EDA Doc

1   Initial

In our initial discussions we decided that we wanted to investigate a model related to time series.

Initially, we considered looking at stock price data and thought it would be interesting to see if we could predict future stock prices based on those we had already seen. After some more investigation into this, we decided against it for several reasons. First and foremost was the lack of variables that is involved in stock price dataset. The large majority of datasets we found only contained information for 4 different variables, the opening price of the stock, the closing price, SAY OTHER TWO VARIABLES HERE. One reason we didn't think this would be good is because it didn't provide much in the way of contextual information about why these prices were what they were, it simply just gave what the prices were at the start and end of a certain time period. Also, we didn't think this would be very appropriate for a project which was centred around massively parallel programming as it simply wouldn't require much processing power, even if we did look at data going back very far. We looked at others attempting to do work involving stock prices and saw that on the whole it hadn't gone down very well, due to the unpredictable and random nature of stocks many models were virtually useless in terms of prediction and we didn't think this would make for a very interesting project.

The next area we considered was the interest rates in the UK. Our main obstacle for this was combining a vast amount of datasets. We first considered what sort of factors might impact the interest rate and looked at datasets for things such as the unemployment rate, inflation, GDP, SAY OTHER FACTORS. We found that a lot of the datasets provided data for different time periods, some were given quarterly, some monthly, and some daily. We didn't see this as a massive problem as we could extrapolate some of these datasets and cut out some of the data from others so that they were all done on the same time frame, ideally monthly. However, we still had the problem of combining these datasets correctly and thought it would require too much effort and nuance just to set up the data before we'd even got into the meat of the project. Furthermore, since the data was only collected monthly, we'd require datasets that date back very far in order to compile training and test sets of an acceptable size to then

conduct our work involving neural networks.

We eventually decided to look at other prediction projects that people had undertaken on Kaggle relating to time series and found an interesting dataset involving air pollution in Beijing. This dataset contained a good amount of relevant features such as wind and weather and had been collected hourly over a five-year period. These characteristics remedied problems we had encountered earlier. Firstly, we had a decent amount of factors that pertained to what we were investigating and would allow us to hopefully go onto make accurate predictions. Additionally, all of the data was in one place, we wouldn't have to spend time tediously and meticulously manipulating and combining datasets to be in the form that we wanted and could crack on with the work we were more interested in. The large amount of data also meant that we had a good amount of data on which to train our model and see how it would deal with very large datasets which it could be generalised to handle. Furthermore, there were plenty of resources we could use for inspiration that involved the massively parallel computing techniques that we were interested in that we could draw inspiration from and that could advise us on our journey.

Now we had to consider how to do the training and test splits for this data that was appropriate for our model. After some research, we decided that either Rolling Window or Enlarging Window would be appropriate. We look at both of these in our work, but since the goal is to make something that could be generalised to much larger datasets for massively parallel modelling we believe that the Rolling Window is the more appropriate of the two.