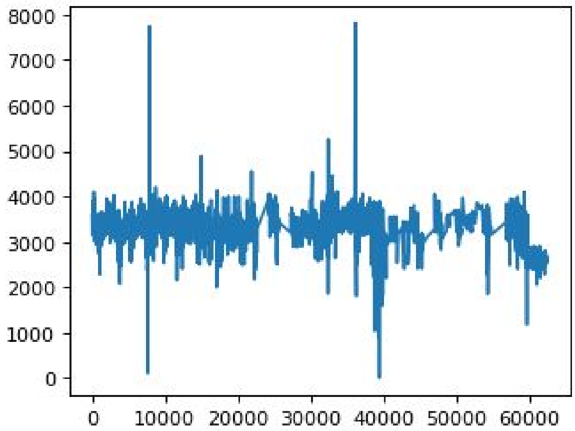
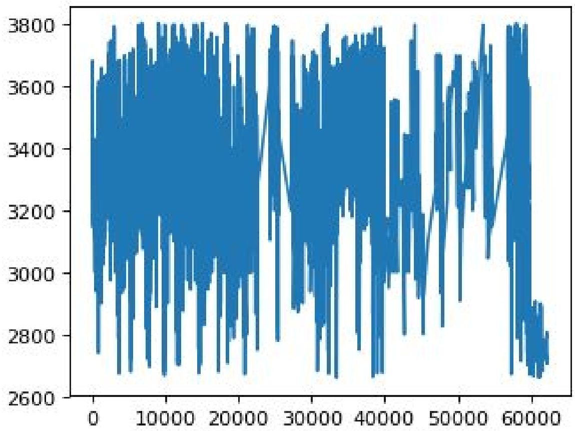
The tasks have been handled in separate Jupyter Notebooks for each major chunk of the pipeline. All in all, we have 4 notebooks.

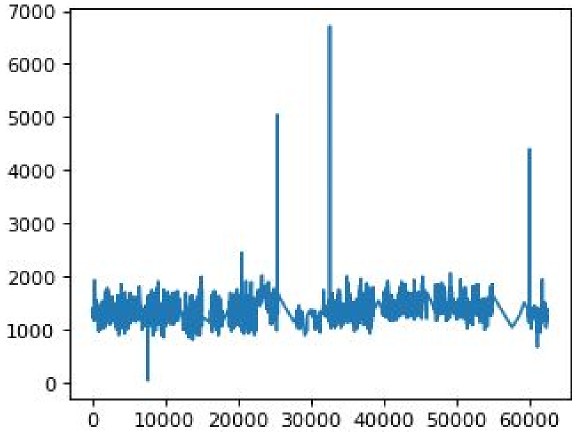
1. DetectAndFIlterOutliersFinal - This notebook scans through the entire dataset. It first finds out the outliers in the modal price attribute of the dataset (or target variable) and then drops the rows that have outliers.
   Assumption - We are considering the values of a commodity that are greater than 95% quantile and lesser than 5% quantile to be statistically considered as outliers. Visualisations have been added as a part of the notebook along with the code to both generate visualisations and detect and remove outliers. Here is an instance of the same.



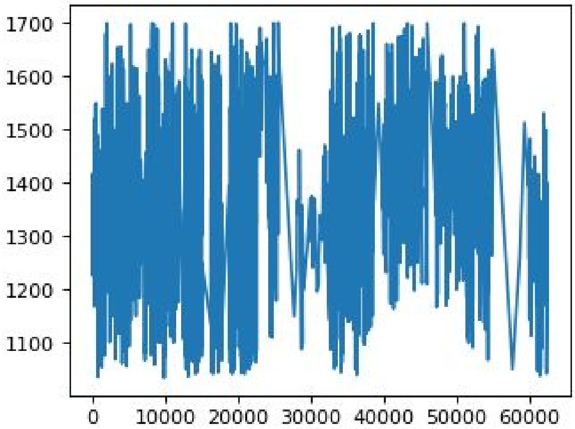
Soybean prices before.



Soybean prices after outlier removal.



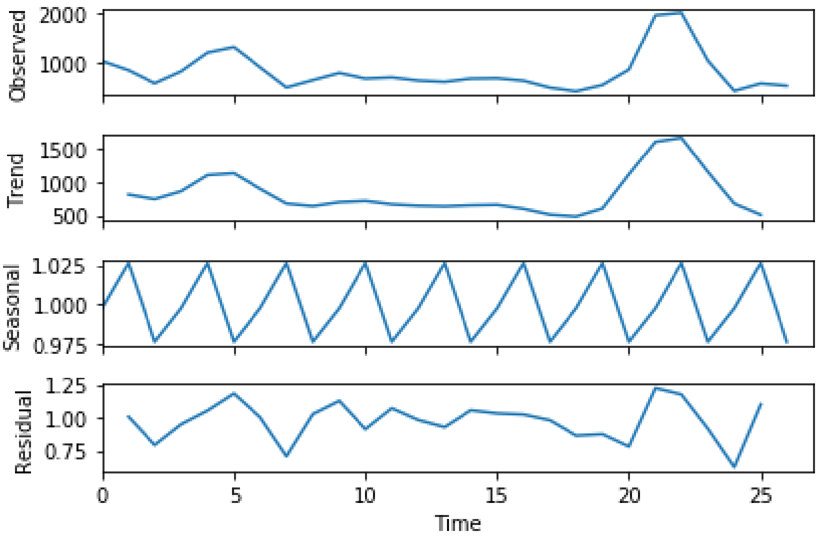
Maize prices before.



Maize prices after outlier removal.

2. The second part of out objective had two sub-parts.

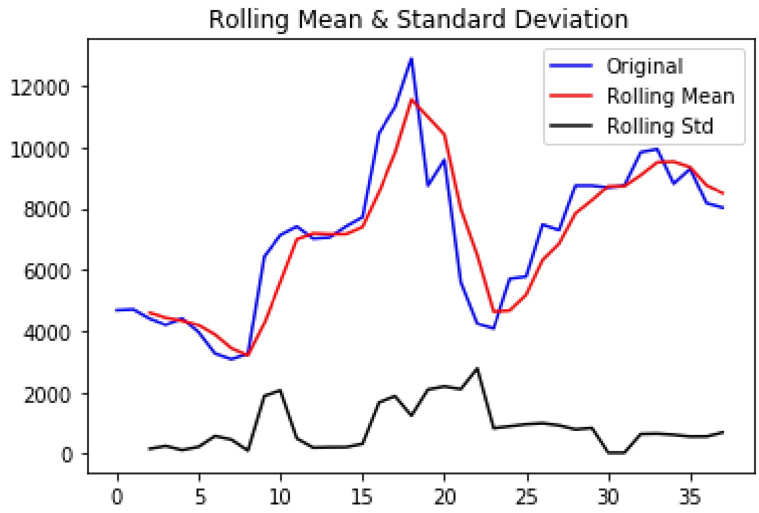
   a. DetectSeasonalityTypeFinal - This notebook scans through each APMC, Commodity pair and figures out what would be the better seasonality decomposition for it between Additive and Multiplicative. We flso figure out the trend and residuals and plot all of them, like so.



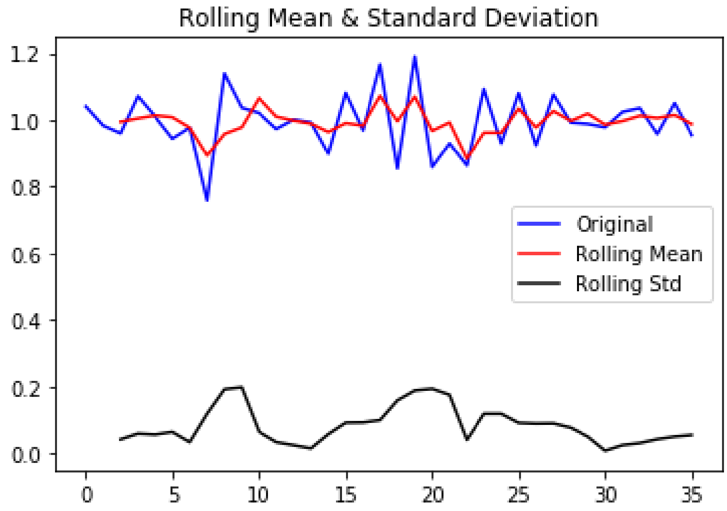
Results of the multiplicative decomposition of modal prices of Cabbage in Mumbai.

Assumption - A ‘better’ decomposition here is defined statistically in terms of the autocorrelation factor of the residuals obtained after the decomposition. A smaller sum of squares of the autocorrelation values of the residuals indicates a better decomposition. A frequency of 3 months has been taken for the time series decomposition. This makes sense as seasons in India as usually 3 months long and trends of most commodities vary according to seasonal changes. Other values were also experimented with. However, 3 gave the best decomposition overall and was therefore chosen. We have modular well built functions in the notebook to run each of the tasks.

B. DeseasonalizingFinal - In this notebook, we want to deseasonalize the modal price of APMC, Commodity pairs in correspondence with the best decomposition obtained for the pair. We build up on the last notebook. However, we have a couple of additional functions this time. One, we use to check whether the data is stationary or not. We use Dickey-Fuller test along with plotting rolling means and standard deviations to both clearly visualise and quantify stationarity. We then deseasonalize our data using the better decomposition type and extract the residuals obtained after making the data stationery. As a sanity test measure, we again check if this newly obtained data is stationary.



The original non-stationary modal price timeseries data of Garlic in Nagpur with time varying rolling mean and rolling standard deviations before Deseasonalizing.



A smoothed out, deseasonalized modal price data of Garlic prices in Nagpur.

Assumption - We still maintain a frequency of 3 for the decomposition like in the previous decomposition. In addition, while calculating rolling mean and rolling standard deviations we keep a window of 3 as well in order to be consistent in our approach.

3. PredictFinal -

The final objective of the analysis was to forecast the modal prices of each APMC Commodity pair for the next 3 months. I had two options here. One, to go with traditional forecasting models like ARIMA or use recurrent neural network variants. I chose the later. This is because of two reasons. One, for a commodity APMC pair, LSTMs would be able to learn non linear functions as well, so we can use naive differencing in order to remove trend and seasonality, because even if it is not totally stationery, LSTMs would take care of it. Difference method makes it easier to get the final output values in the same scale. Two, the data points for a pair were very few and for autoregressive integrated moving average (ARIMA) models, the rule of thumb is that one should have at least 50 but preferably more than 100 observations (Box and Tiao 1975). So, I decided to place my bets on LSTMs to give me better results with fewer data.

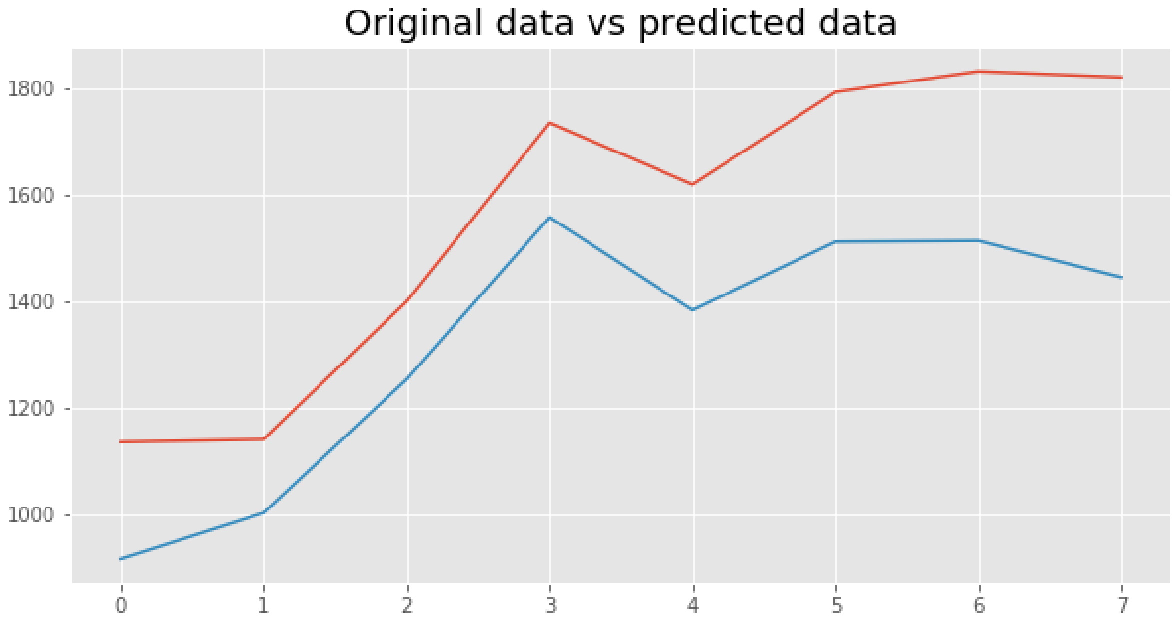
The notebook uses the filtered dataset with outliers removed. It has numerous helper functions that have been well documented in the notebook and these aid in several preprocessing steps, right from taking differences to creating lag features to turn our problem into a supervised learning problem.

There are a lot of things that can be tweaked in here. Based on experimenting with different values, I decided to create lag features based on past 5 months and take differences at an interval of 3 months. (This combination performed better empirically than others.)

We also perform train test splits to evaluate our performance before predicting. We can sometimes see very high RMSE values. This is however, expected and can be improved using the following measures.

1. We are using the same architecture for all pairs. We can tweak the model architecture and also parameters such as lag features and difference interval for each pair individually to improve performance.
2. We can use more epochs to train our LSTM model and also increase its depth.

Finally, we predict for such APMC Commodity pairs that have >10 datapoints. Predicting with <10 datapoints actually does not make much sense. We have too few data in such a case. I have put up 333 results in my output file, but the function if allowed to run will generate outputs for all pairs. I had to preempt it because of lack of time. And it takes quite some time to run, because technically we are tailor fitting a model for each APMC Commodity pair and predicting using that. The output csv file has APMC Commodity names and predictions for the next 3 months in order.



Here’s how our model does on the train test split for Arvi prices in Mumbai by just training for 1 epoch.

The task was quite fun to do and challenging at the same time. I learnt a whole lot of new things in the way. I have all the code well documented and with short sharp functions in the 4 notebooks that I have sent, one for each task. I have stated all my assumptions, approach and methodologies in this report. I have also put in the visualisations that I used to derive insights. Thank you for the opportunity.