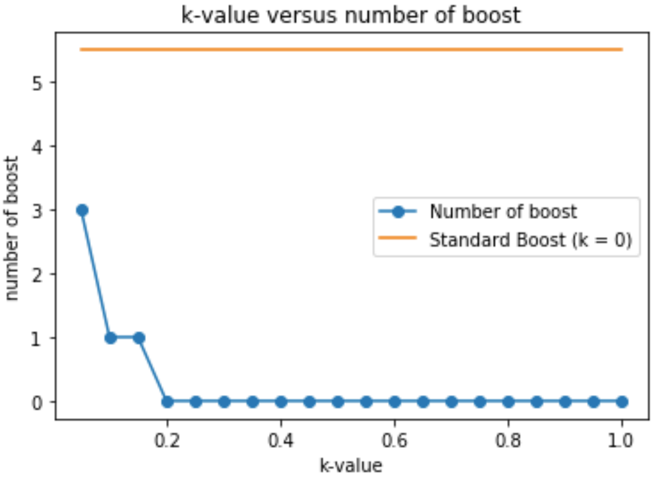


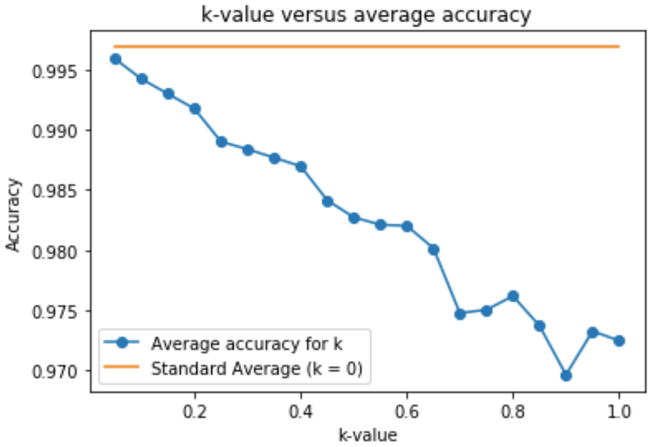
Breast





Mush room

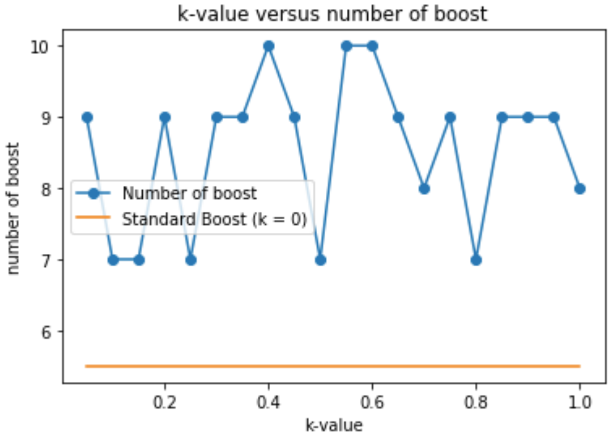


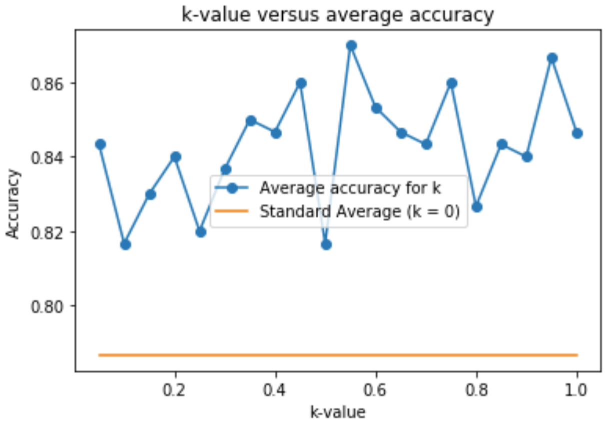


Number of boost

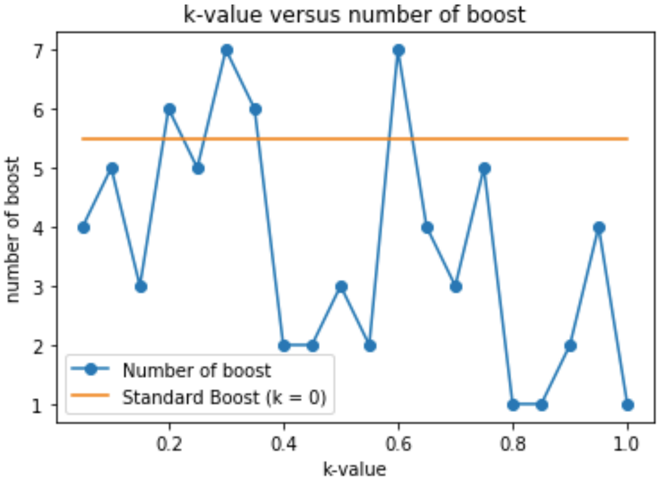
Average Accuracy

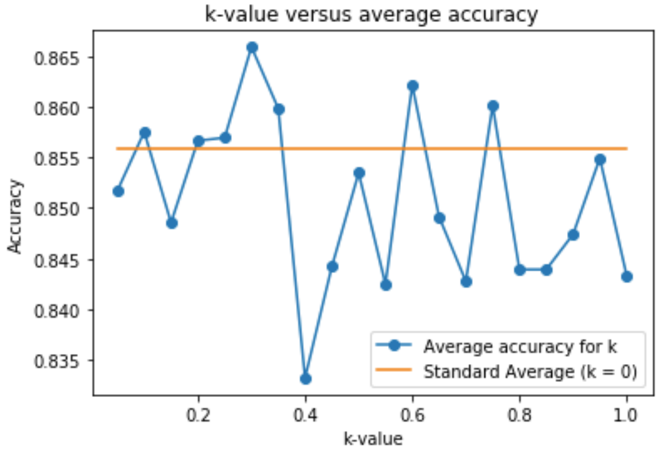
Lymph ograph



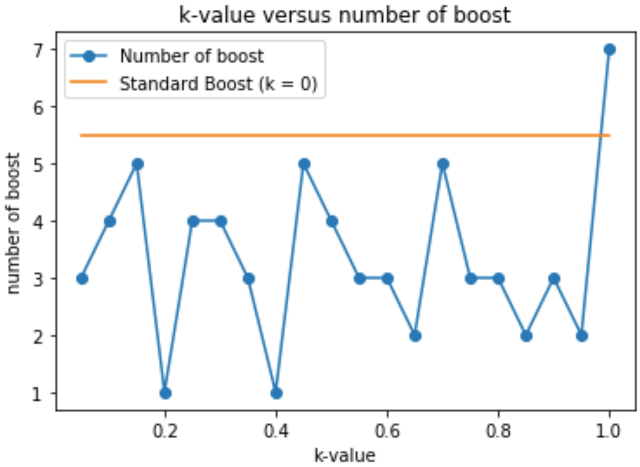


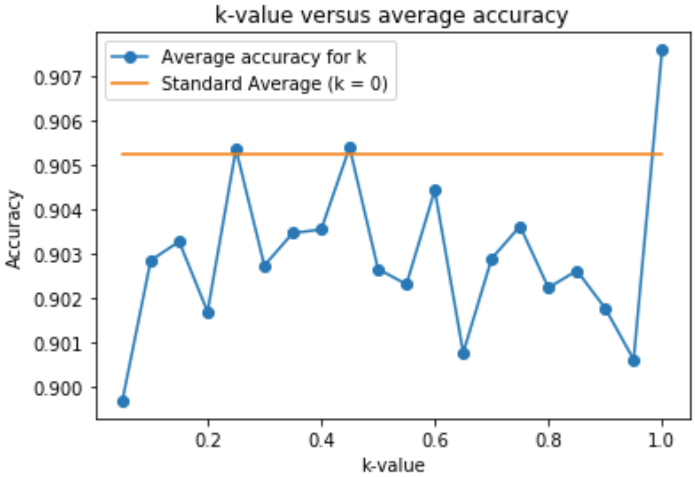
Car



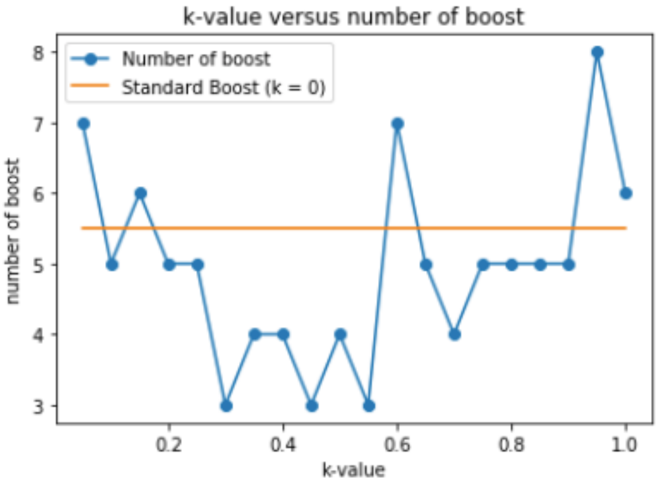


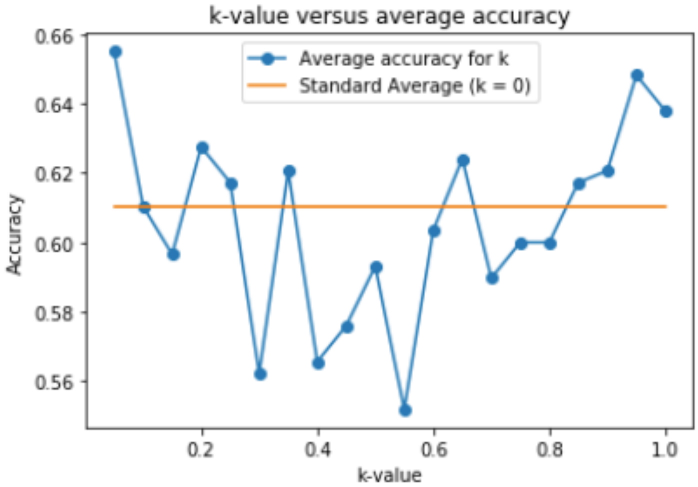
Nursery





some rville

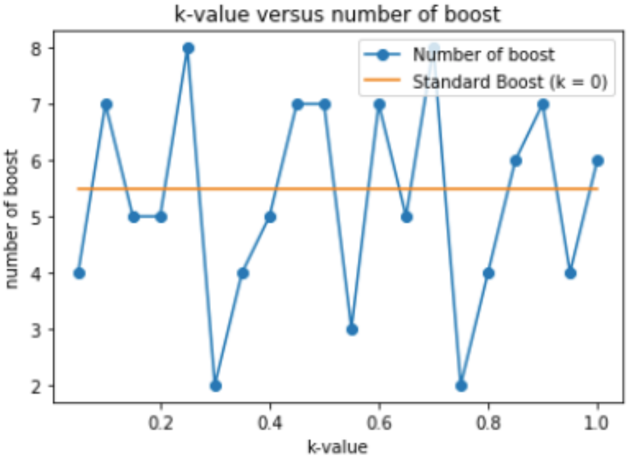


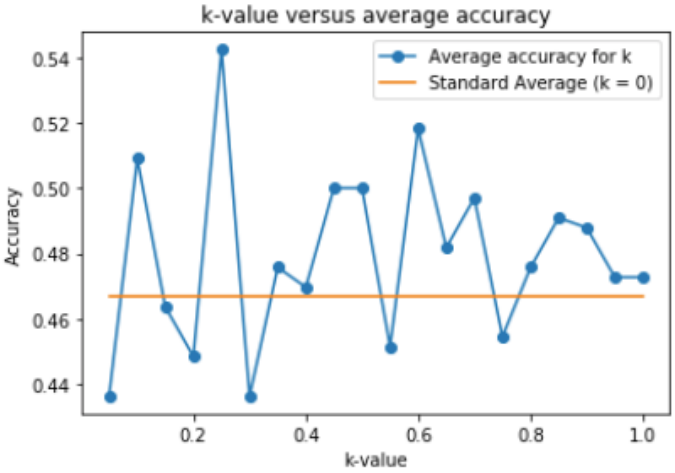


Number of boost

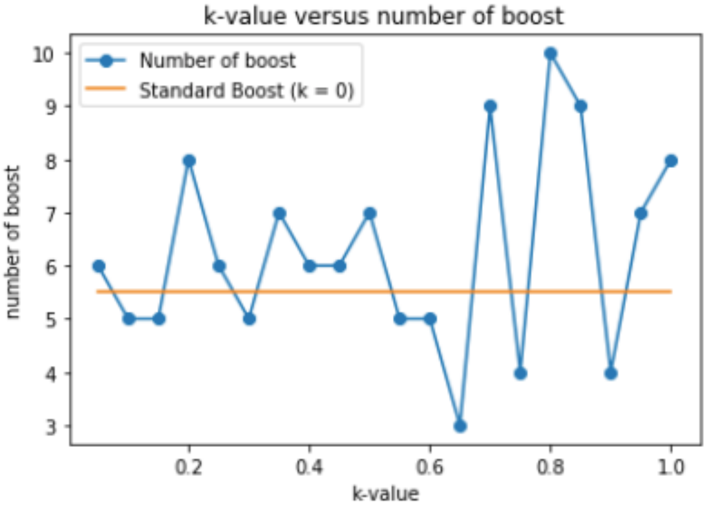
Average Accuracy

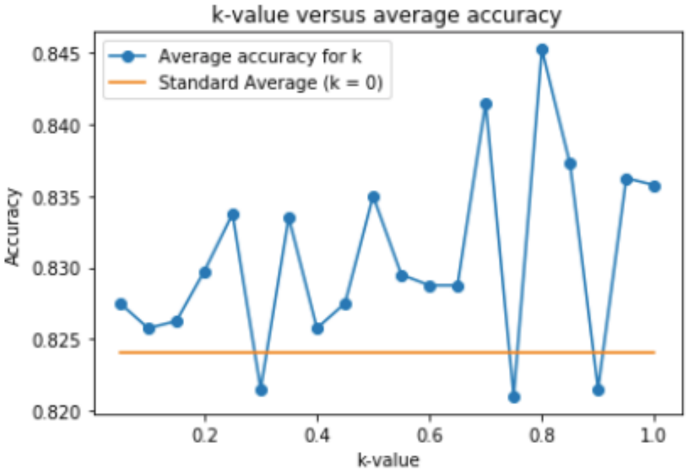
University



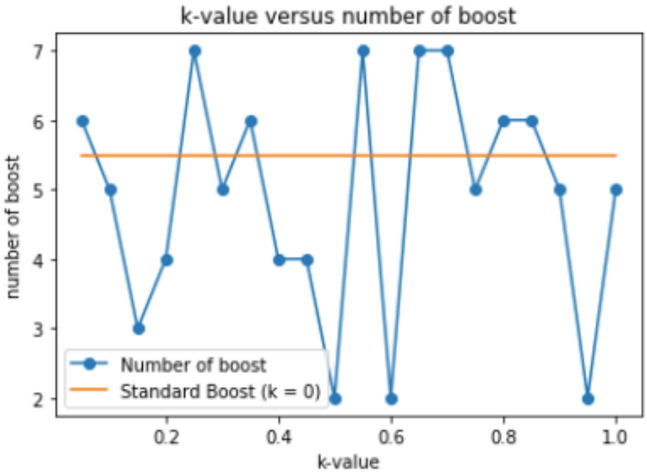


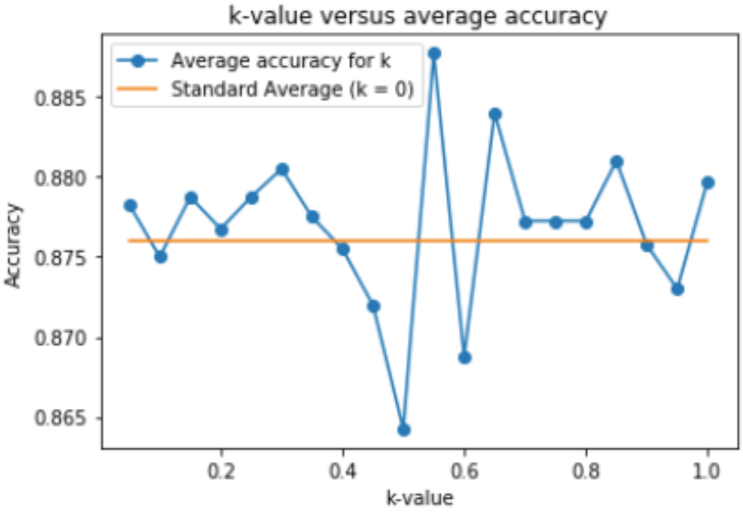
Adult





Bank


k-value versus number of boost
Number of boost
Standard Boost (k = 0)
number of boost
k-value


k-value versus average accuracy
Average accuracy for k
Standard Average (k = 0)
Accuracy
k-value