Math 218: Project Plan of Analysis

Haydoja

Payoja Adhikari and Hayden Hunskor

November 21, 2022

Data description

We picked the Ames Housing dataset compiled by Dean De Cock, which is publicly available on Kaggle. The dataset includes extensive information on the sale of residential properties in Ames, Iowa from 2006 to 2010. It contains 1460 observations with 79 variables on various house specifications. The variables are a mix of continuous, categorical and discrete types.

The Kaggle zip file included both a train and a test set. However, only the train set included the sale price of the houses. Since we want to analyze the performance of our model by finding the the errror rates (for which it is necessary to have information on the true sale price), we decided to treat the train set as our entire data.

Project Plan of Analysis

Research Question 1: How do housing prices differ in Ames, Iowa based on the different housing features? Although we have a total of 80 housing features, that might be a lot of features to take into consideration to estimate the sale price in reality. Thus, to answer this research question, we decided to implement the lasso method so that coefficients of features that are not as important shrinks to 0. To implement lasso method, we will divide our data into two sets 'train' and 'test'.Then, using all available housing features we will fit the lasso model on the training set, with λ chosen by cross-validation with k-fold equals 10. Our dependent variable will be the sale price of house and independent variables will be the remaining 78 housing features available in the dataset. To validate the performance of our model, we plan to predict the sale price of houses in the test set using the lasso model fit on the training set. Then, we will examine the test error rate.

Packages required: glmnet

We might end up performing ridge regression as well just to check which of the two models performs better. The plan of analysis will be the eaxct same as the one described above for the Lasso model. The only difference will be that coefficients of features that are not as important will shrinks **towards** 0 (unlike the Lasso model where the coefficient of some features will shrink **to** 0.)

Research Question 2: Which neighborhood do the houses belong to based on the the different housing features? To answer this classification question, we plan to use two different methods.

Method 1: Tree-Based Classification To implement this method, we will divide our data into two sets 'train' and 'test'. Then, we will fit a decision classification tree to the training data, using 'neighborhood' as the response variable. For predictors, we will use all other variables. To validate the performance of our model, we will predict the response on the test data and produce a contingency table comparing the true test labels to the predictions.

Packages required: tree

Method 2: XGBoost XGBoost is one of the most powerful model that uses a decision tree-based methodology. XGBoost is an ensemble method in which the algorithm looks at a decision tree fitted for the data, figures out what the model got wrong and then focuses on improving the performance on the stuff that the model got wrong. The algorithm repeats this process iteratively until it finds a model that performs really well. Some sources that we looked at to understand XGBoost Classification are:

- https://www.kaggle.com/code/rtatman/machine-learning-with-xgboost-in-r/notebook
- https://towardsdatascience.com/beginners-guide-to-xgboost-for-classification-problems-50f75aac5390
- https://www.youtube.com/watch?v=8b1JEDvenQU&list=PLblh5JKOoLULU0irPgs1SnKO6wqVjKUsQ&index=2&t=270s&ab_channel=StatQuestwithJoshStarmer
- https://www.youtube.com/watch?v=0ikyjpaUDFQ&ab_channel=CodingTech

R has a builin package for XGBoost called 'xgboost'. We will be using this package for our analysis. To begin with, we will split out data into test and train set and use the train set to fit our model. For now we are thinking that our response variable will be neighborhood and our predictor will be sale price and year sold.

Data Cleaning

The 'xgboost' package only works with numeric data. Thus, we need to convert our categorical value into numeric ones. For example: we can encode the neighborhood 1,2,. . . 25, to represent the 25 neighborhood in Ames, Iowa. Then, we will convert our dataframe with just numeric values into a matrix. This step is necessary because the package only takes matrix as an input.

Training the model

The 'xgboost' function in 'xgboost' package requires three key informations in order to fit the model.

- data to be used
- number of iterations that we want the model to perform
- objective function: Here, we need to pass 'multi:softmax'. This basically tells the algorithm that the our response variable has multiple classes.

Cross-validation

We will use the model fitted on the train data to predict the class for the test data. This can be done using the 'predict' function. finally, we will use find the misclassification rate.