CS 7641: Assignment 1 Supervised Learning

Angelo Petrolino
apetrolino3@gatech.edu

I. Introduction - Dataset Explanation

Here we introduce two interesting classification problems approached with four supervised learners. The first dataset, the "Rice (Cammeo and Osmancik)" dataset [1], was chosen for its entirely numerical features and binary target class. The "Estimation of Obesity Levels Based On Eating Habits and Physical Condition" dataset [2], was chosen for opposing characteristics relative to the first dataset: mixed features (categorical and numerical) with a multi-class target variable.

For the Obesity Dataset, due to its purpose as an aid in medical evaluation and the impact of false negatives in that field, the metric that will be focused on is Recall. Because of the reasonably balanced classes, it will be a Macro Recall metric. The Boosted Decision Tree learner's ability to reduce variance and bias with non-linear data through an ensemble of weak learners is expected display the greatest performance across multiple metrics. With correct tuning to avoid overfitting, the Neural Network learner is expected to be the best performing learner by accuracy. A nonlinear-kernel SVM classifier will perform well with this dataset if it's features are correlated, but due to sensitivity to kernel choice and scaling it should be less robust than the Neural Network learner in most metrics. The k-NN learner is expected to be the least performant with this higher-dimensional dataset, further exacerbated if the data exhibits noise or irrelevant attributes. This dataset was pre-processed with one-hot encoding of the categorical features.

For the Rice Dataset, many of the features are continuous (area, perimeter, etc.), it's sufficiently balanced (57% Class 1 and 43% Class 2) and has no missing/garbage values. This dataset was chosen as a juxtaposition to the prior dataset to explore different model learning methods and compare. Here, we will measure model performance by accuracy. The Neural Network learner should outperform the others in this more sterile environment, but will trade off effectiveness for slower training times. The SVM learner should be the next best performing learner due to a high possibility of easily separable data. The k-NN learner will be the weakest if the dimensionality of the dataset proves high enough. Boosting for Decision Trees could end up beating Neural Network for best performer due to binary class data which Decision Trees have little trouble with.

II. KNN

We tune K first as it controls the bias-variance trade off. We should expect at larger values of K to have a larger loss in variance for larger gains in bias [6].

A. Obesity Dataset

Expected behavior is displayed in 2, but a choice is made in the area of $[K = 2, K = 20]$ (as $K = 1$ is overfit and values > 20 underfit). With $K = 10$, it follows expected behavior that the Manhattan distance metric had the best recall as it's robust to the distribution of data and does not overemphasize outliers that could be present. As for the weighting; due to the data's multi-class nature and the likelihood of local points carrying more relevant information than distant ones, the 'distance' weighting should outperform the 'uniform' weighting. That is the behavior we see here I. The tuned learner exhibits what seems to be overfitting 3, but the behavior of the training recall could be attributed to the 'distance' metric.

Most interestingly, the model's ability to perfectly memorize training data with a low K and a 'distance' weighting implies that the data in this dataset is locally related to each other and there may be little overlap of classes.

TABLE I: k-NN Distance Metric Recall (Obese)

Weighting	**Training Recall**	**Cross-Validation Recall**
Uniform	0.88661056	0.85727988
Distance	1.0	0.89030106

B. Rice Dataset

A quick N neighbors search shows 4 K values under 5 heavily overfit and values over 60 is an asymptotic convergence. The best performing distance metric was unexpected (Canberra) 5. However, the correlation distance metrics performance indicates that the datasets features do, in fact, heavily correlate with each other. So it must follow that heavily correlated features exhibit locality and so a 'distance' weighting should out-perform 'uniform' weighting, and it does II. The trajectory of our

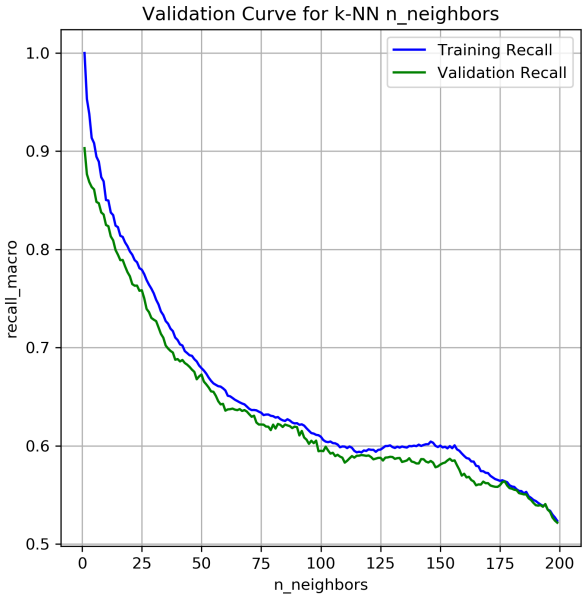


Fig. 1: N Neighbors Validation Curve

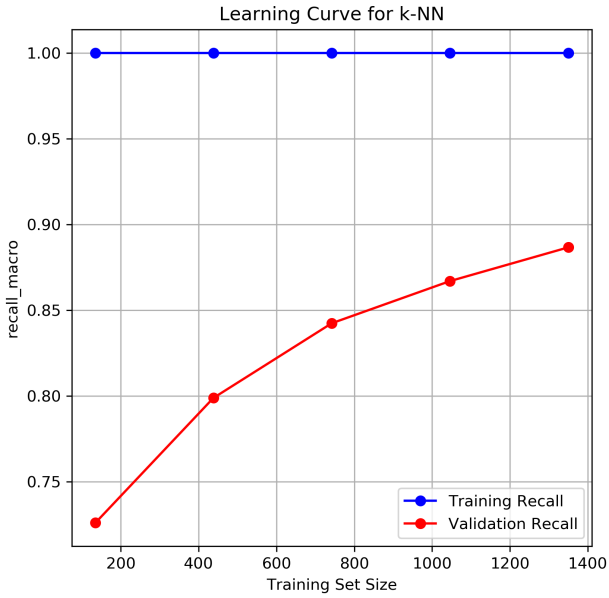


Fig. 3: Tuned k-NN Learning Curve

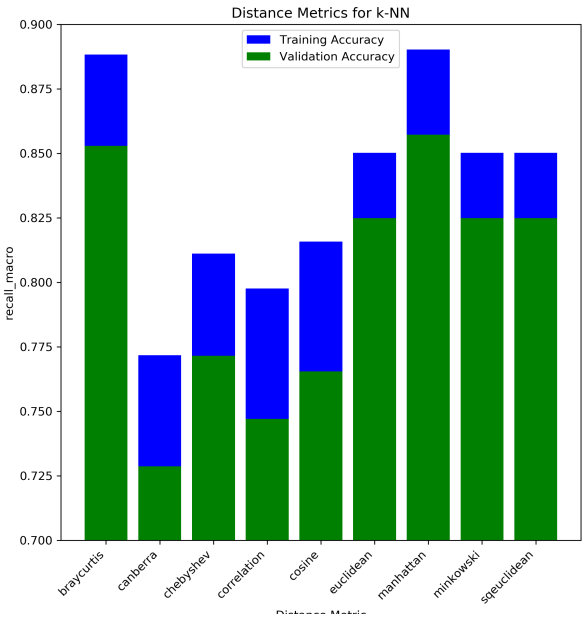


Fig. 2: k-NN Distance Metrics

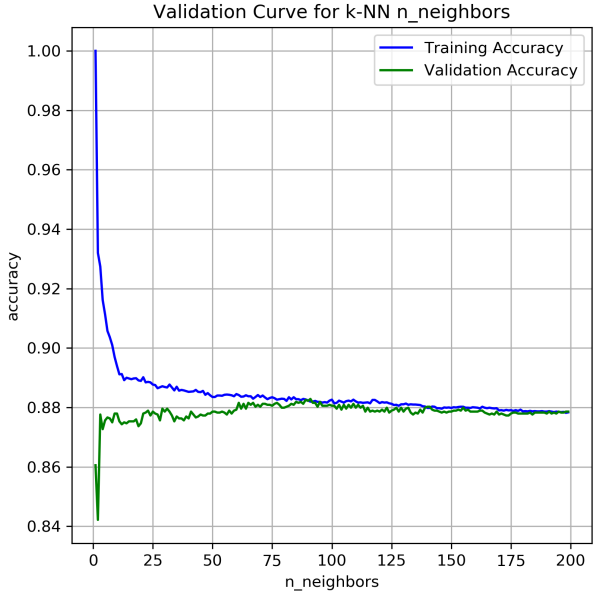


Fig. 4: N Neighbors Validation Curve

validation curve in our learning curve suggests more data won't help, but both curves are close enough to deny overfitting 6.

III. Support Vector Machines

A. Obesity Dataset

An "out of the box" SVM learner performs with 53% on this dataset. Though it performs better than chance, the first hyperparameter will be the kernel type as it will yield different assumption of the data such as its linear separability [4]. Expecting likely non-linearity in the dataset, the RBF kernel should display the best performance. However 7 implies the dataset is, in fact, highly linearly separable. The C hyperparameter should display worse performance against training data because of the larger margin (implying better generalization) and vice versa for larger C's. In 8 we can see training score increase with C as predicted and weaker ability to generalize at higher C from the dip in the cross-

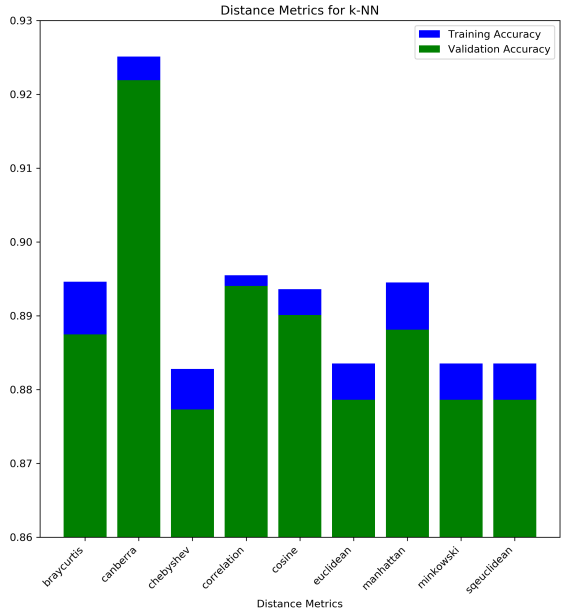


Fig. 5: k-NN Distance Metrics

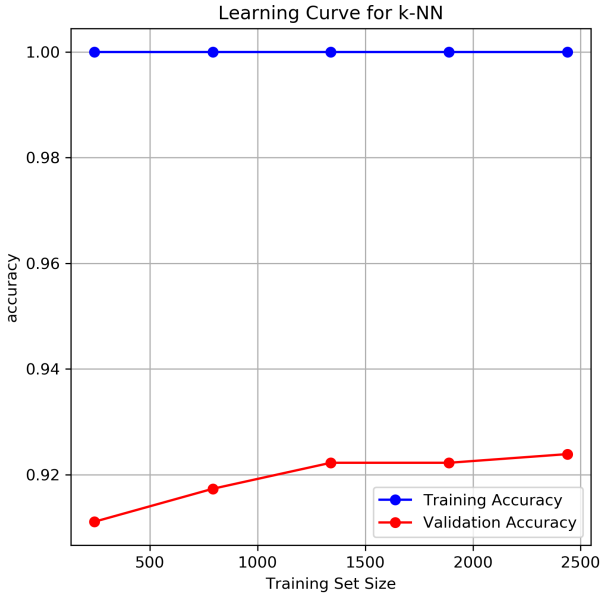


Fig. 6: Tuned SVM Learning Curve

validation curve.

B. Rice Dataset

As this is the more linear dataset, I predict the linear kernel will far outperform the other. This validation sweep confirms the linear kernel is optimal 10. An oddity here is validation values that are at least equal, but possibly greater than, training values across all kernels (the training values are, in fact, there). Our validation curve suggests strong generalizing, so the model can

TABLE II: k-NN Distance Metric Recall (Rice)

Weighting	**Training Accuracy**	**Cross-Validation Accuracy**
Uniform	0.92511462	0.92191714
Distance	1.0	0.92388543

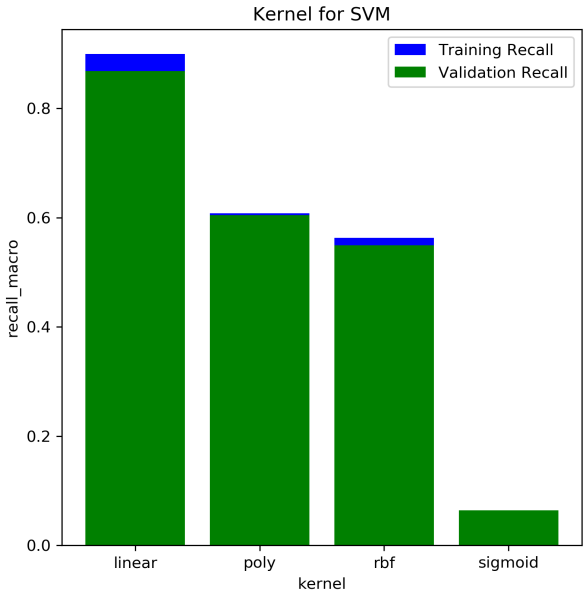


Fig. 7: SVM Kernels by Recall

get away with a higher C without being in danger of overfitting. The results 11 show an increase in C. With the final parameter tuned, we observe it's learning curve 12. It overfits on smaller sets, but has good generalization on larger sets, and the consistent gap between the curves could signify minor overfitting.

IV. Neural Networks

Neural Network hidden layer dimensions were found by tuning for depth and then width sequentially.

A. Obesity Dataset

While training this learner, many iterations ended without reaching convergence. Adjusting for *learning_rate_scheduler* to compensate, an interesting find occurs: though both *invscaling* and *adaptive* schedulers should reach convergence more consistently than *constant* through more aggressive or dynamic learning steps respectively, yet all 3 rate schedulers return nearly identical results. This is possibly due to early convergence and overfitting. If the model is complex enough to overfit the data within a few iterations, all rate schedulers would lead to a similar

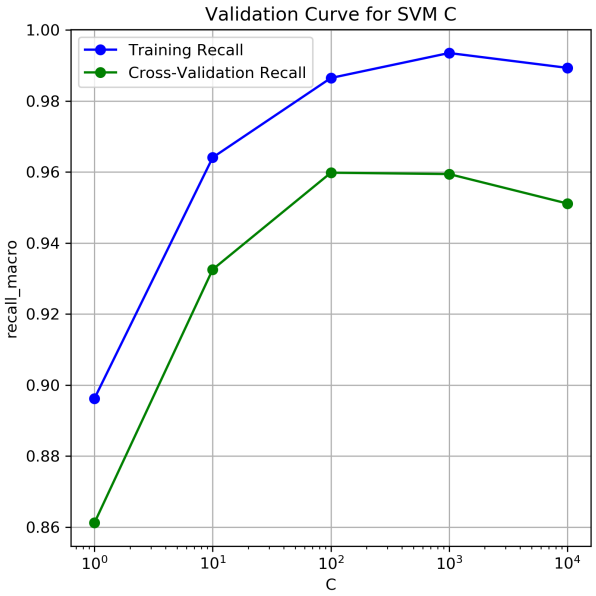


Fig. 8: Validation Curve for C

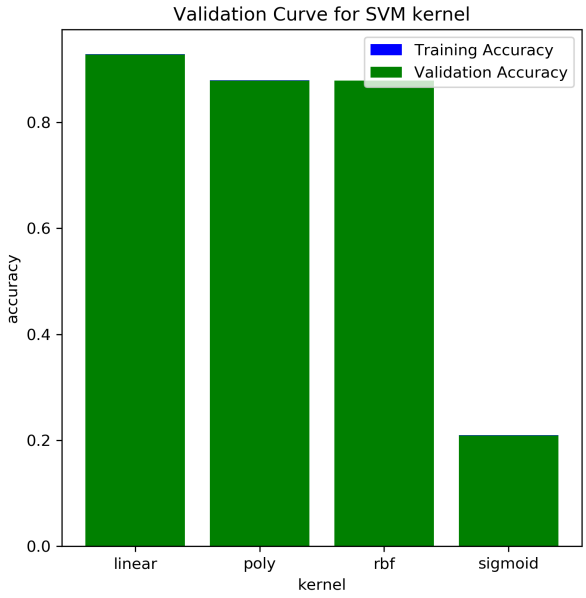


Fig. 10: SVM Kernels by Accuracy

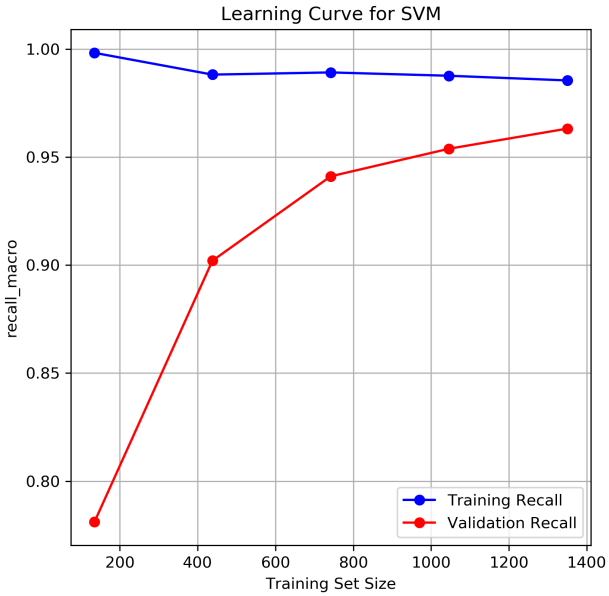


Fig. 9: Tuned SVM Learning Curve

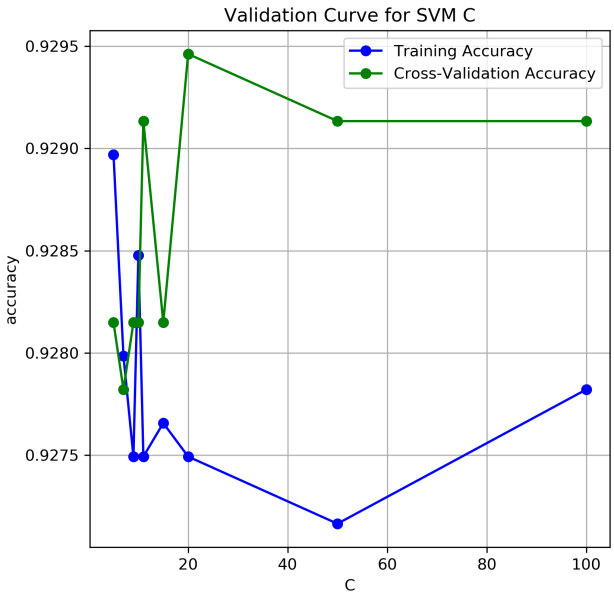


Fig. 11: SVM C by Accuracy

solution since the problem is the model is memorizing data rather than generalizing. We can see from the loss curve 14b that overfitting does occur and within the first 20 epochs. This evidence of overfitting is the motivation to tune $L2\_regularization(\alpha)$, as it primarily addresses overfitting [3] and compensate with a larger *learning_rate*. Gridsearch across both parameters give an $\alpha = 0.5$ and $learning\_rate = 0.01$ giving us this result 15 which exhibits quick convergence and a stable Cross-Validation Loss around 10 epochs.

B. Rice Dataset

Exhibiting similar behaviors as when applied to the Obese Dataset, the *learning_rate_scheduler* is constant across all options. To explore a different approach, *learning_rate* is the next parameter to be tuned so to speed up the tuning following this. A value to high will lead to instability during training, and too small leads to convergence issues. A smaller value was expected since there were slight signs of overfitting, but a larger value was calculated to be optimal 17. It's possible that the

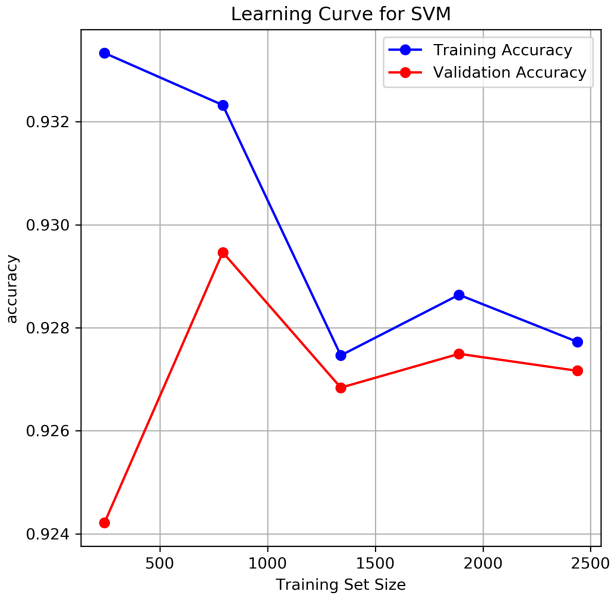
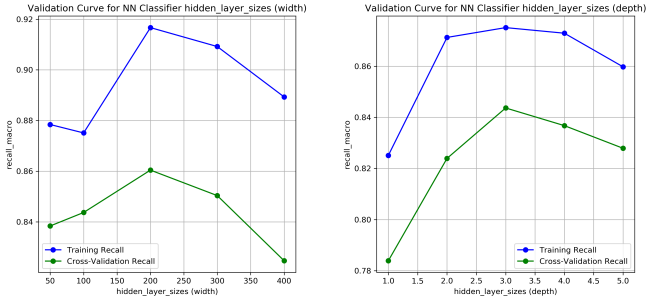


Fig. 12: Tuned SVM Learning Curve



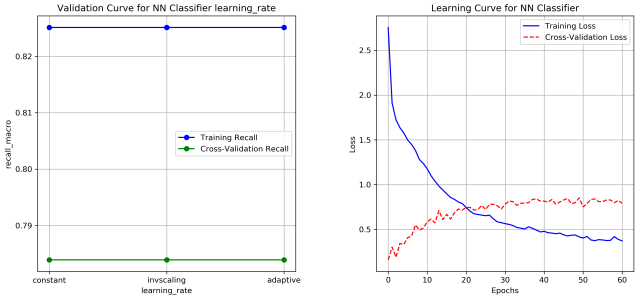
(a) Width (b) Depth

Fig. 13: Hidden Layer Dimensions Validation Curves

model is underfitting in some scenarios. Next parameter is the $activation_function$. As this dataset it more ordered, I predict the Identity function will outperform due to the linearity in the data itself. The results are here 18 and the learning curve our finalized model outputs here 19. We see signs of underfitting also slow growth past 10 epochs.

V. Boosting For Decision Trees

The approach is to use AdaBoost as the booster. We are going to also apply pruning, as each decision tree weak learner tends to overfit as they reach their full 'depth'. Keeping them 'shallow' traps them down as weak learners (keeps them simple), and it is the error from these multiple weak learners that boosting can take advantage of. The first hyperparameter to be tuned will be the weak learner's hyperparameter not the AdaBoost model.



(a) Learning rate scheduler (b) Loss Curve with Cross-Validation

Fig. 14: Rate scheduler tuning and loss curve for identical model

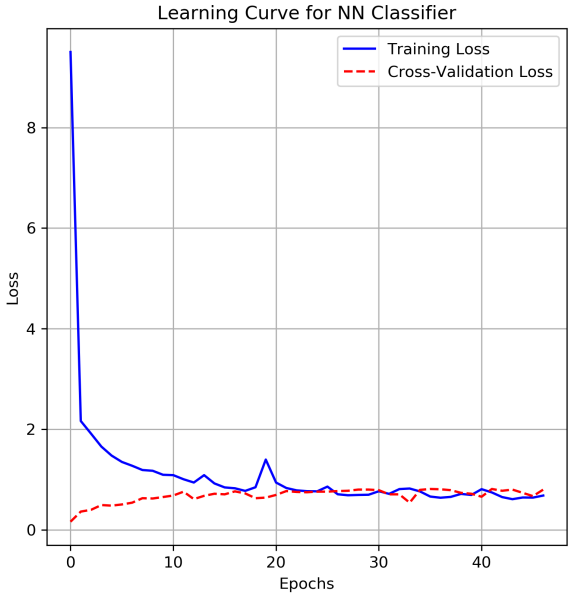
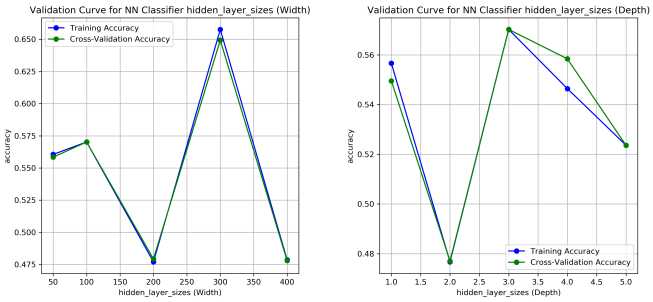


Fig. 15: Loss Curve with Cross-Validation

A. Rice Dataset

Sweeping over values of *max_depth* shows a trade of bias and variance already in 20. At $max\_depth = 1$, both curves are closest - indicating the point of highest bias in this sweep. At $max\_depth = 3$, both curves are farthest - indicating the point of highest variance in this sweep. It's less obvious as we tune for *min_samples_leaf* in 21. These are both examples of variance vs bias trade off *within* a weak learner. We can now tune for *n_estimators* and see how the weak learners themselves affect the Boost model 22. At low values of *n_estimators* (a small number of weak learners), we see high bias in the Boost model's accuracy and at high values of *n_estimators*, we see the reverse. Which highlights an important relationship: **adding more weak learners decreases bias but increases variance**.



(a) Width (b) Depth

Fig. 16: Hidden Layer Dimensions Validation Curves

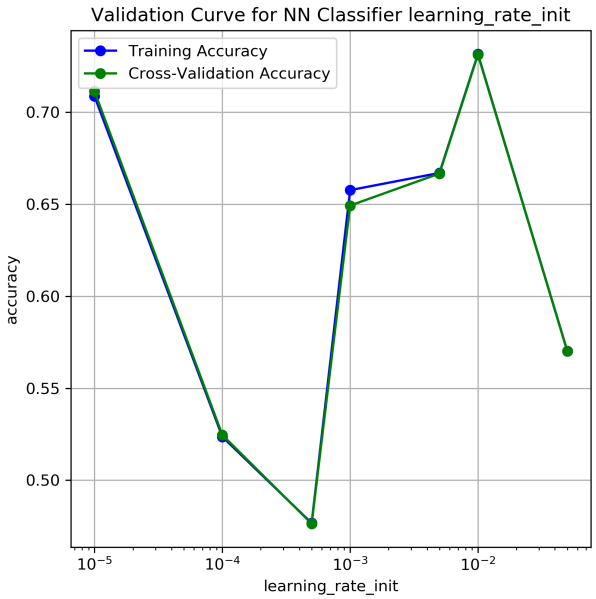


Fig. 17: NN Learning Rate Validation Curve

B. Obesity Dataset

Again, we sweep through $max\_depth = 1$ and examine something interesting 24: **much like how adding weak learners increases variance in the AdaBoost model, adding more depth increases variance in the weak learner.** We see similar behavior when sweeping across $min\_samples\_leaf$ as we did with the Obese dataset.

VI. Conclusion

For the Obesity dataset, we predicted performance in the order of: Neural Network, SVM, and k-NN. The results IV show 0 out of 4 predictions were correct. For the Rice dataset, we predicted performance in the order of: Neural Network, SVM, and k-NN. The results III show 2 out of 4 predictions were correct.

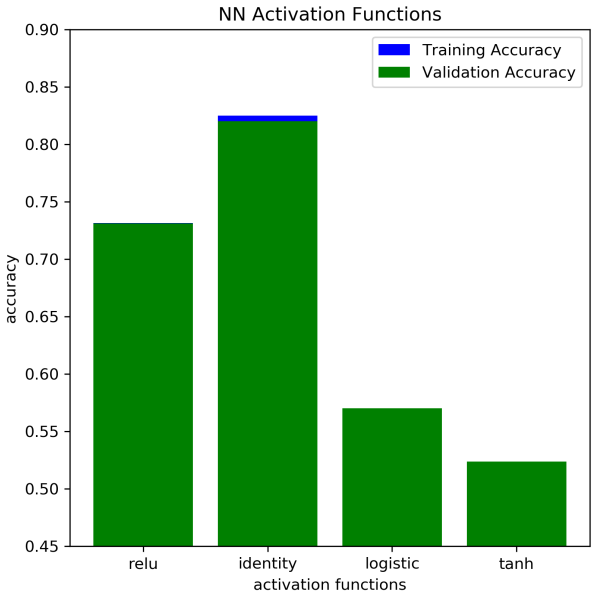


Fig. 18: NN Activation Function Validation

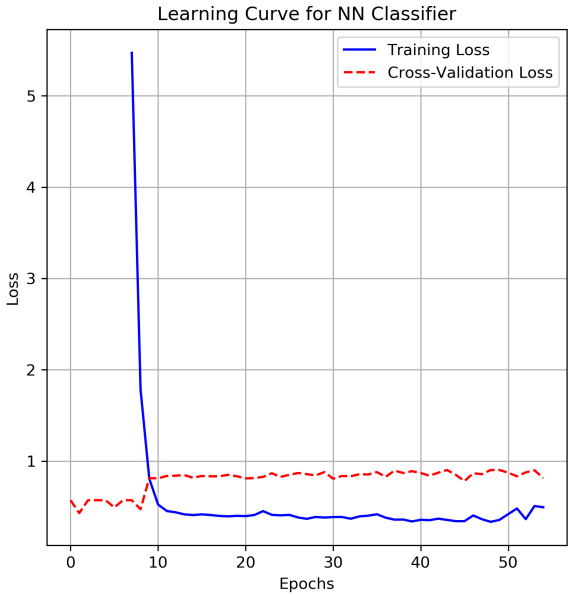


Fig. 19: Tuned NN Learning Curve

VII. References

[1] "Rice (Cammeo and Osmancik)," UCI Machine Learning Repository, 2019. [Online]. Available: https://doi.org/10.24432/C5MW4Z.

[2] "Estimation of Obesity Levels Based On Eating Habits and Physical Condition ," UCI Machine Learning Repository, 2019. [Online]. Available: https://doi.org/10.24432/C5H31Z.

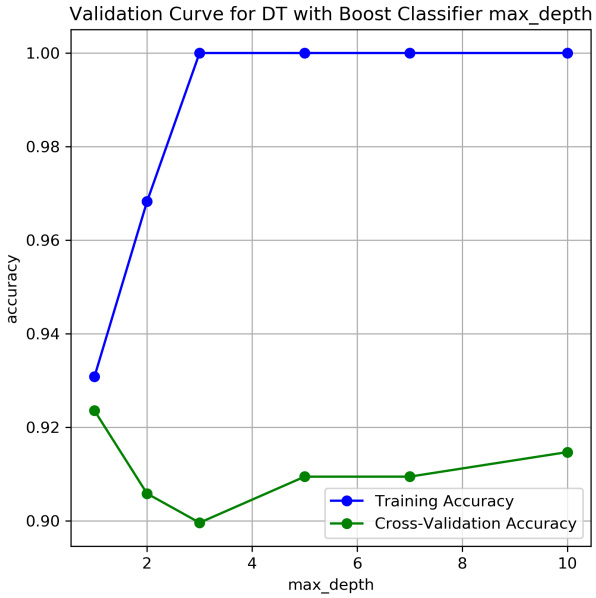


Fig. 20: DT Max Depth Validation Curve

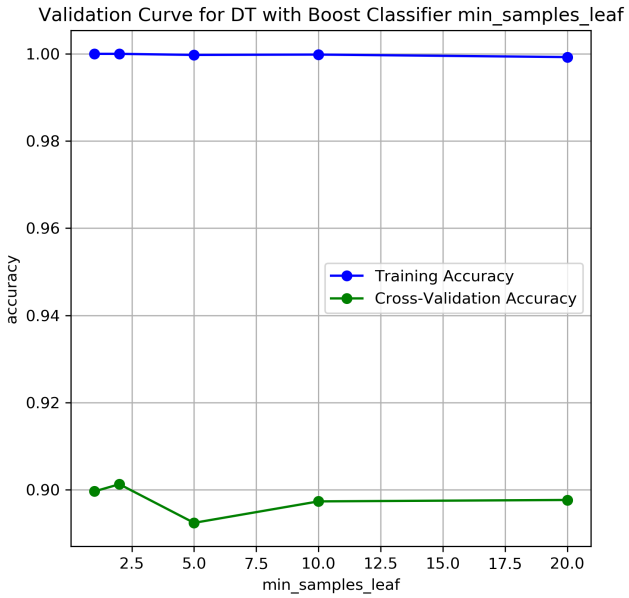


Fig. 21: DT Min Samples-Leaf Validation Curve

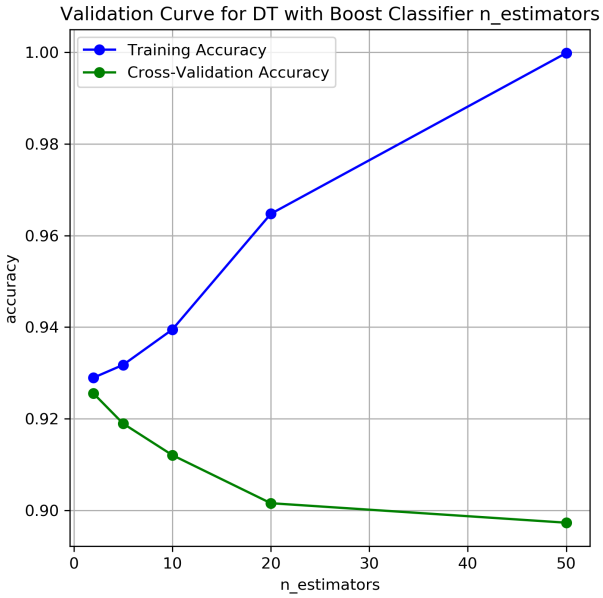


Fig. 22: Boost N-Estimators Validation Curve

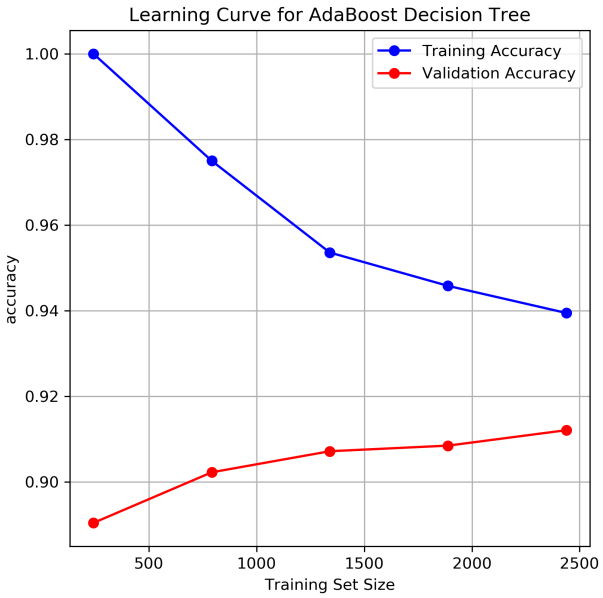


Fig. 23: Tuned AdaBoost w/ Decision Trees Learning Curve

[3] *MLPClassifier* Scikit-learn. https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html.

[4] *SVC* Scikit-learn. https://scikit-learn.org/stable/auto_examples/svm/plot_svm_kernels.html.

[5] Tom M. Mitchell, "Artificial Neural Networks" in *Machine Learning*, McGraw-Hill, 1997, ch. 4, pp. 81-126

[6] Tom M. Mitchell, "Instance-Based Learning" in *Machine Learning*, McGraw-Hill, 1997, ch. 8, pp. 230-247

[7] Tom M. Mitchell, "Decision Tree Learning" in *Machine Learning*, McGraw-Hill, 1997, ch. 3, pp. 52-78

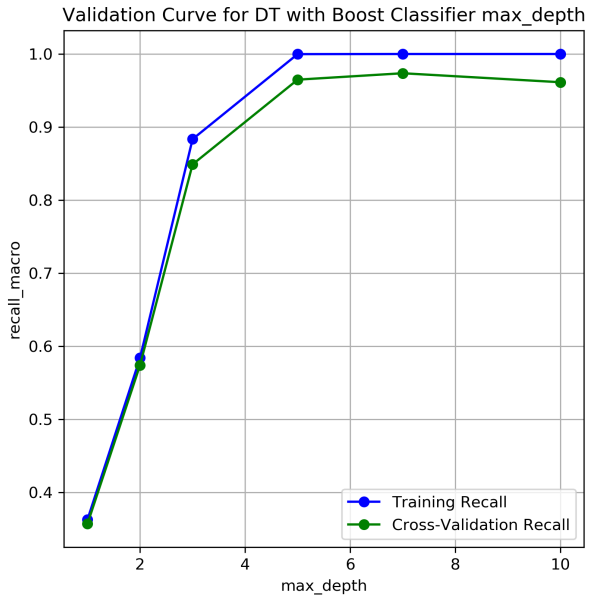


Fig. 24: DT Max Depth Validation Curve

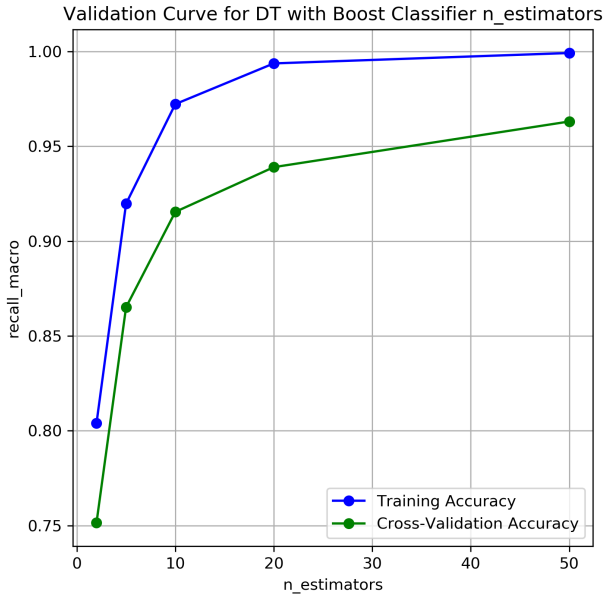


Fig. 26: Boost N-Estimators Validation Curve

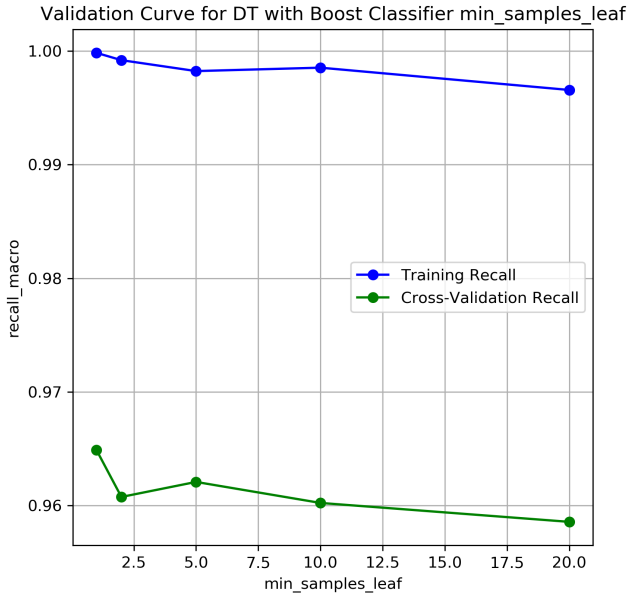


Fig. 25: DT Min Samples-Leaf Validation Curve

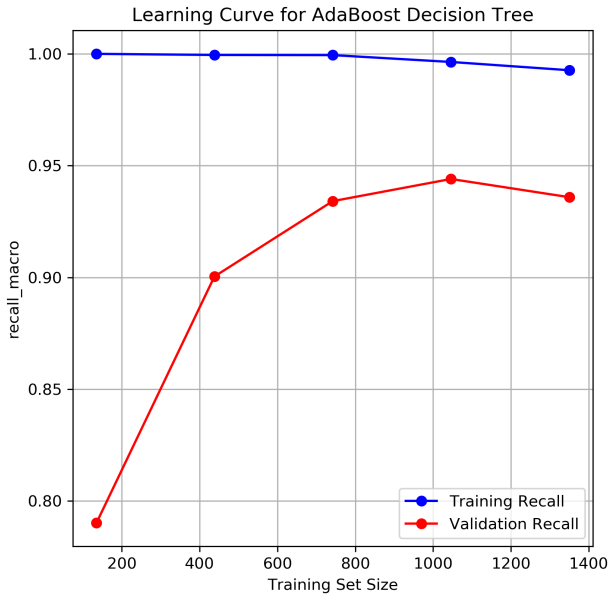


Fig. 27: Tuned AdaBoost w/ Decision Trees Learning Curve

TABLE III: Test Performances on Rice Dataset

NN	SVM	KNN	DT-Boost
0.87	0.94	0.91	0.92

TABLE IV: Test Performances on Obese Dataset

NN	SVM	KNN	DT-Boost
0.72	0.95	0.76	0.95