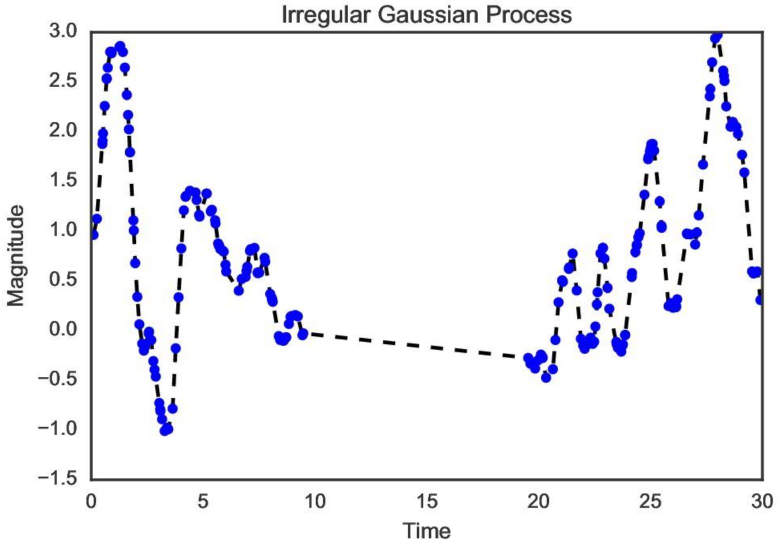
Deep Neural Networks for Irregular Noisy Time Series

Abhishek Malali, Pavlos Protopapas

With applications to Astronomical Time Series Prediction

Motivation

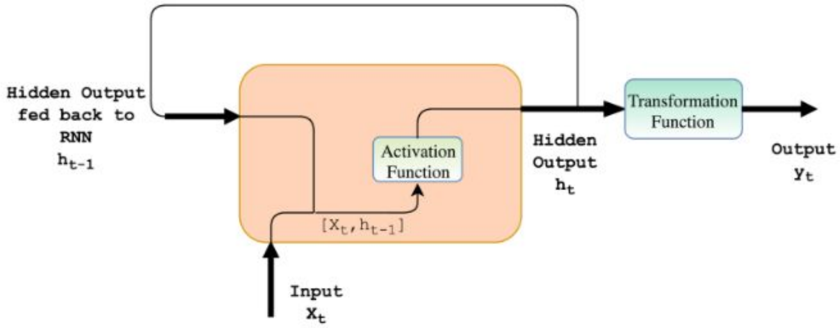
- Irregular time series can be found in transactional data, event logs and astronomy.
- Currently the series are converted into regular time series.
- Standard methods like ARIMA, Kalman filters and Markov models are used to predict.

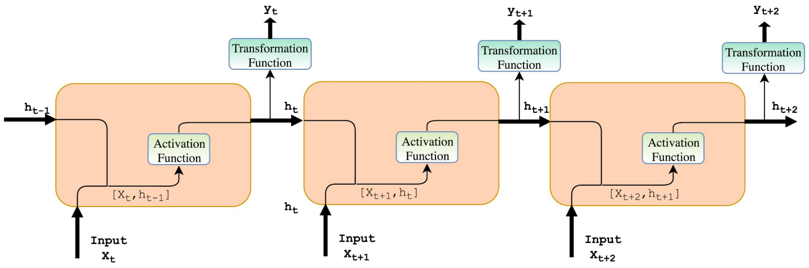


Example of an Irregular and noisy time series.

Recurrent Neural Networks (RNN)

- RNNs have been previously used in text and speech recognition.
- Suffer from vanishing gradient problem.
- Does not allow RNN units to remember long term dependencies.

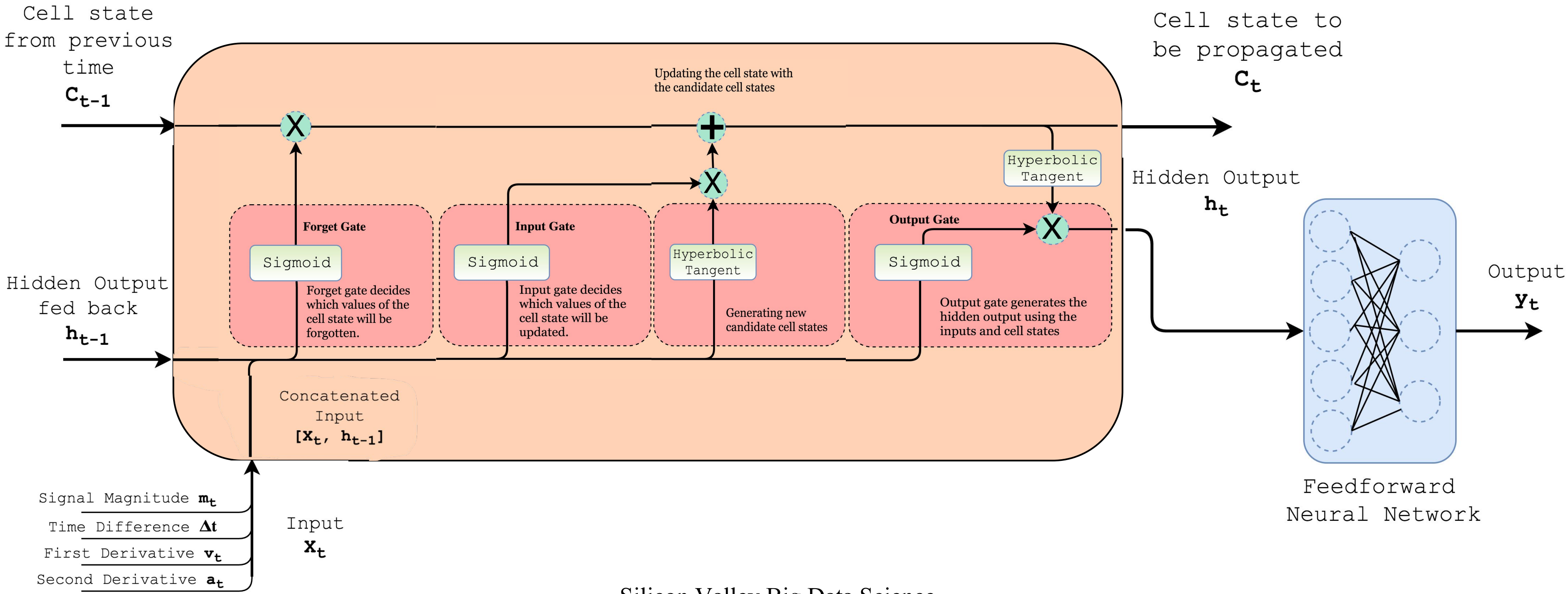




Long Short Term Memory(LSTM) Models

- LSTM models are capable of remembering long term tendencies in sequences and suitably adjust to the short term signal behaviour.
- Do not suffer from vanishing gradient problem due to addition of the cell memory state.
- The memory updates over time depending on the input.
- Powerful recurrent neural network architecture and adaptability to variable length sequences.

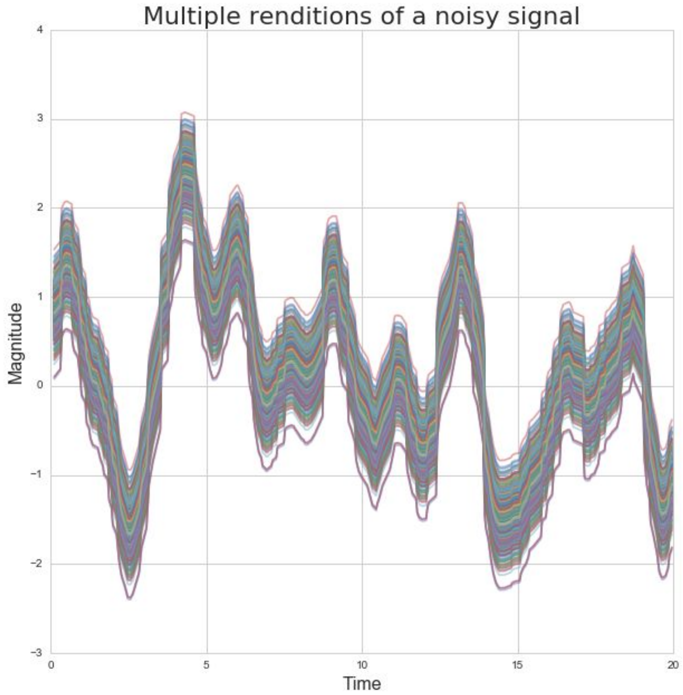
Network Architecture



Multiple Error Realizations

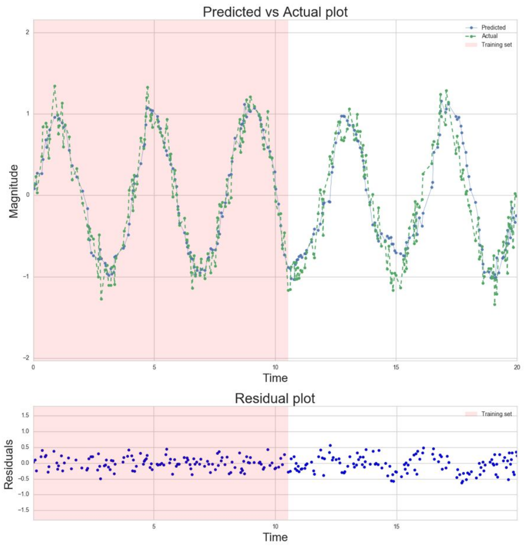
- With error budget for astronomical light curves, new realization of time series are generated.

- Results indicate that using multiple realizations of the model, helps the model become noise invariant and hence, predict better.

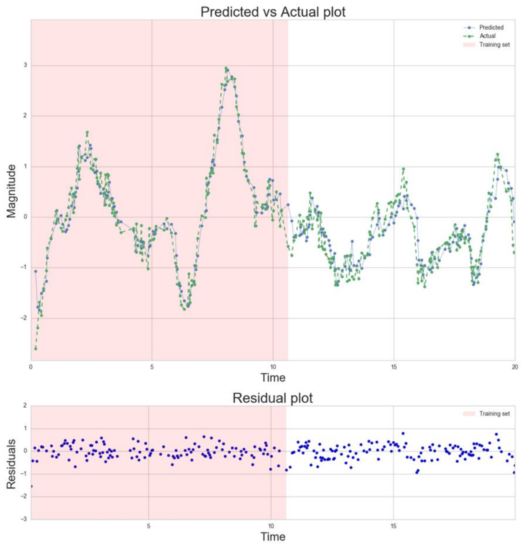


Results

1. Irregular Sinusoids

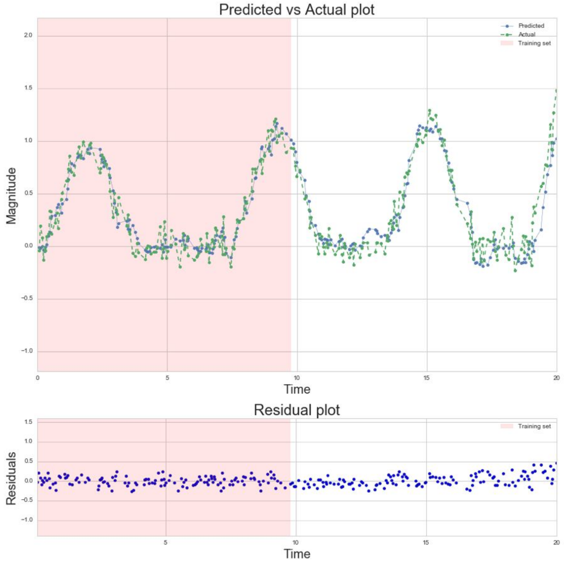


2. Irregular Gaussian Processes

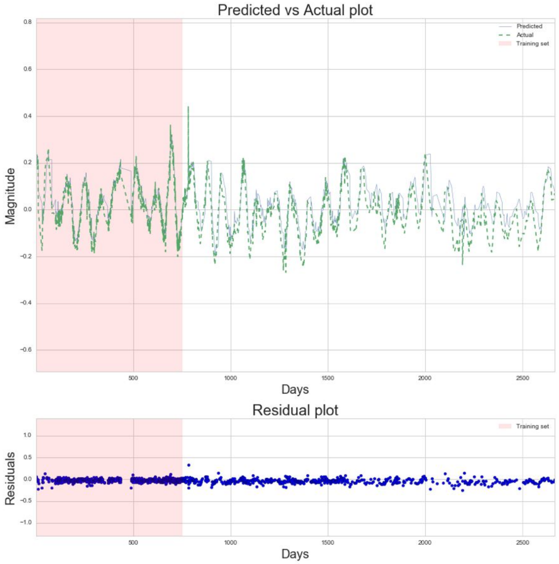


Results

3. Transient Signals

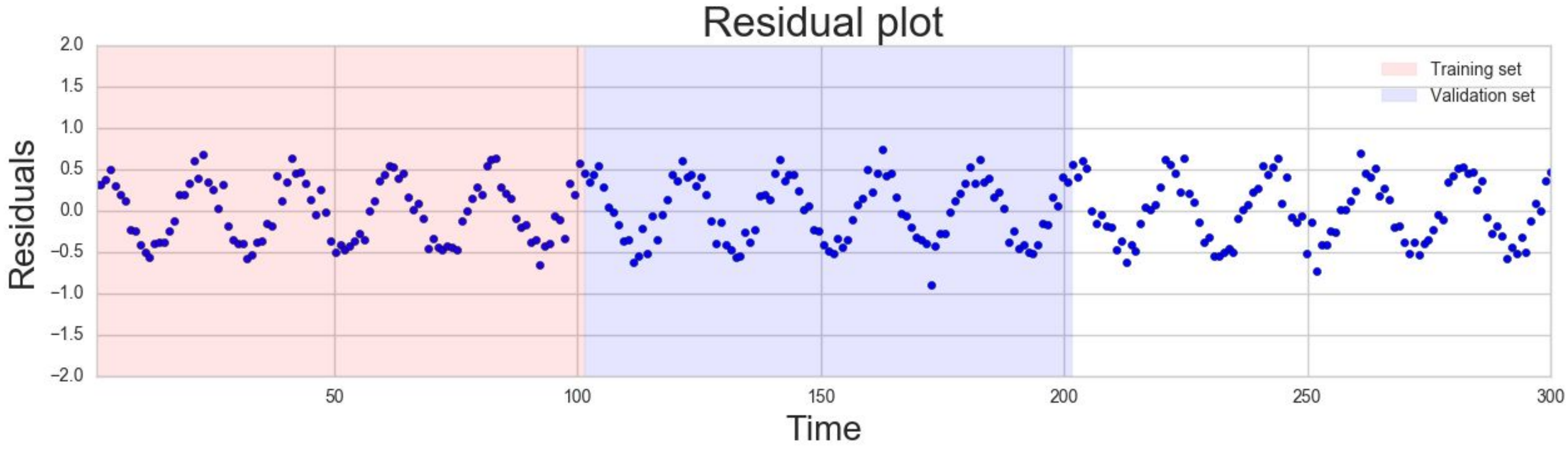


4. Long Period Variable stars



Correlations in Residuals

We find correlations in the residuals, which means improvement in the prediction can be made.



Correlations in Residuals

- For regular time series, the autocorrelation at lag k is defined as

$$\rho_k = \frac{\sum_{i=1}^{N-k}(R_i - \bar{R})(R_{i+k} - \bar{R})}{\sum_{i=1}^{N}(R_i - \bar{R})^2}$$

where *R* represents the residuals.

Correlation in Residuals

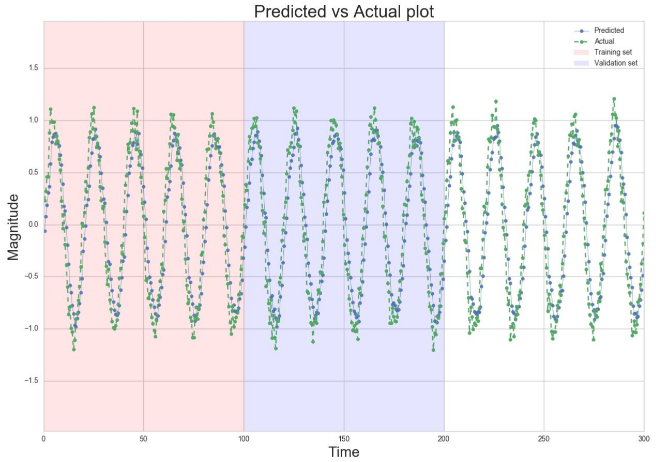
- The aim is to modify the Loss function in order to make the network learn not only to optimize on the RMSE but on the autocorrelation of the residuals as well.

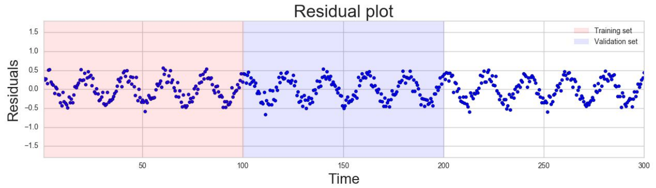
$$Loss = RMSE + \lambda\phi^2$$

- Φ here is the normalized autocorrelation of the previous n lags. [For our experiments, we chose this n to be 5]

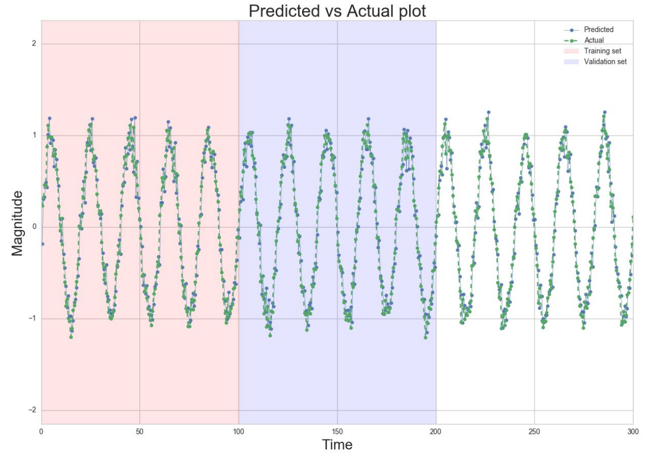
Results

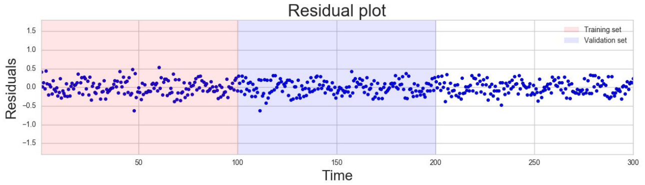
λ = 0





λ = 100





Correlation in Residuals

- Equation to define the correlated residual behaviour in irregular series

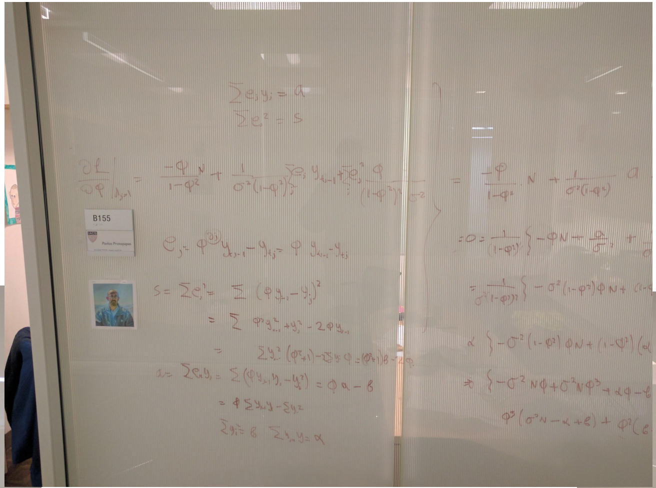
$$y_{t_j} = \phi^{\Delta t} y_{t_{j-1}} + \sigma\sqrt{1 - \phi^{2\Delta t}}\epsilon$$

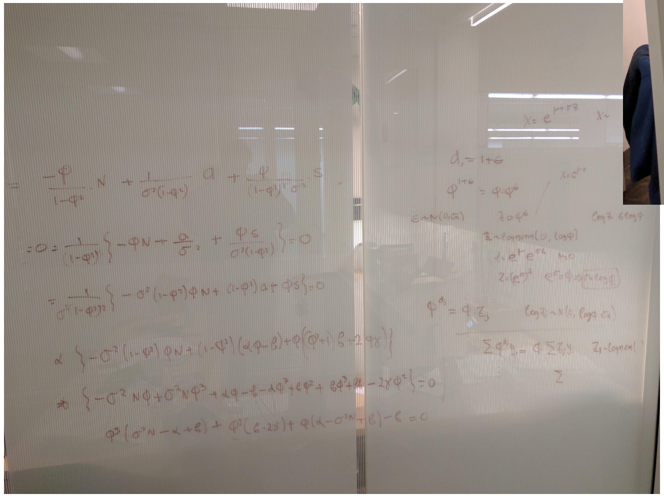
- Log Likelihood

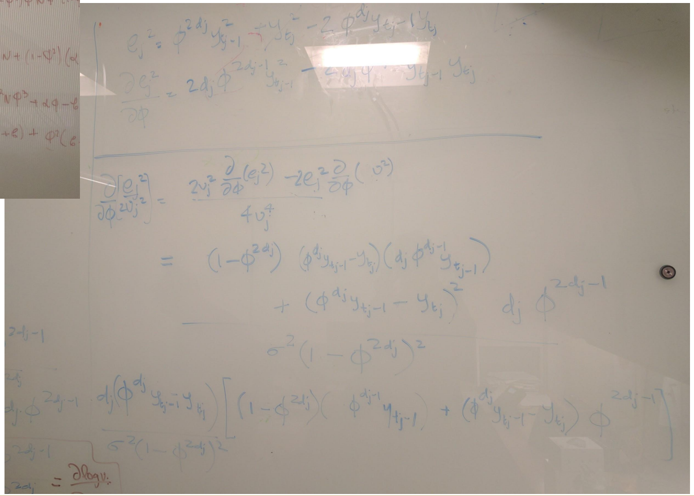
$$log(LL) = \sum_j log\nu + \sum_j \frac{e^2}{2v^2}$$

$$\nu = \sigma\sqrt{1 - \phi^{2d_j}} \qquad d_j = t_j - t_{j-1} \qquad e = \phi^{d_j} y_{t_{j-1}} - y_{t_j}$$

Correlation in Residuals







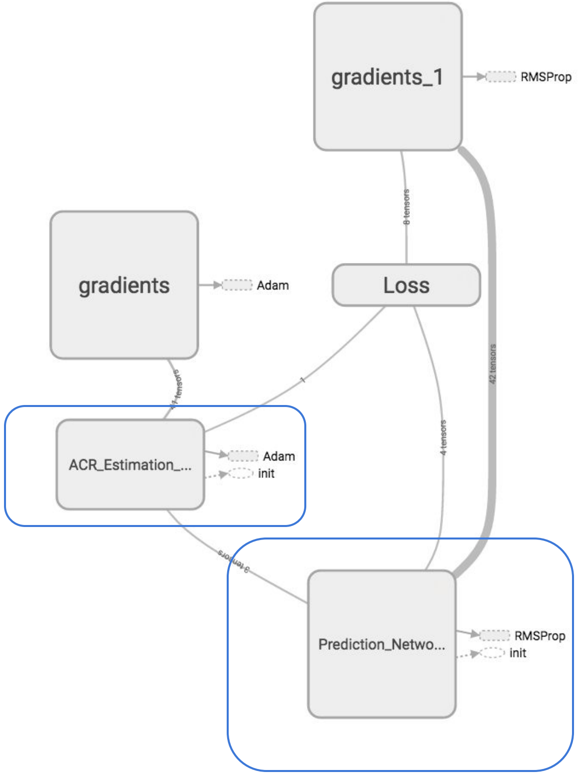
Correlation in Residuals

- The autocorrelation Φ is estimated by minimizing the log-likelihood using stochastic gradient descent.

- Once Φ is determined, gradients are propagated during training to modify the LSTM weights. During this process, the network used to estimate Φ is static.

Network Architecture in TensorFlow

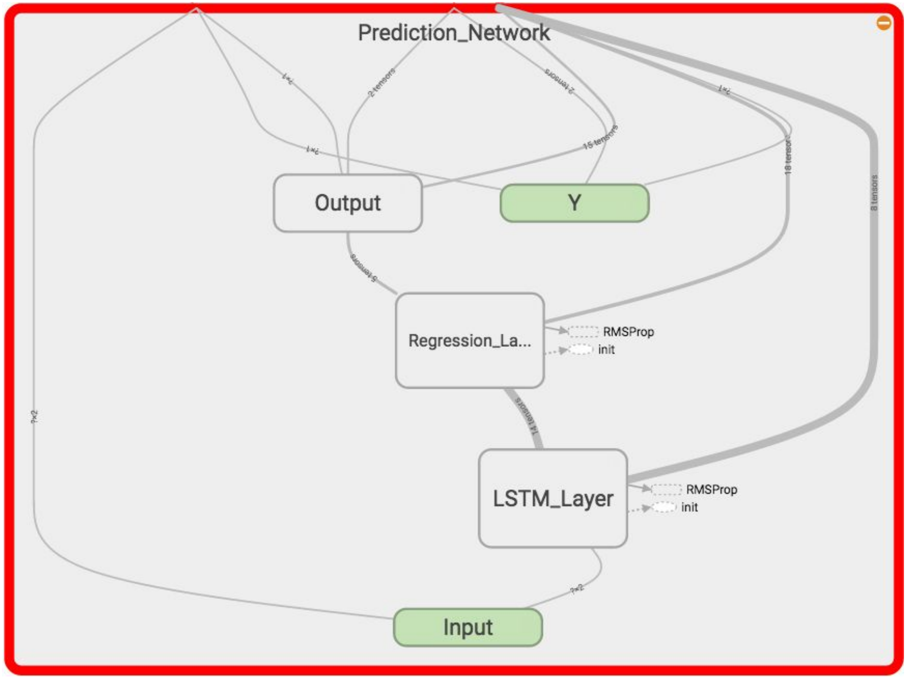
Autocorrelation Estimation Network



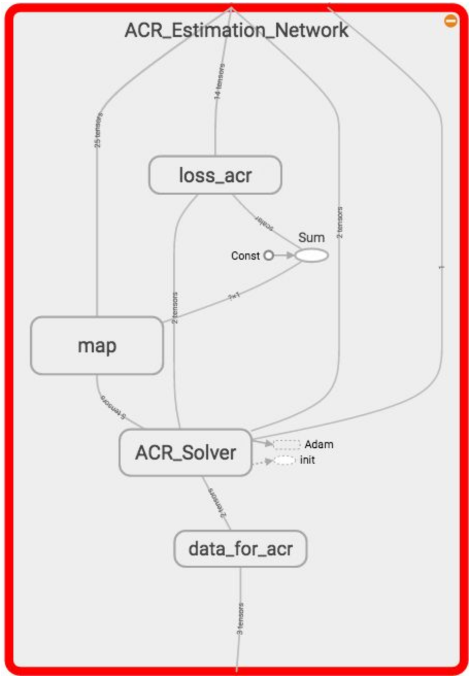
Prediction Network

Network Architecture in TensorFlow

Prediction Network



Autocorrelation Estimation Network



Additional work

- Focusing on LSTM based time series classifiers for variable length sequences.

- Building a Tensorflow based time series machine learning library. Can be found at https://github.com/abhishekmalali/TimeFlow

- Time series generation library(TimeSynth). The repo can be found at https://github.com/TimeSynth/TimeSynth