Project IV Proposal

Kenneth Wirjadisastra

June 17, 2025

1 Statistical Question

Time series analysis involves the use of previously recorded data to predict future outcomes. In most cases, such as stock market prediction and weather forecasting, this technique will require full data without gaps. In the former case, the data is generally well documented at various resolutions. However, the latter is prone to missing values. This can be due to a number of reasons, including faulty recording equipment or maintenance. This issue raises the need for the imputation of the data. There are many ways to accomplish this task. The goal of this project is to answer the following question: **How does the imputation method impact the performance of our model? Does this change as the number of missing values changes?**

2 Data

The goal is to test the effect of the number of missing values and the imputation methods used. The number of missing values k is based on the percentage of data we wish to alter p. We wish to test low, moderate, and high values of $p \in \{0.1, 0.25, 0.4\}$. The imputation methods are as follows: 1) Kalman Smoothing (KS), 2) 3 day moving average (MA), and 3) linear interpolation (LI).

To simulate extended maintenance outages, we will insert sequences of five consecutive missing data points into the training and validation data only. The data sets will have roughly 10%, 25%, and 40% of their values missing. For the training set, this corresponds to 104, 259, and 414 sequences respectively. For the validation set, this corresponds to 22, 56, and 89 sequences respectively.

We have a complete dataset of N points (indexed from 0 to $N-1$). We sample indices from the set $\{3, 8, 13, \ldots, 5(n-1)+3\}$ without replacement where n is the quotient of $N-2$ and 5. This is critical since the imputation methods used will rely on having the first three data points or the final data point available.

Then at each sampled index i, we insert missing values at indices $\{i, i+1, i+2, i+3, i+4\}$. This is done using the three imputation methods to create three new datasets.

We can repeat this process multiple times to obtain different datasets for each $\langle p, \text{imputation} \rangle$ configuration. For now we will only be using data from a single station, and repeat the process 15 times (possibly more if runtime is not an issue).

3 Estimates

To evaluate the models, we will compare their RMSE. The RMSE will help us quantify the average error of our model. We pick the RMSE in particular because the model from Project 3 tended to underestimate high temperatures and overestimate low temperatures. The RMSE will penalize these errors more heavily. The RMSE was chosen over the MSE because it is in the original units, making it more interpretable.

4 Methods

In this experiment, we will not be testing different models / model parameters against each other. Instead, we aim to evaluate how a single model performs based on the amount of missing data and the imputation methods used. The model we are testing is the same as the one from Project 3. It is single layer LSTM model that maps a sequence of previously observed temperatures to the following day's temperature. The model is trained over 30 epochs with an MSE loss function and the Adam optimizer. We expect the model's performance to decrease as the p increases. We also expect the model to perform best with KS imputation and worst with LI imputation.

5 Performance Criteria

To evaluate the methods, we will compare the mean and standard deviation of the RMSE for each $\langle p, \text{imputation} \rangle$ configuration. We can show these results in a table, as well as graphically using box plots.

6 Simulation Plan

To perform this experiment, we generate 15 datasets for each $\langle p, \text{imputation} \rangle$ configuration. There are 3 imputation methods and values of p we wish to test. Therefore, we have a total of 9 $\langle p, \text{imputation} \rangle$ configurations, and 135 total datasets. The imputation methods and values of p are outlined in Section 2. To keep things consistent between all simulations, we will set the same seed before each training loop. We pick a seed of 42. We will record the loss for each model on their respective testing data. The loss is the MSE, so we can take its root to obtain our response variable in the RMSE. Once we have the RMSE for each model, we can compute the mean and standard deviation among the 15 models. Given that we want to find the bigger picture overall, we treat these 15 as a sample rather than the population of models.

7 Challenges & Limitations

The main issue with this experiment is the required runtime. We will need to train 135 different models. We can also save the model parameters so that we won't have to train the models over and over again.