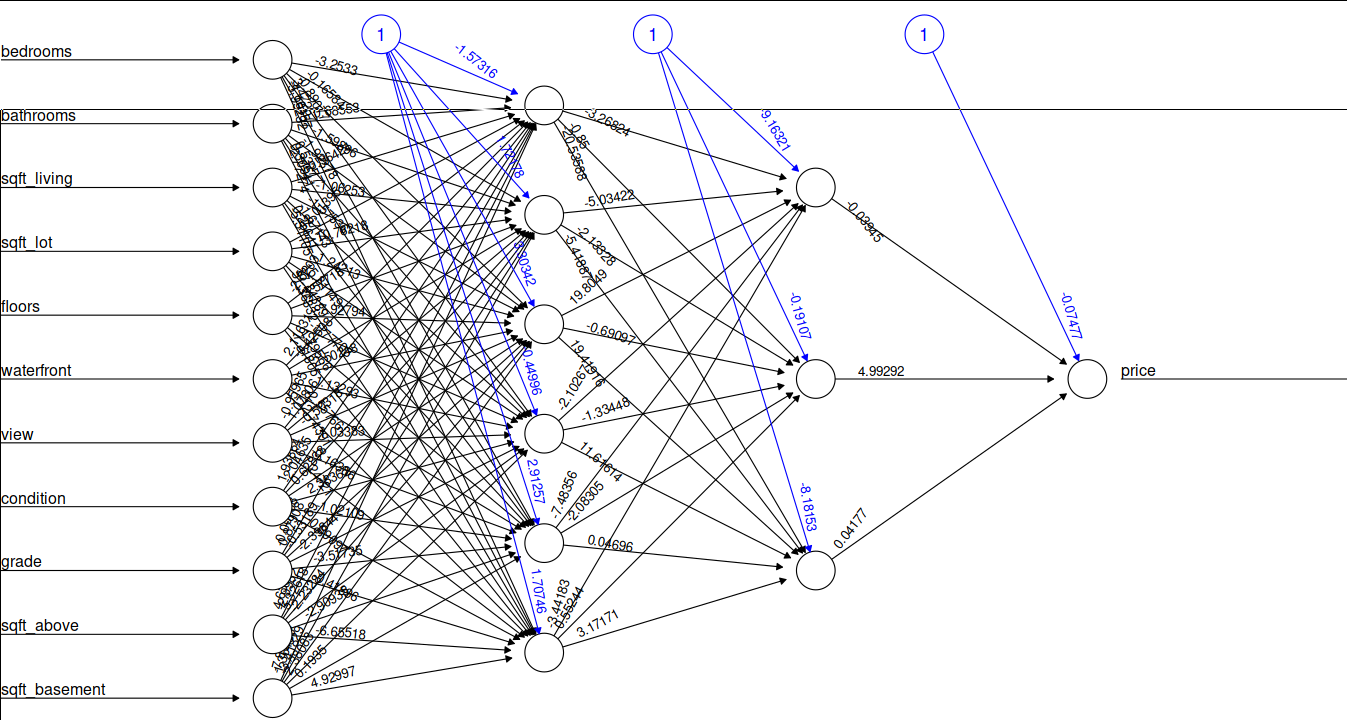
Doug Woodward

CS613 HW 12

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Neural Net | Training MSE | Training Dataset Correlation between observed values and predicted values of the response variable | Testing MSE | Testing Dataset Correlation between observed values and predicted values of the response variable |
| 11-1-1 | 43462077258 | 0.8246159201 | 44800255270 | 0.8034745212 |
| 11-2-1-1 | 41867594130 | 0.8372213291 | 42732933432 | 0.8072778614 |
| 11-3-2-1 | 40200063747 | 0.8443048083 | 43291808562 | 0.8045043929 |
| 11-6-3-1 | 37514895241 | 0.8555879434 | 41034450323 | 0.8179172439 |



library(neuralnet)  
data = read.csv("kc\_house\_data.csv")  
  
  
# pick out elements we want ot use  
df = data[,c("price","bedrooms","bathrooms", "sqft\_living","sqft\_lot", "floors","waterfront","view","condition","grade","sqft\_above","sqft\_basement")]  
# normalize  
maxValue = apply(df, 2, max)  
minValue = apply(df, 2, min)  
scaled <- as.data.frame(scale(df, center = minValue, scale=maxValue-minValue))  
# test train split  
train.size <- floor(0.7 \* nrow(scaled))  
train.index <- sample(sample(seq\_len(nrow(scaled)), size = train.size))  
train <- scaled[train.index,]  
test <- scaled[-train.index,]  
  
n <- names(train)  
f <- as.formula(paste("price ~", paste(n[!n %in% "price"], collapse = " + ")))  
f   
nn <- neuralnet(f, data=train, hidden=c(6,3), linear.output=T)  
  
predictions.train = compute(nn,train[,2:12])  
  
predictions.train <- predictions.train$net.result \* +(max(df$price)-min(df$price)) + min(df$price)  
actualValues.train <- (train$price)\*(max(df$price) - min(df$price)) + min(df$price)  
"Train MSE"  
(MSE.train = sum((predictions.train - actualValues.train)^2)/nrow(train))  
  
predictions.test = compute(nn,test[,2:12])  
  
predictions.test <- predictions.test$net.result \* +(max(df$price)-min(df$price)) + min(df$price)  
actualValues.test <- (test$price)\*(max(df$price) - min(df$price)) + min(df$price)  
"Test MSE"  
(MSE.test = sum((predictions.test - actualValues.test)^2)/nrow(test))  
  
  
plot(nn)  
"Train Correlation"  
cor(predictions.train, train$price)  
"Test Correlation"  
cor(predictions.test, test$price)