# CSC730: Report for Assignment 4

South Dakota School of Mines and Technology

Kris Jensen

February 10, 2024

## 1 Introduction

The fourth assignment of this course is meant to introduce semi-supervised classification methods. Semi-supervised learning is a type of machine learning that uses a small amount of labeled data and a large amount of unlabeled data to train models. This is in contrast to supervised learning, which only uses labeled data, and unsupervised learning, which only uses unlabeled data. Semi-supervised learning is particularly useful when labeled data is scarce, as is often the case in practice [1].

The assignment is divided into three parts. The first part of the assignment is to select a supervised learning method then train the model on the full data set. The second part is to train the model on a small subset of the data, and the third part is to train the model on a small subset of data then use a wrapper method to implement semi-supervised learning.

## 2 Description

The assignment was provided by Dr. Loveland as a Jupyter notebook. This notebook contained the code for generating the initial dataset. Empty cells were available for the three parts of the implementation. In the first part we were to select a supervised learning method and train the model on the full dataset. I began this assignment by selecting the linear support vector classifier (SVC) as the supervised learning method. This method seemed like a good choice at the beginning. However, I ran into issue with the class-membership probability prediction. After some research, I was unable to determine the root cause of the issue. This led me to select a different supervised learning method.

At this point, I reviewed several classifiers available in scikit-learn [2]. I selected the random forest classifier as the supervised learning method. This method seemed like a good choice because it is an ensemble method that uses multiple decision trees to make predictions. Also, this method has been studied by others for use in co-training [3]. I was able to train the model on the full dataset, the two-point data set, and create a wrapper method for semi-supervised learning. The results of the predictions are shown in the results section.

If we inspect figure 1 we can see the entire dataset of twenty points. There are three classes of data points in this figure, these are called Class 1, Class 2, and unlabeled.

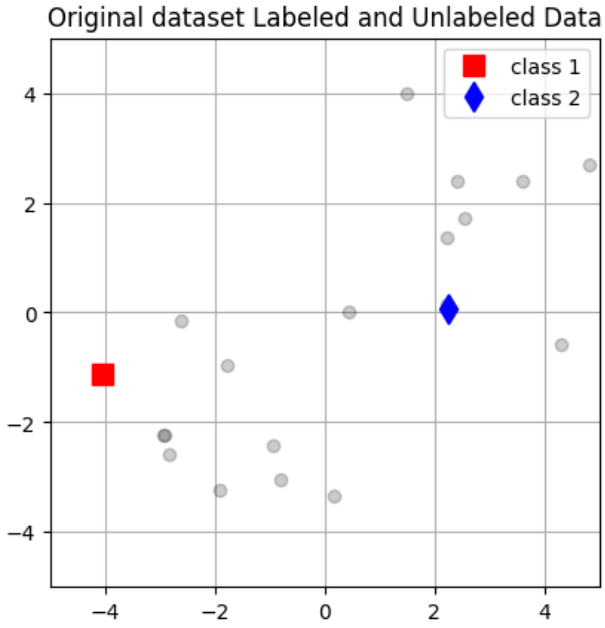


**Figure 1:** Initial Data Set

## 3 Methodology

The third part of this assignment will be to classify the dataset using a wrapper method starting with the two-point dataset.

The wrapper method will be implemented using the random forest classifier. The random forest classifier is an ensemble method that uses multiple decision trees to make predictions. This method has been studied by others for use in semi-supervised learning [3].

**Pseudocode for Wrapper Method for Semi-Supervised Learning**

1. Generate classifier
2. Create a list for psuedo labeling
3. Add first two data points and labels to psuedo set
4. Start with a high probability threshold, 0.9
5. Call fit() using labeled data: $X\_L$, $y\_L$
6. Predict membership of each point in the $X\_U$ dataset against the initial fit
7. If datapoint probability exceeds threshold
   (a) Add pseudolabel and datapoint to tracked points
   (b) Delete point from master list of points
8. Run fit() on new dataset
9. Adjust probability threshold for next iteration
10. plot data for this iteration
11. check stopping condition
12. If stopping condition is not met, go to step 5

The per-iteration results of this method are shown in figure 6 through figure 13.

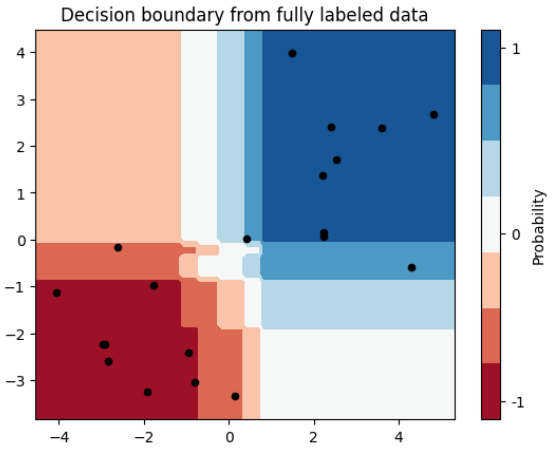


**Figure 2:** Random Forest Model Fit on the Whole Dataset

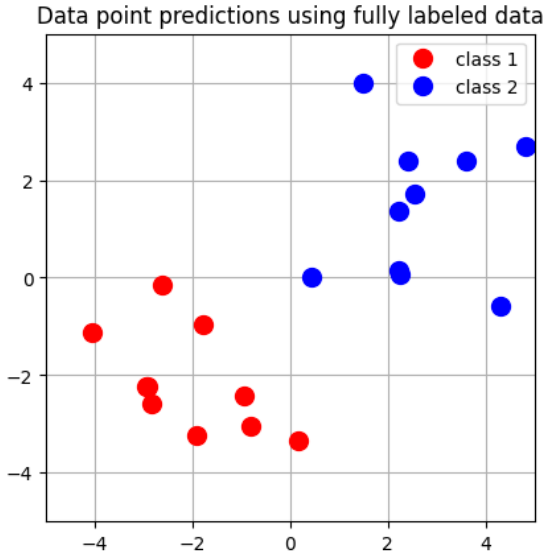


**Figure 3:** Random Forest Model Predictions

# 4 Results

In the first section of the assignment we were asked to run our supervised classifier on the entire dataset. In figure 2 we can inspect the result of running the random forest model fit on the whole dataset. A colorbar on the right side of the figure shows the class-membership probability of the data points. The colorbar ranges from -1 to 1 where -1 is class 1 and 1 is class 2. The colorbar is centered at 0, which is the unlabeled class. The colorbar is a good visual representation of the class-membership probability of the data points. The predictions of the model are shown in figure 3. The predictions are colored by class membership.

In the upcoming figures, figure 4 and figure 5, we can see the results of running the random forest model fit on the two-point dataset. The colorbar on the right side of the figure shows the class-membership probability of the data points. The colorbar ranges from -1 to 1 where -1 is class 1 and 1 is class 2. The colorbar is centered at 0, which is the unlabeled class.

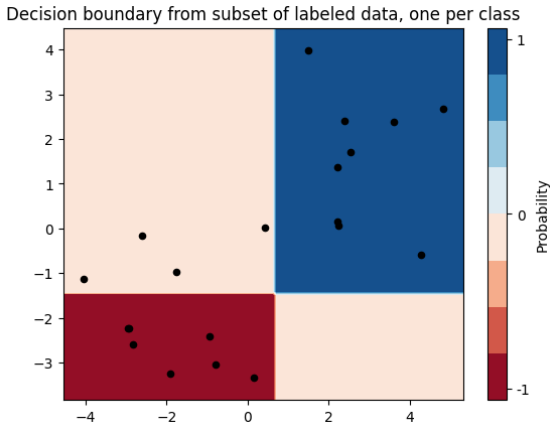


**Figure 4:** Random Forest Model Fit on the Two-Point Dataset

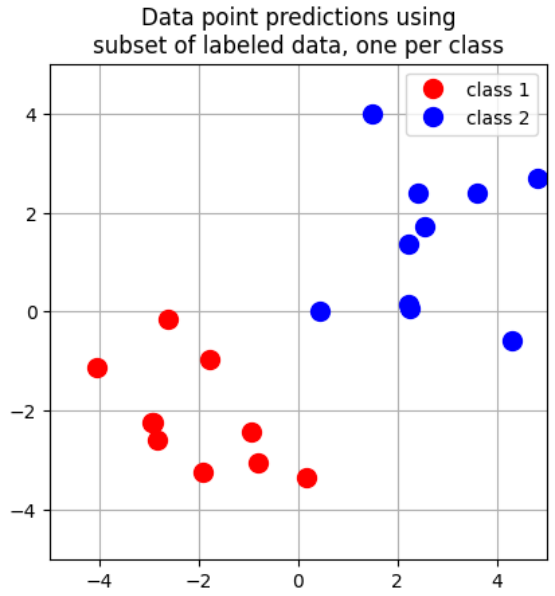


**Figure 5:** Random Forest Model Predictions

The graphs of the predictions shown in figure 3 and figure 5 are similar. The predictions are shown as the background color of the plot.

As described in the methodology section, the wrapper method for semi-supervised learning was implemented using the random forest classifier. The per-iteration results of this method are shown in figure 6 through figure 13.

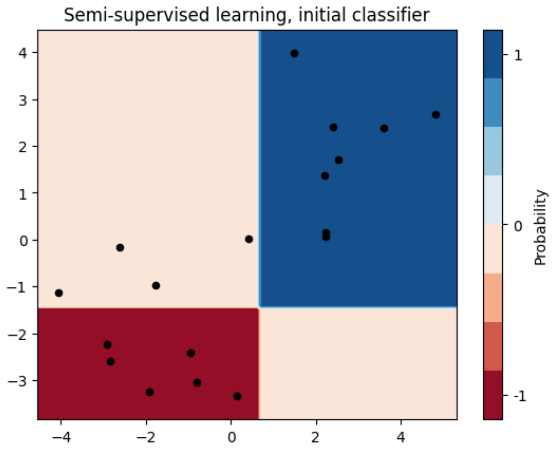


**Figure 6:** Wrapper Method Iteration 1

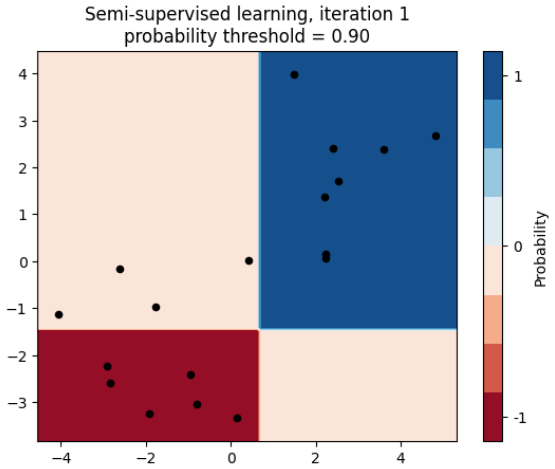


**Figure 7:** Wrapper Method Iteration 2

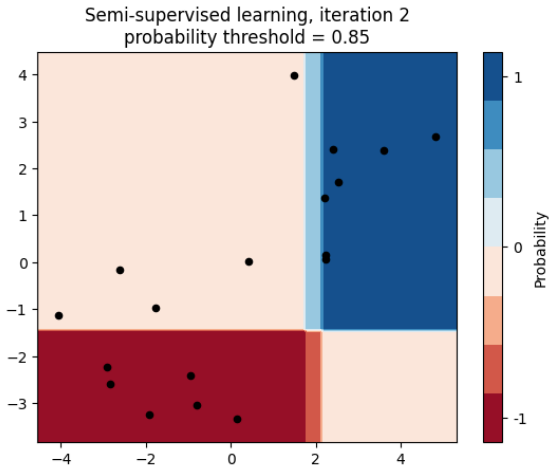


**Figure 8:** Wrapper Method Iteration 3

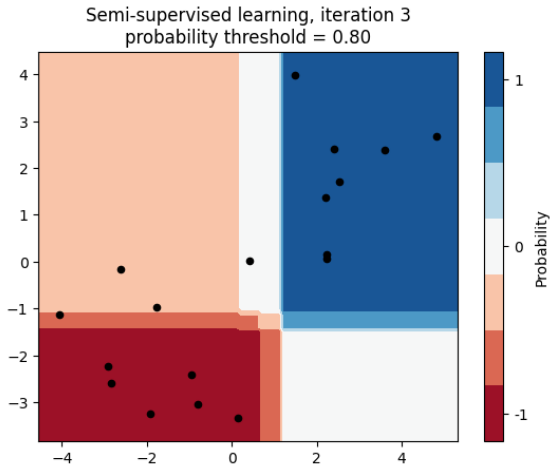


**Figure 9:** Wrapper Method Iteration 4

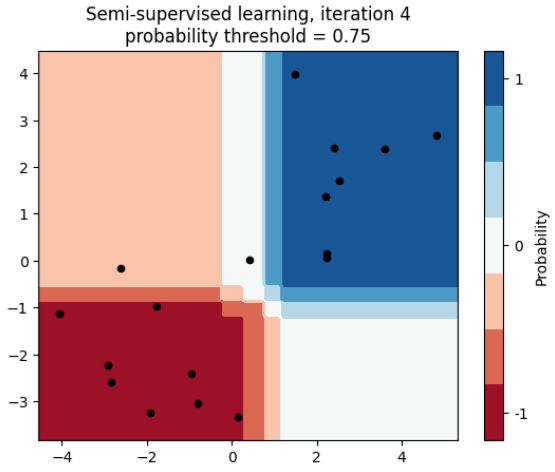


**Figure 10:** Wrapper Method Iteration 5

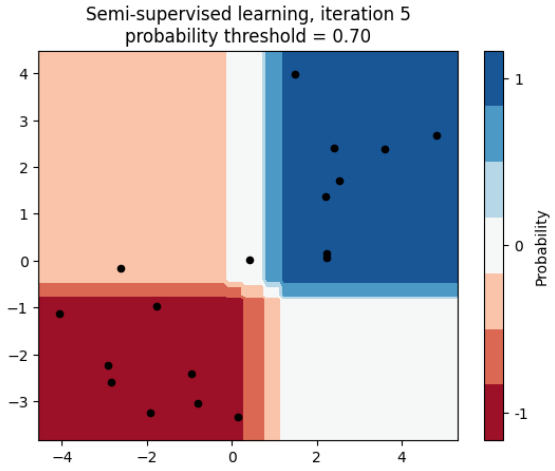


**Figure 11:** Wrapper Method Iteration 6

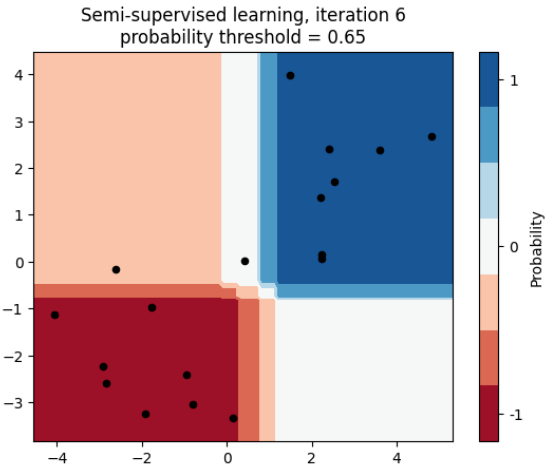


**Figure 12:** Wrapper Method Iteration 7

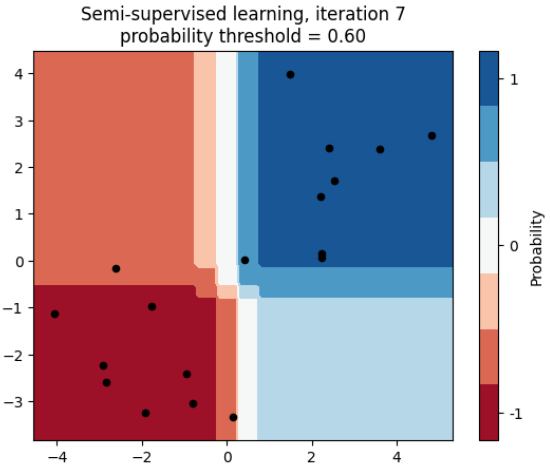


**Figure 13:** Wrapper Method Iteration 8

Reviewing the iteration progress for the wrapper method provides an opportunity to inspect the results of the semi-supervised learning method. These graphs also have an attached colorbar on the right side of the figure. The colorbar ranges from -1 to 1 where -1 is class 1 and 1 is class 2. The colorbar is centered at 0, which is the unlabeled class.

This results of this method provided good results when the labeled data was near other members of the same class. However, the results were not good when the labeled data point was far away from its class members. I am unsure if this is a limitation of this method or if there is a bug in the implementation.

Finally, the assignment requests that all methods are plotted in the same figure. This is shown in figure 14.

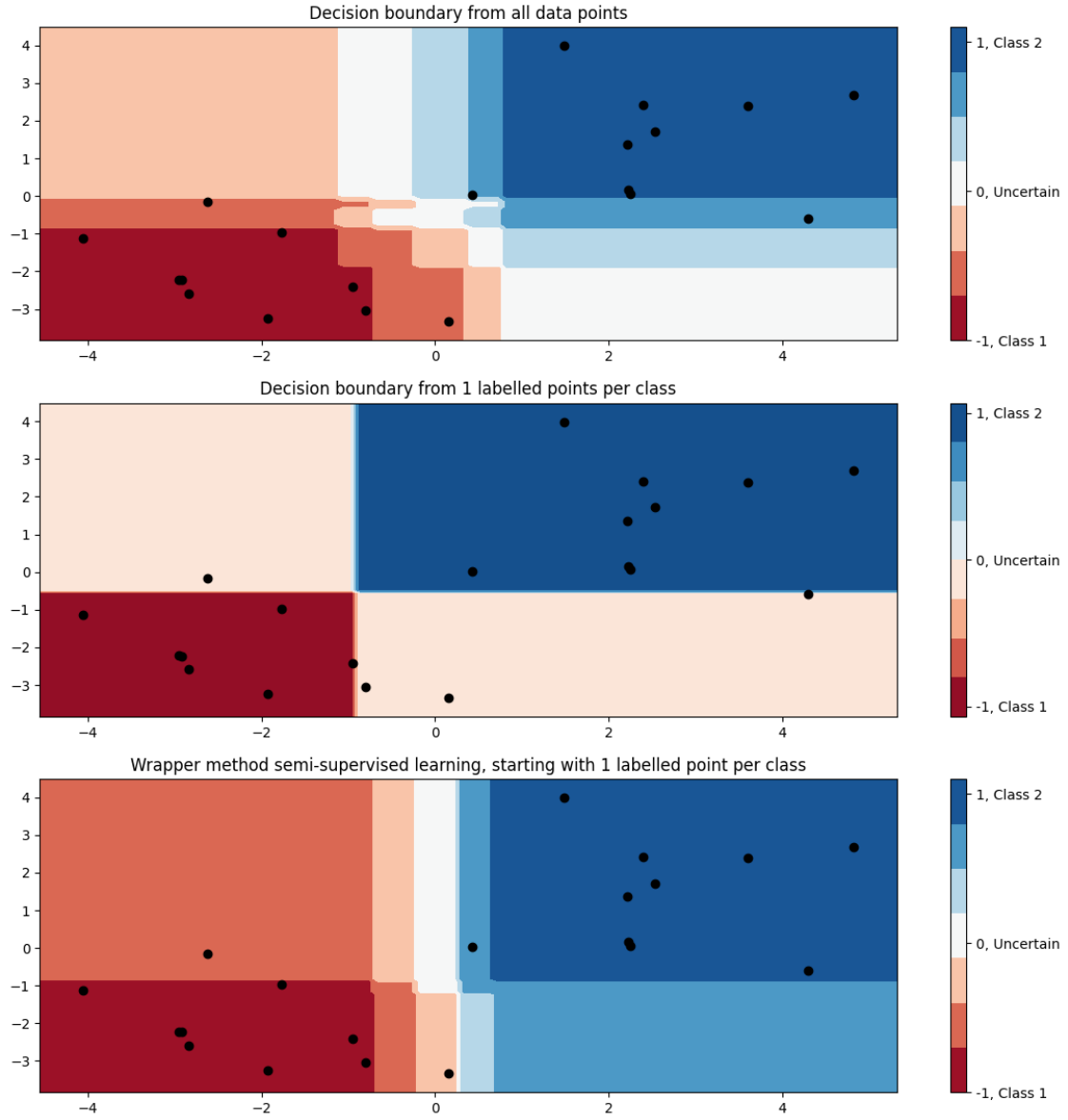


**Figure 14:** All Methods Plotted Together

# 5 Conclusion

Assignment 4 provided an excellent opportunity to explore semi-supervised learning. In this assignment I was able to learn about the wrapper method and see how the random forest classifier from scikit-learn works. I was also able to find a paper that discusses co-training with random forest classifiers.

# 6 References

[1] van Engelen, J.E., Hoos, H.H. A survey on semi-supervised learning. Mach Learn 109, 373–440 (2020). https://doi.org/10.1007/s10994-019-05855-6

[2] *https : //scikit−learn.org/stable/auto_examples/classification/plot_classifier_comparison.html#sphx − glr − auto − examples − classification − plot − classifier − comparison − py*

[3] Wang, W., Zhou, ZH. (2007). Analyzing Co-training Style Algorithms. In: Kok, J.N., Koronacki, J., Mantaras, R.L.d., Matwin, S., Mladenič, D., Skowron, A. (eds) Machine Learning: ECML 2007. ECML 2007. Lecture Notes in Computer Science(), vol 4701. Springer, Berlin, Heidelberg. *https : //doi.org*/10.1007/978−3−540−74958−5_42