This project was an interesting challenge for me, since I hadn’t used a neural network model before, and have primarily worked in R rather than Python.

**Fitting the neural network models**

After importing and inspecting the data in Python, I found that the majority of the predictor variables were categorical. I considered converting them to numeric and including them, but many of these variables didn’t have values that could be ordered easily. For example, the “Fence” variable has four different values, but it’s not clear which values might be considered more expensive. I also considered using dummy variables to code these categorical predictors and including them in the neural network. However, this would have added many more predictors, and I felt that the increased processing demand and complexity of the model weren’t worth the tradeoff. Thus, I included only numeric predictors in the neural network.

I fit three neural network models, with increasing complexity. The baseline model had a single hidden layer with as many neurons as predictors (37). The Two Layer model had an additional hidden layer with about half as many neurons (19). The Three Layer model added yet another hidden layer with 10 neurons. Using ten-fold cross validation and the root mean square error, I found that the Two Layer model performed best.

**Interpreting the results**

This result was actually a little surprising to me, because I assumed that the simplest model would perform best. I figured that normal house buyers and sellers use fairly simple mental formulas to estimate house value. These mental formulas might be thought of as a version of linear regression: the value of a house is increased by some constant for each feature, without including interactions or higher-order features of the data. I didn’t think that these buyers and sellers would rely on anything similar to the complexity of a two layer neural network model.

In addition the neural network, I did a little more exploratory analysis in R. I used each predictor separately to predict the housing price in a simple linear regression, and checked the resulting R squared values to see which variables were most predictive. I found that some variables had high R squared values (>.5), which may provide support for my initial theory that home buyers use simple mental formulas. I also noted that many of the most predictive house features were categorical (such as neighborhood), suggesting that my neural network models, which included only numerical variables, were missing important data.

**Future directions**

This was obviously a first attempt at the problem, and there are a number of next steps I would take to improve the accuracy of my model. The obvious choice would be to dummy code the categorical variables and add them to the neural network models. I could also do a lot more tuning of the neural networks to see which structure worked best. The other important step would be to conduct a stepwise linear regression and see if it out-performed the neural networks model.

post script- I just realized that my Python script includes the ID variable as an input to the neural networks! This shouldn’t compromise the results, but it’s obviously something I’d remove in the future.