Keith Mertan
CS 7641 - Machine Learning
Assignment 1

Supervised Learning Algorithms Analysis

Data

The two datasets chosen for this analysis are focused on Indonesian contraception choices and breast cancer detection, respectively.

The Indonesian contraception choices data is from the UCI Machine Learning Repository and contains 1472 rows, each corresponding to attributes of an Indonesian woman's life and her contraceptive choices. There are nine independent variables. Two of those variables are continuous, six are ordinal categorical variables and the last is a nominal categorical variable. The nominal variable (occupation type of the woman's husband) was split into four dummy variables before fed into each algorithm. The response variable, contraception choice, has three levels: "no-use", "short-term" and "long-term". The majority class is "no-use", constituting 42.7% of the data. This will be used as a baseline to compare algorithms to, as anything lower than 42.7% is worse than guessing "no-use" every time. Although I'm unfamiliar with Indonesian culture, I imagine the practical significance of predicting contraceptive choices in somewhere like the U.S. would be to help mitigate overpopulation and dependence on social welfare programs.

The breast cancer data is from mldata.io, contains 569 rows which each represent a patient, and 10 independent variables. All 10 variables are integers which measure attributes of cell nuclei, such as shape and size. The dependent variable is binary, taking values "malignant" or "benign". The majority class is "benign", making up 62.8% of the data. This will also be used as a benchmark to which we will compare the performance of each algorithm. The data is practically significant because breast cancer is a serious condition and early detection can be extremely beneficial to the patient.

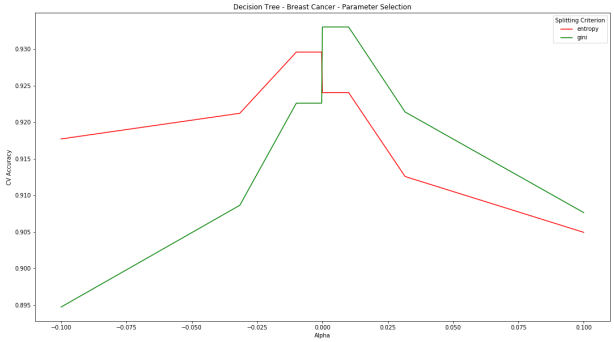
There are two apparent differences between the datasets before running any algorithms on them. First, one is a multi-class classification problem while the other is

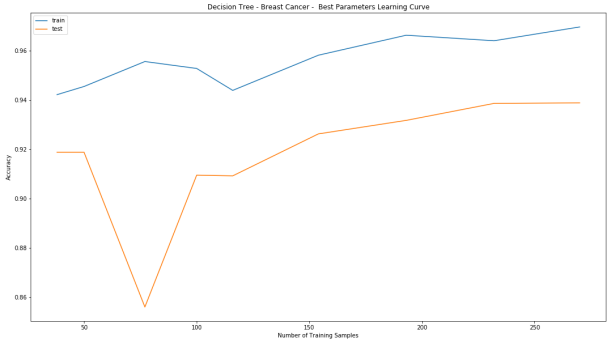
binary. Secondly, the contraceptive data includes categorical data of both ordinal and nominal types while the breast cancer data contains only integers. Before we discuss the differences in terms of algorithm performance, let's quickly introduce the software utilized.

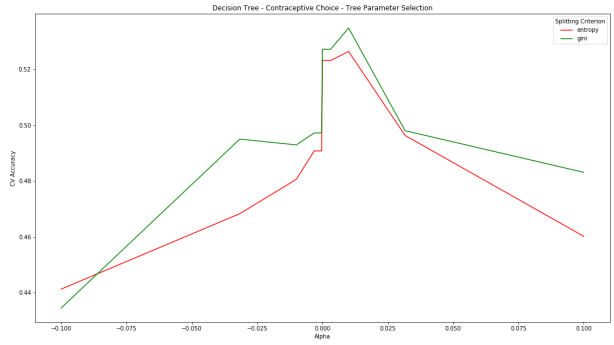
Software and Methods

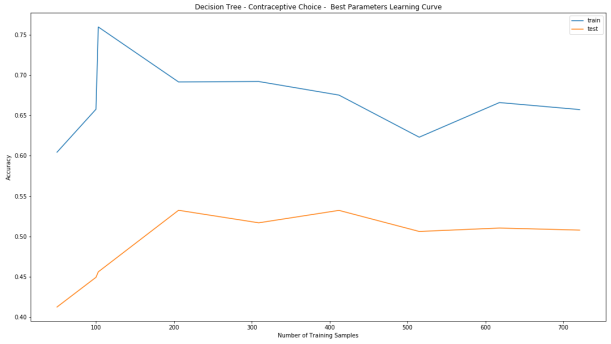
The analysis is performed in Python using Jupyter notebooks. The code is an adaptation of Jonathan Tay's and implements all of the algorithms using the scikit-learn library. GridSearchCV is used for parameter selection using 5-fold cross validation for each dataset and algorithm. Matplotlib is used for plotting both parameter performance and learning curves. All learning curves are produced by running cross validation on the best set of parameters from GridSearchCV. Each data set is split using stratification into 70% training, 30% testing and final test accuracies are reported using balanced accuracy to account for differences in class distributions.

Decision Trees









For decision trees, different values of both the alpha parameter and splitting criterion were varied to select the best with respect to mean cross validation score. Alpha, which acts as a form of pruning by controlling the number of nodes in the tree, is varied from -1 to 0.1. The higher the alpha value, the more aggressive the pruning and therefore the fewer nodes in the tree. The splitting criterion is either entropy or gini impurity. These are scores to help decide which independent variable to split on at each node in the tree, maximizing the information gain in the case of entropy or minimizing impurity in the node for gini.

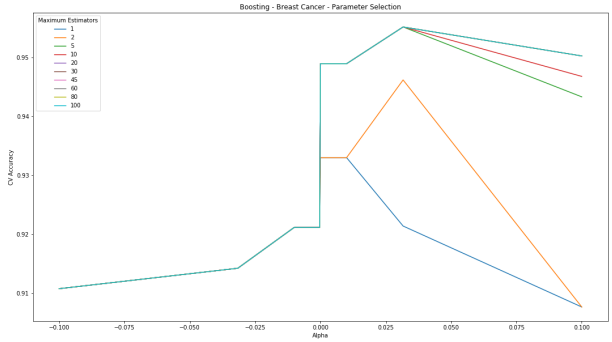
The best parameters for the breast cancer data are an alpha of 0 using gini impurity. Interestingly, as alpha increases past 0 the cross validation accuracy of gini suddenly becomes greater than that of entropy. More aggressive pruning seems to favor gini, suggesting more significant splits toward the top of the tree using that splitting criterion while entropy may have more consistent gains throughout the tree. The training and testing accuracy curves track each other fairly well with a final test accuracy of 96.0%.

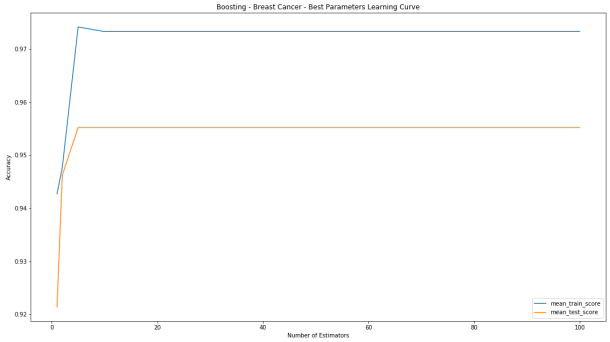
For the contraceptive choice data the best alpha is 0.01, also using gini impurity. This alpha tells us that more aggressive pruning was used to address the problem of overfitting in a fully grown tree. Training and testing curves track each other well for this data too and the tree yields an accuracy of 51.0% on the final test set.

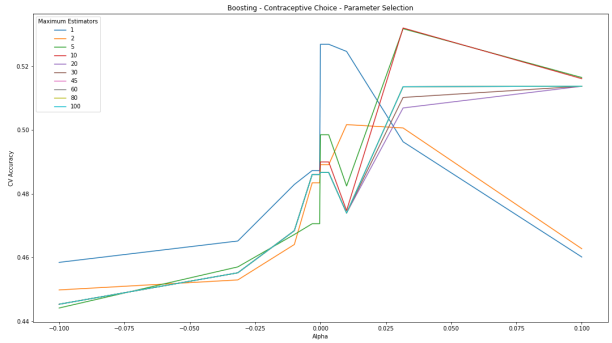
Both decision trees in the case utilize pruning and the gini impurity and do significantly better than the benchmarks set by the proportion of majority class examples.

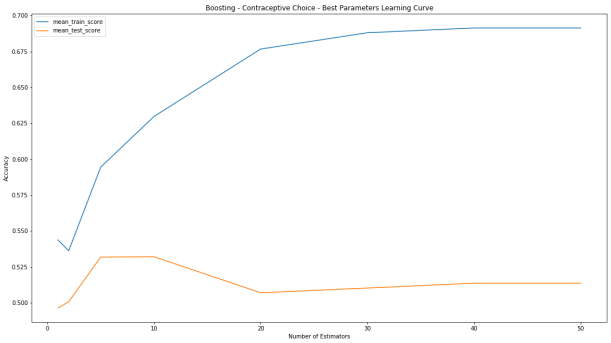
Boosting

Boosting will be discussed next as a follow on to decision trees. Although boosting can use any base estimator to create weak learners, decision trees were chosen for this analysis. An improvement could be to utilize additional base estimators, such as linear SVMs. The parameters that are searched over are alpha and the number of estimators created. Alpha in this case relates to the underlying decision trees and therefore acts as the pruning parameter. Like decision trees, this is varied from -1 to 0.1. The number of estimators is how many decision trees are created. Ten values from 1 to 100 are tested for the number of estimators.









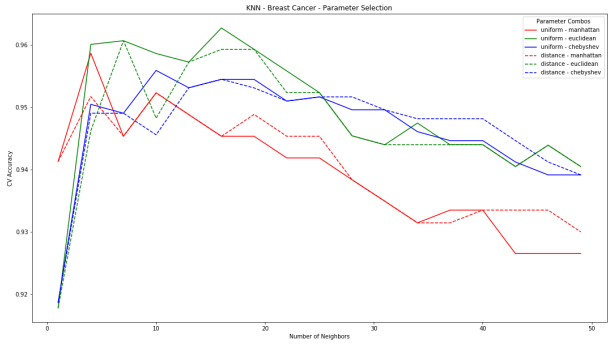
The parameters chosen for the breast cancer data are alpha of approximately 0.03 and 5 boosted trees. Compared to the pruning used for the decision tree on the same data, we see that boosting favors a more aggressive approach and fewer nodes in each tree. This is characteristic of boosting since it is undesirable to have each individual classifier capture too many nuances of the data. The training and testing accuracy remain constant after roughly 5 estimators in the learning curves. The final testing set produces an accuracy of 96.6%, slightly outperforming decision trees.

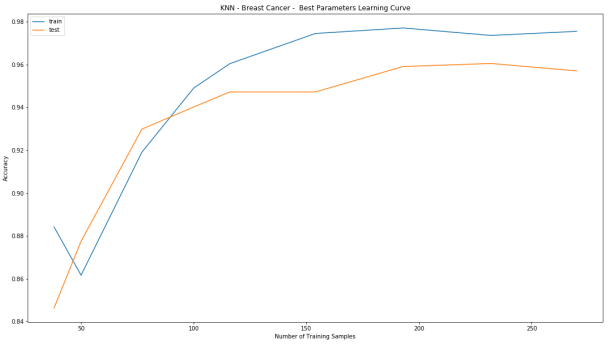
The contraceptive choice data saw its best performance at alpha of approximately 0.03 and 10 boosted trees. Compared to its decision tree counterpart we see more aggressive pruning again. Interestingly, the mean test accuracy drops after 10 estimators in the learning curves while the training accuracy continues to rise, signifying an overfit to the training data. Although boosting is generally robust to overfitting, the logical explanation for its occurrence in this case is noise in the data. At the end of the day, contraceptive choice is not necessarily a deterministic function of socioeconomic factors. The ensemble method is doing well at understanding the intricacies of the training data but unfortunately that data may not fully represent the ground truth of how contraception choice is made. To address this overfit, the ensemble with the best cross validation score should be used and all estimators added

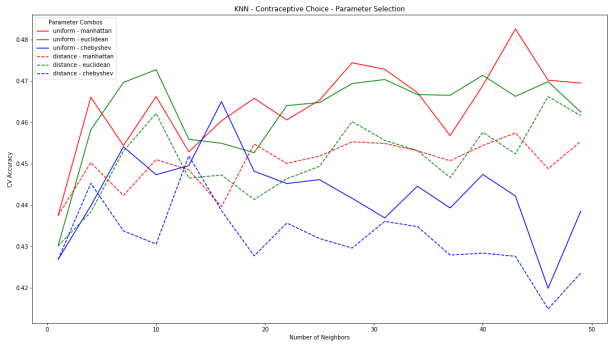
afterward should be discarded. The accuracy on the testing set is 55.0% which is a significant boost compared to decision trees (pun fully intended).

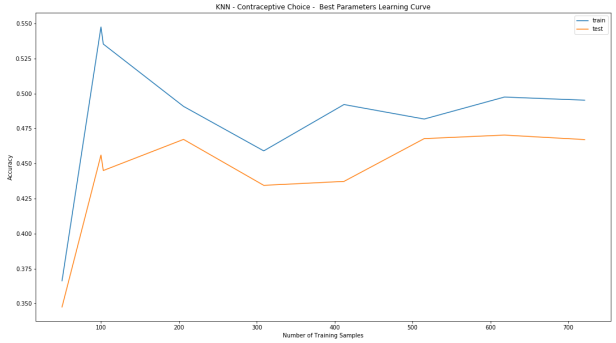
For both trees we see a large jump in performance after alpha of 0. Again, boosting favors weak learners that do not overfit to the training data. It should be noted that the contraceptive choice data requires double the number of estimators to achieve optimal performance. Intuitively this makes sense. The breast cancer data captures a physical phenomenon (relating physical cell characteristics to cancer diagnosis). On the other hand, contraceptive choice is influenced by many more factors and likely inherently have far greater variability. Boosting helps represent this variability by fitting additional weak learners to the data.

KNN









The K-Nearest Neighbors algorithm relates data points based on some measure of proximity to one another. Three parameters are varied in this algorithm: distance metric, number of neighbors and weights. The distance metric is how we score how “close” points are to one another. The three values it takes are the Manhattan distance,

Euclidean distance and Chebyshev distance. The number of neighbors (K) is the closest K points to be considered when classifying a new point. Values from 1 to 49 are considered here. Finally, the weights are how the influence of each neighbor is calculated. Weights are either uniform, where every point is weighted equally, or distance weighted, where values of the chosen distance metric are used to make closer points more influential.

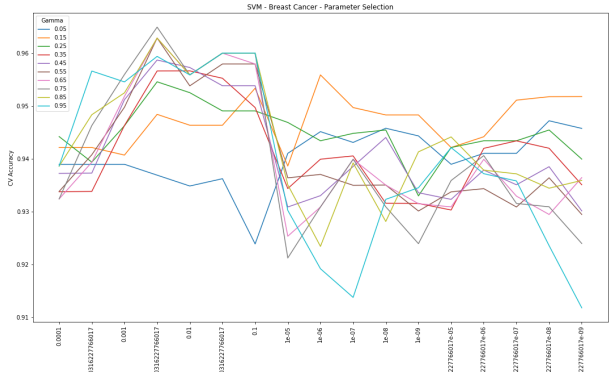
The breast cancer data favors using 16 nearest neighbors, uniformly weighted and measured by Euclidean distance. The predictive ability of the model quickly falls as the number of neighbors is increased past this optimal point. Since the data describes something physical this makes sense. The more “different” the cells being measured are from one another, the less likely they are to be related in terms of cancer diagnosis. Uniform weights outperformed distance weighted calculations for each distance metric as well. As for the learning curves, both training and testing accuracies increase almost monotonically. This suggests little noise in the data. As the model sees more examples, it can more accurately assign classes to test points since they are representative of the underlying target concept. The test accuracy of the model is 96.6%. This compares favorably to other models and shows that similarity is relevant in the case of cell attributes related to cancer diagnosis.

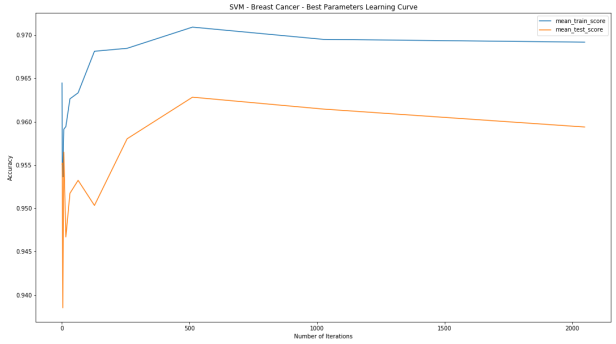
For the contraceptive choice data, the optimal parameters chosen are 43 nearest neighbors, uniformly weighted and measured by the Manhattan distance. The learning curves show that accuracy begins to drop after only about 100 training examples. Since the data is representative of human behaviors, we see that noise affecting performance. Different “neighbors” act with high variability, therefore more data does not imply improved performance in this case. The testing accuracy of the model is 48.6%, showing that similarity of points is less relevant in the case of contraception choice when compared to other models.

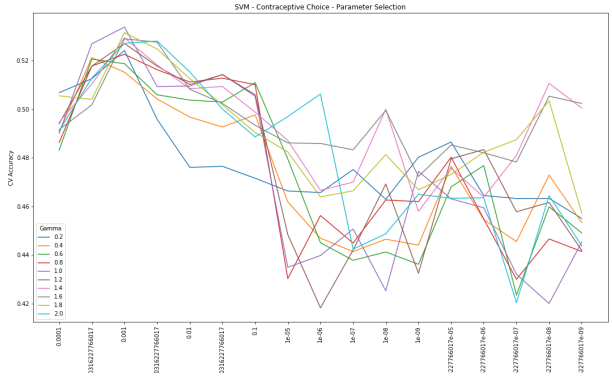
Comparing the performance of KNN on the two datasets, we clearly see that the contraceptive data has much more accuracy variance across the potential parameters than the breast cancer data. This, again, speaks to the difference in noise between the datasets. Another way to view this is by comparing the number of neighbors. With more than twice the number of neighbors required for the contraceptive data than the breast cancer data, contraception choice is clearly a more noisy decision than cancer diagnosis. Both algorithms favor uniform weights. This indicates that the relative similarity of points is not as important as the aggregate of all the neighbors.

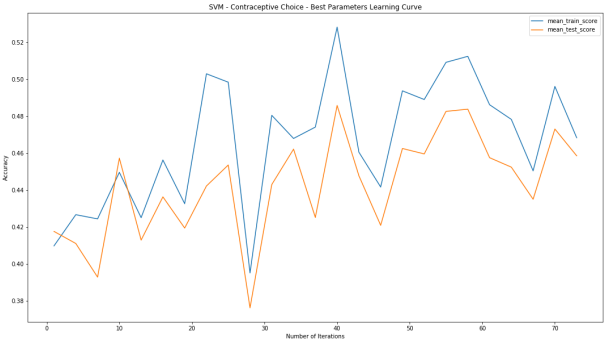
SVM

The support vector machine algorithm creates hyperplanes to separate the data into classes given a kernel function which allows it to form a variety of decision surfaces with different capabilities. The parameters tested for this algorithm are alpha, gamma and the kernel function. Alpha is used to calculate the learning rate and balance model complexity versus accuracy. This parameter is varied from 1e-9 to 0.1. Gamma controls the influence of points on the support vectors based on the distance to the decision boundary. The higher the gamma, the farther away the influence of points reaches. This value is varied from 0.05 to 0.95 for the breast cancer data and from 0 to 2 for the contraception data. Admittedly, these values are carried over from Jonathan Tay's code and could be improved by intentionally selecting them. Lastly, the kernel is chosen as either linear or radial basis function (RBF or Gaussian). Although both were tested for this analysis, it was realized that the linear kernel is a degenerate version of RBF and will never outperform RBF when both are properly tuned. Therefore, results shown are only for RBF. Another improvement would be to test other kernels, such as polynomial.









In the parameter selection charts above, please note that the x-axes represent alpha and are unfortunately not in order numerically. To deal with scaling of the searched alphas, their values were converted to strings to evenly space them across the axis.

The best parameters for the breast cancer data are an alpha of approximately 0.003 and a gamma of 0.75. This fairly high gamma value indicates that points far from the boundary are still significant in determining how to best separate the data. Learning curves for both training and testing quickly increase with the number of iterations but accuracies begin to fall slightly after about 500. The final testing accuracy on this data is an impressive 97.4%.

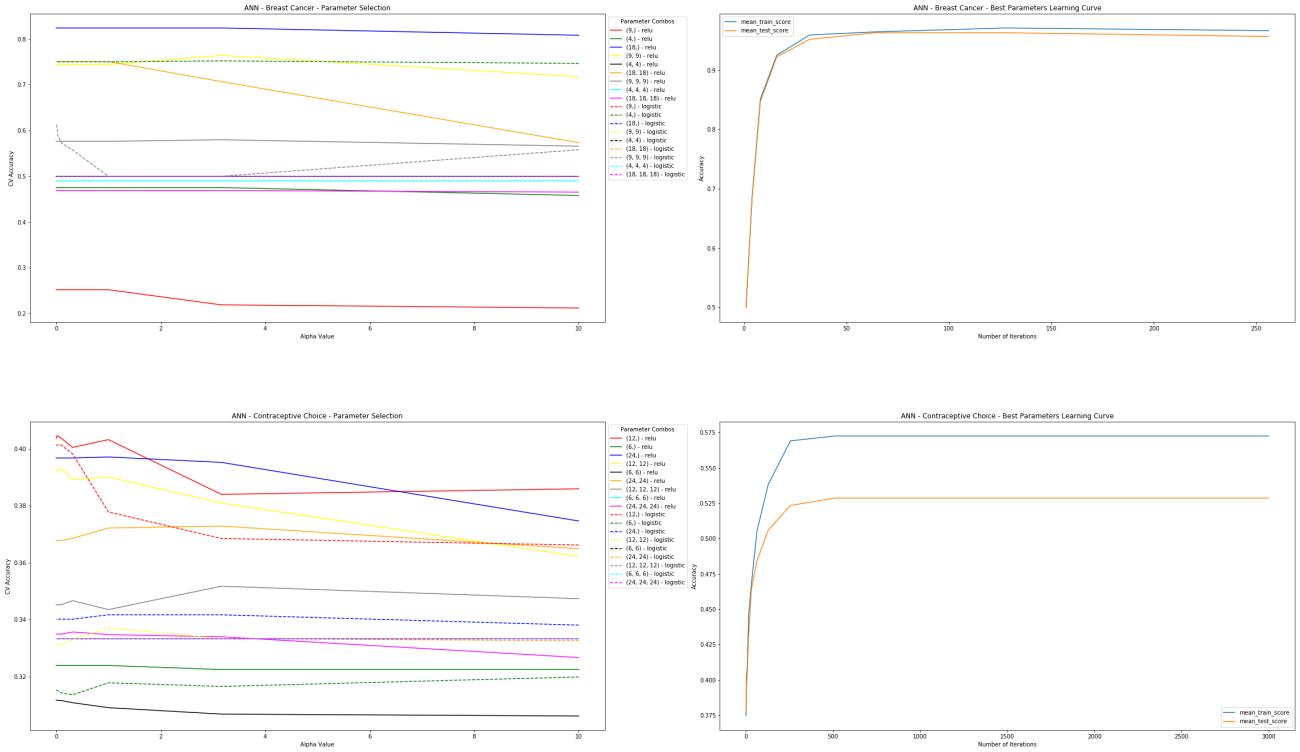
The contraception data sees its best performance when alpha is 0.001 and gamma is set to 1.0. It is clear to see that performance drops significantly when using smaller alpha values. The learning curves track each other well and, despite heavy fluctuation, see an overall increase with the number of iterations. With a middle-of-the-pack gamma, the algorithm tells us that the data far from the decision boundary shouldn't be considered as heavily as the close and medium range points. Since we're looking at human activity data, this makes intuitive sense once again. This suggests that outliers far from the boundary don't represent the most generalizable hypothesis function and should therefore be given less weight. The final accuracy on the test set is 53.6%.

Comparing the performance of SVMs on the two datasets, we once again see the difference in data that models a physical event versus a social event. In the breast cancer case, we give significant weight to points far from the boundary because they still accurately predict the cancer diagnosis. The noise in the contraception data forces the model to give less weight to extreme values to avoid overfitting. Additionally, the alpha value of the contraception data is three times that of the breast cancer data. Since alpha is multiplied by the penalty (hinge loss in this case), this shows that the breast cancer data punishes the model much more severely for misclassified points. The contraception model allows for greater model complexity given its smaller alpha value, which speaks to the complicated nature of data surrounding social decisions.

ANN

The parameters tested for Artificial Neural Networks include alpha, the activation function and the shape and size of the hidden layers. Alpha, or the learning rate, determines how drastically the weights change during backpropagation. The values

tested in this analysis are between 1e-5 and 10. The activation function allows the network to model non-linearity and is chosen from relu or logistic (or sigmoid) activations. The hidden layers of the network vary in both the depth (number of layers) and size (number of neurons in each layer) depending on the size of our data. Since our two datasets are of different sizes, values for each can be seen in the parameter selection plots below.



For the breast cancer data, a single layer network with 18 neurons, alpha of approximately 3.16 and relu activation performed the best. Interestingly, the network seemed to perform worse as the number of layers increased. Since these networks can model more complex relationships with deeper networks, this suggests that such complexity is unnecessary for this data and deeper models overfit. The choice of alpha in this case seems practically insignificant when looking at the parameter selection chart. This means that the aggressiveness of the weight updates in backpropagation is having little effect on the model's learning ability. This points again to the nature of the data. With little noise in this physical phenomenon, the model doesn't tend to fall into local minima throughout the training process and can update to the global minima

given nearly any learning rate. On the final testing set this algorithm achieves 94.9% accuracy.

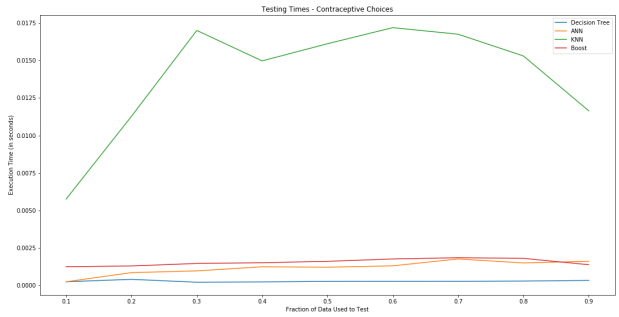
The contraceptive choice data gets its best performance from a single layer network with 12 neurons, alpha of approximately 0.03 and the relu activation function. The performance of the network varies greatly across hidden layer size and shape combinations likely due to the inherent noise in the data. The single layer it chose allows it to avoid overfitting. The learning rate is small and suggests that there are plenty of local minima acting as pitfalls for the algorithm to fall into. In most of the parameter selection curves, we see decreased performance as alpha increases. If the learning rate is too high the algorithm bypasses its optimal hidden layer weights or may bounce back and forth on either side of optimal. By having a low learning rate, the algorithm slowly approaches a minimum error with its weight updates. The most important observation about this algorithm performed on this dataset is the final testing accuracy, which comes in at a mere 38.02%. This is the only algorithm which performs **worse** than our baseline case, which is to always guess the majority class. An explanation for this may come from the fact that neural networks generally require large amounts of data to train. In this case, our 1472 samples and 13 independent variables (after dummy encoding) may simply cause this dataset and algorithm combination to suffer from the curse of dimensionality.

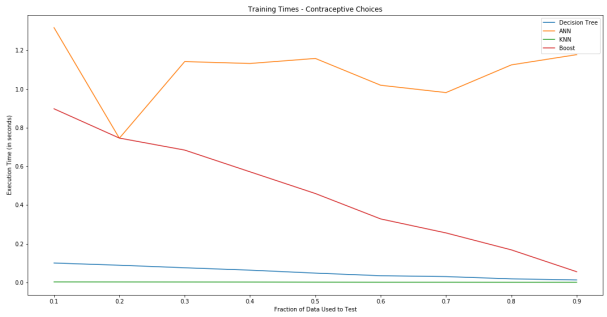
Both datasets favor a network with relu activation. This is not surprising, as recent developments in machine learning point to consistent outperformance of sigmoid activations by relu activations to avoid problems such as the vanishing gradient. Both networks also favor shallow networks, but their reaction to varying alpha values is very different. The breast cancer data's network can update its weights much more quickly and confidently than that of the contraceptive choice data, which most slowly approach its minimum error. The two networks also stabilize nicely, though the breast cancer model generalizes much better than the contraceptive model.

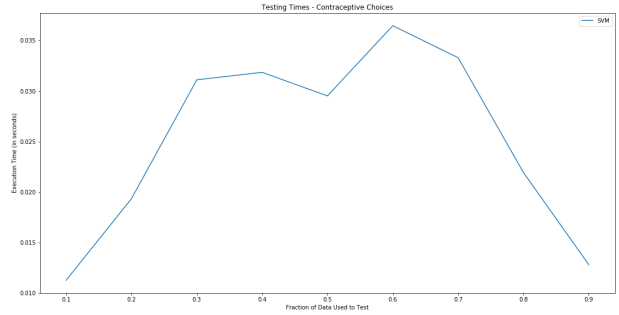
Time Complexity

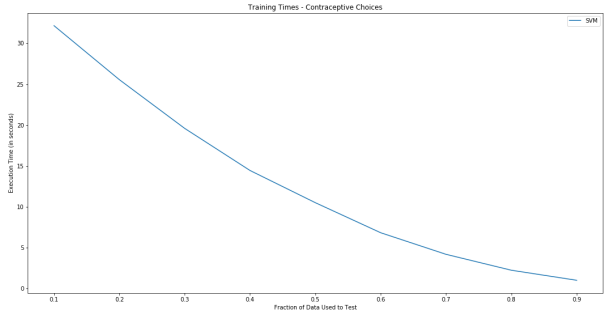
Below are the training and testing times of each algorithm, plotted against the fraction of data used to test. Given the similarity of the plots between the contraceptive data and breast cancer data, only the contraceptive times are shown. Four of the algorithms are plotted on the first row with SVMs given their own row as the training

times are on a vastly different scale. Please mind the scaling of the y-axes when comparing.









One interesting point to note is that KNN is the only algorithm with a smaller testing time than training time. As described in the lectures, this is a trait of a lazy learner which does little to train and much to test. Another notable observation is that SVM training time increases exponentially as the fraction of data used to train increases (in this case we see an exponential decrease versus the testing fraction). This results in a roughly 30 second training time at worst and will be troublesome when dealing with larger datasets. Boosting and Decision Trees have constant testing times and training times for both increase linearly with the amount of training data. ANNs take the longest time to train second to SVMs and have a linearly increasing testing time.

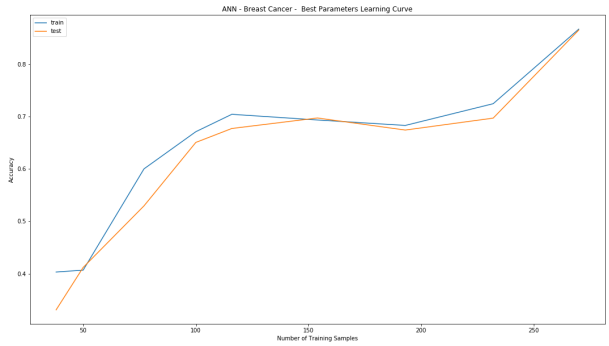
Best and Worst Performing Algorithms

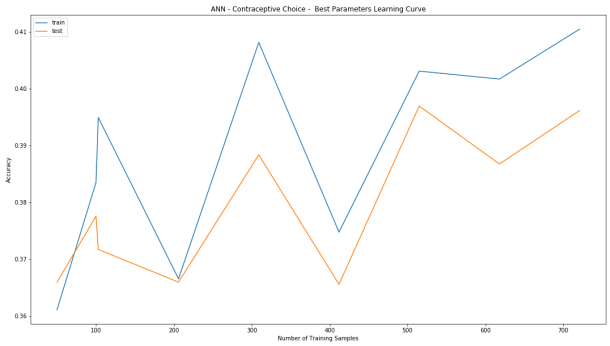
The two datasets saw varying performance across all algorithms. The breast cancer data was best suited for support vector machines and saw its poorest

performance on artificial neural networks. The contraceptive data performed best with boosted decision trees and also performed the worst on neural networks.

Since the SVM implemented uses an RBF kernel, its flexible decision boundary has allowed it to train and generalize very well on the breast cancer data. Boosting performed well on the contraceptive data likely because it intentionally focuses on correctly classifying the “hard” data points with each additional classifier in the ensemble, allowing it to discover nuances which are unrecognizable by the other algorithms. Perhaps more interesting than the best performing models, in this case, are the worst performing models.

Neural networks, though often glorified as the hallmark of modern machine learning, seem to have dropped the ball for both of these datasets. So are these algorithms simply overrated? Let us look at the learning curves for them, relative to our two datasets, from a different perspective.





When looking at the learning curves against the number of training samples instead of the number of iterations of the algorithm, we see that the number of training samples for each is just not enough to allow the accuracies to converge. Particularly in the case of the breast cancer data, the slope of the line toward the end of the x-axis values suggests that there is great room for improvement if we were to provide the algorithm more data. The contraceptive data has more variance to its accuracy as the number of training samples increases but also seems to suggest that given additional data, the algorithm could see a jump in performance.