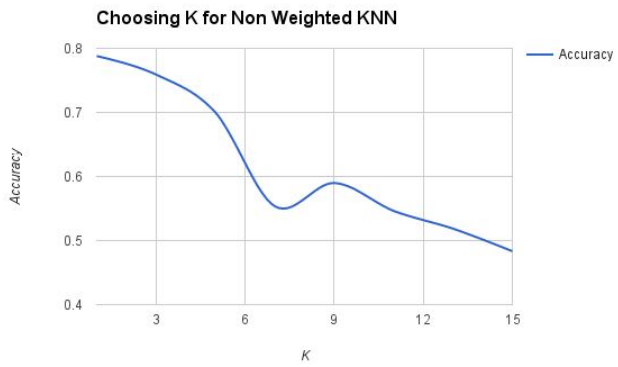
Isai Mercado Oliveros
March 18, 2016
KNN Algorithm

- Use knn without weighting for the magic telescope problem.
- Try it with k=3 with normalization and without normalization and discuss the accuracy.

Rounds	Normalized	Non Normalized
1	0.7589	0.7386
2	0.7568	0.7365
3	0.7617	0.7317
Average	0.7591	0.7356

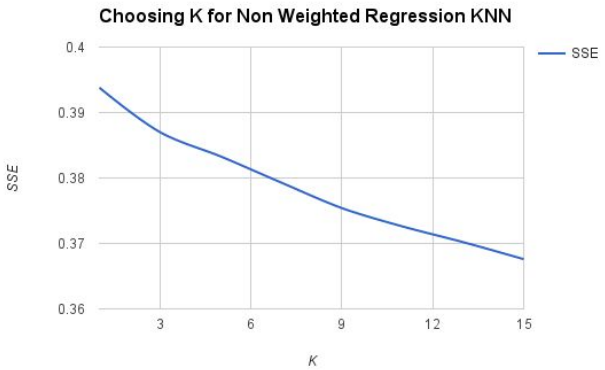
This experiment clearly shows that normalized data generates better results, so we should always normalized our data.

- With just the normalized training set, graph accuracy on the test set with odd values of k from 1 to 15



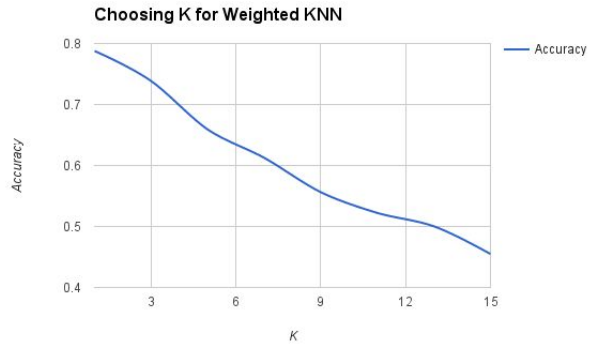
From this graph, I can see that the k nearest neighbor algorithm works better when it classifies based on the single closest neighbor. I think, this is the case because it resists outliers, and if the point is next to the dividing hyperplane, it will not take into consideration the point on the other side of the hyperplane, unless the problem is very hard and the current features do not make a big gap between points from the output classes. However, I do not think that this problem with knn. I think the problem is the lack of features.

- Use the regression knn without weighting for the housing price prediction problem
- Report Mean Square Error on the test set as your accuracy metric for this case.
- Experiment using odd values of k from 1 to 15. Which value of k is the best?

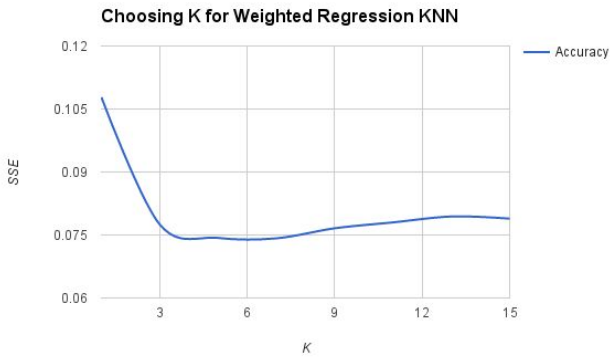


From this graph, I can see that for regression, the error goes down if you use a big K. I think that this happens because it calculates the next point in the space by returning a type of average calculated from the points around it, so the more points it has, the better the average because the effect that points on the edges have in the prediction gets minimized.

- Repeat your experiments for magic telescope and housing using distance-weighted



From this experiment, I learned that There is not too much difference in the accuracy from a non weighted knn and a weighted knn. The only first sight difference, that I can perceive, is that the curve is more smooth. I think this is the case because the prediction of output labels in weighted knn is not so cutting as in the non weighted version of knn.



From this graph I learned that the accuracy of the weighted regression knn is better that the normal regression knn. The curve of the non weighted regression knn starts at error 0.4, and it goes down to error 0.37. However, the curve of this weighted regression knn starts at 0.1 and goes down to 0.075. Another thing that I can notice is that the error starts to increase again if K gets too big. I think this happens because the average of the points cannot center the predicted point that well because points have different vote. However, the overall error is lower.

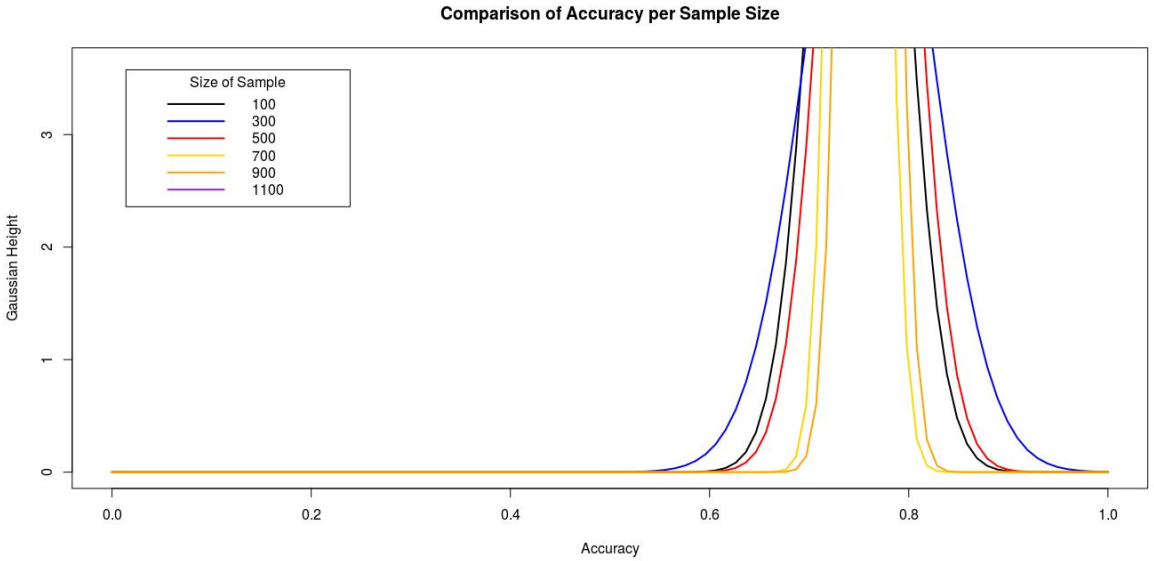
- Use knn to solve the credit-approval data set.
- Implement and justify a distance metric which supports continuous, nominal, and don' t know attribute values.
- Use your own choice for k, training/test split, etc.

My accuracy was 80%

I chose a K of 1 because according to my experiments, classification works better with K = 1. In addition, I run 10 cross fold validation, so my result is the average of 10 random splits. My distance is implemented by the euclidean distance. I handle continuous values as they are. That is... I just let them be their real values in the space. Nominal values are mapped to an integer that let us use the nominal value as a point in the space, and finally don't knows are handled in the following way:
For continuous values I put a -1 since I wanted it to be its own point in space.
For nominal values, I added a new feature DONT_KNOW to be its own point in the space.

I wanted dont-knows to be their own value because I think there is a reason for them to not be there. That reason, whatever it is, might be linked to the predicted class because, for example, in studies, things that cannot be measured are prevalent across the entire study, or, another example, people that do not want to answer surveys, do not want to give an explicit answer for a reason which is related to the output class. It is just that it is implicit rather than explicit, but the machine learning model can use that implicit relationship to predict the output class when it sees another dont-know instance.

- Do an experiment of your own.



In my experiment, I computed prediction accuracy from increasing sample sizes for the telescope data set, and measured variance based on several trials. The graph shows that the variance of the normal curve gets smaller in relation to the increasing size of samples. That means that the result gets more precise with more and more points in the training data set. However, the mean of the accuracy of all the samples does not change a lot, so even with an small sample from the data the expected value would be very close to the accuracy of the whole data set.