Report for Assignment

Hassan Sherwani

I am writing this report not as step by step coding tutorial. I have provided description of code, relevant concepts and theory already in Jupyter Notebook. I am providing summary of used model, methods to build such models and results that they produce.

1- Methods

I have used three main notebooks that contain coding from three data science domains i.e econometrics, machine learning and deep learning. It is worth mentioning that statistician love to work with econometrics, Data scientists support machine learning approach and computer scientists prefer deep learning. My Idea is to apply three different approaches to get most optimal predictor. As for evaluation matrices, I earlier used R^2 and RMSE, MSE, MAE. Eventually, I only focused on loss function i.e RMSE to evaluate overall model performance.

Before applying any model, I did exploratory data analysis and dataset inspection. I found no missing values, no outliers, no duplicate values, no multi-collinear suspects, no correlation issues. After taken care of all data pre-processing, I moved forward with model building.

2- Outcome of each method

a) First, I used linear regression method (OLS). I fit the model, predicted and then tested my model. I used train-test split approach for model testing. I found R^2 value of 0.0161 and RMSE value of 0.9047. I was not done with results so, I used backward elimination technique. I checked null hypothesis using p-values. Those features which were not significant were taken out of model. I again fit, predicted and evaluated model. This time I got R^2 value as 0.9044 and RMSE as 0.0166. It is only decimal improvement. But, it is important that backward elimination actually effected our model.

b) In second notebook, I used couple of machine learning model. I used few linear models such as Ridge and Lasso Regession, non-linear models such as SVM, KNN, Bagging models i.e Random Forest, Extre-Tree Regressor, Boosting models such as XGBRegressor, GradientBoosting, and neural network i.e multi-layer perceptron. In result, I got lowest RMSE score of 0.9066 with XGB Regressor. So, I considered that most optimal model. However, I didn't stop there. I found these result on train-test split technique so, I applied additional GridSearchCV in trying to further explore my model.I fit, predicted best

parameters and found RMSE score of 0.92 which is higher than already applied XGB Regressor. So, for this reason I ll keep my previous XGB model as most optimal model with train-test split approach.

c) Finally, I applied deep learning technique using neural network. I have already applied multi-layer perceptron in previous notebook which is only feed forward neural network. For a back propagation neural network which would provide updated weights and reduced biases. Additionally, I assumed that features(columns) in given dataset are months therefore, I take it as time series. There might be a repeating pattern hence, I used recurrent neural network. One of RNN's type is LSTM that is being used here. I built neural network with less complexity i.e not too many hidden layers, fit model, predicted on test data and finally evaluated results. I found that RMSE is 0.24 which is lowest of all.

Models	RMSE
OLS Regression(with backward elimination applied)	0.9044
Lasso Regression	0.9107
Ridge Regression	0.9055
K-Nearest Neighbor	0.9906
Support Vector	0.9242
Random Forest	0.9538
Extra Tree Regression	0.9537
Gradient Boosting Regression	0.9067
XGB Regression	0.9066
MLP Regression	0.9087
Recurrent NN	0.24

3- Conclusion

By these results, I conclude that RNN model performs best as it did reduce our loss function at lowest point. Another option (if memory conscious) is to use OLS regression model as it provides better R^2 value and second best lowest RMSE score comparing to all other models.

Note:

- In notebook 3, final plot of test and train loss score is as per mse not with rmse. I later calculated rmse separately using given model.
- I have used different algorithms on three different datasets. As I had to keep three datasets as per assignment requirement. For my own satisfaction, I applied LSTM neural network on all three and it performs best for all. Hence, recurrent neural network (using LSTM) is winner in all cases.