**Discussion 8: Regression**

1. What is the difference between supervised and unsupervised machine learning?

Supervised machine learning works with labeled datasets in which we try to classify or predict outcomes with an algorithm with the best accuracy. Unsupervised machine learning works with unlabeled datasets in which we try to group the data together based on their similarities.

1. What is the purpose of a validation set?

The validation set is used to determine which of the different algortithms to be chosen to be used on the test data that were created previously used training data. Once we have our chosen algorithm we see how it does on unseen test data to get a sense of how much bias it has.

1. How many different classes does a logistic regression model predict? Can you give an example of a logistic regression model prediction?

A logistic regression model takes only two possible classes for a possible prediction. For example we can predict if a machine works or fails using a logistic regression model.

1. What is the difference between a Random Forest model and a Classification and Regression Tree (CART) model?

A Random Forest model is essentially a modified CART model in that it uses tree logic but creates many different trees by taking random samples of the data and using random features (independent variables) rather than necessarily using all of the features like CART. Each time it takes a sample and feature selection, it creates a new tree. With multiple trees, it creates a “forest” of modified CART models in order to see which features/variables are the most important in predicting the accuracy of the dependent variable.

1. What is the difference between a k-Nearest Neighbor (KNN) model and a Support Vector Machine (SVM) model? Does a KNN model represent supervised or unsupervised machine learning model?

A KNN is an unsupervised clustering machine learning model that takes longer to run (more computationally expensive) and harder to interpret than SVM but works with more possible sets of patterns than SVM.

1. In what way is a Perceptron model similar to a logistic regression model? In what way is a Perceptron model similar to a Neural Network model?

Perceptron models are similar to logistic regression models in that they only predict two classes. However, Perceptron models are a subset of Neural Networks that only work with the binary output for classification and work in the same way as them by modifying the inputted weights for the neurons using an activation function.

1. What inspired Neural Networks to be created (what does a Neural Network mimic with regards to humans)? Is the output of a Neural Network more or less interpretable than a regression model?

Neural Networks were designed to mimic the way the human brain learns.

1. Why do we scale our data? Does this help or hinder the interpretability of our model?

We scale data because many machine learning algorithms use the distance between two data points as part of their computations. However, since not all data goes by the same scale (say inches for one measurement and millimeters for another), the features with higher magnitudes of difference will weigh a lot more for the calculations in the model. Scaling makes sure that all variables have a fair chance to be weighted equally. However, this can make the final interpretations of the model more difficult as some features won’t have the same unit of measurement that they were inputted with.

1. Why use a confusion matrix? What information does it give an analyst that training, validation and testing accuracy doesn’t?

A confusion matrix can show how well a model’s classification works for all the different classes. Using solely training and testing accuracy can lead to faulty conclusion, take for example this scenario. We are trying to identify active and inactive volcanoes with a model. Say we know that 90% of volcanoes are inactive and there are 10 volcanoes. If we guess that a volcano is inactive and it proves to be correct, we win $50 but if we guess wrong we lose $1000 and everything else yields no return. We create two models:

Model A – always guess inactive (correct 90% of the time)

Model B – mixed guesses (correct 50% of the time)

Confusion Matrix Model A:

|  |  |  |
| --- | --- | --- |
|  | Guess Active | Guess Inactive |
| Actual Active | 0 | 1 |
| Actual Inactive | 0 | 9 |

Confusion Matrix Model B:

|  |  |  |
| --- | --- | --- |
|  | Guess Active | Guess Inactive |
| Actual Active | 1 | 0 |
| Actual Inactive | 5 | 4 |

Model A has a higher accuracy (90% much higher than 40%) but we actually lose money (9 \* $ 50 – $ 1000 = - $ 550) whereas Model B makes money (4 \* $ 50 = $ 200). The confusion matrix shows us that Model B, while less accurate still provides us a better return.