Microlensing Event Detection with RNN and Hybrid Models

Dibyajit Banerjee, Justin Mae, and In-Ho Yi

New York University

Abstract

We study the use of RNN, ARIMA-RNN and GP-RNN hybrid models in detecting microlensing events. We both simulate the data and use existing data to validate the performance of the baseline and hybrid model.

1 Introduction

Gravitational microlensing is a technique used to detect background objects, such as, exoplanets and dark matter, by detecting abnormal light curves caused by gravitational effects of a foreground object that overshadows it. The light rays forms a ring known as Einstein ring. With this method, we can discover the existence of exoplanets (planets that are located more that thousands light years away from the observation point)that are otherwise hard to detect. However, such events are rare, and given the rarity it is helpful to be able to detect the event at its earliest onset.

The primary objective of this project is to implement an early warning system as described in [1], and extend on the idea to experiment with other types of RNN configurations, and the use of GP in place of ARIMA for more accurate prediction. In contrast to [1] and other approaches, we develop an end-to-end model with RNN, namely we train the model to emit the probability of the anomaly occurring as an output.

2 Method

Models are trained to give a series of probabilities of the event occurring at given time segments. Light curves are fed into the model. The light curves are normalized with first 20% of the data.

The performance of the baseline model is then augmented with ARIMA. The main idea is to produce in-sample rolling predictions with preceding data set with the length of 20% of the total data, and we feed the prediction of ARIMA as a side-channel, or we only feed the residual to the model.

2.1 Simulated Data

One challenge of this project is to acquire the data. The Mini-GWAC data used in [1] is not yet public. In order to proceed with the project, we produce simulated data set ([2]), as explained in detail in [3].

Simulated light curve has three terms:

$$X(t) = I_{s0}(t) + A(t) + w_t$$

I_{s0} is a baseline periodic light curve, and is simulated by the formula:

Preprint. Under review.

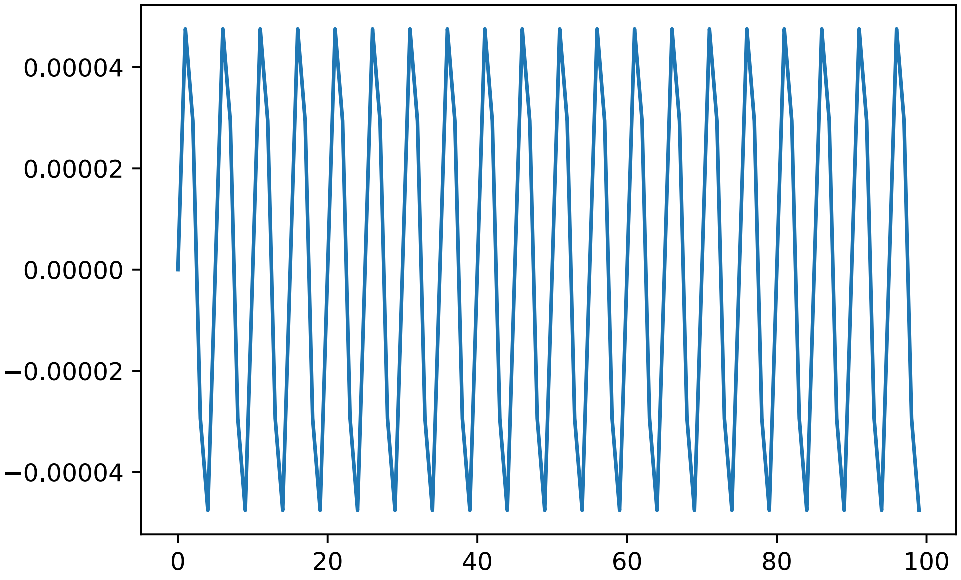

Figure 1: Baseline Curve Example

$$I_{s0}(t) = m_A \sin\left(\frac{2\pi t}{T} + \phi\right)$$

where m_A is amplitude, T is period, and ϕ is phase. Figure 1 shows an example of the baseline curve.

$A(t)$ is Lens Magnification, i.e. jump in the light curve due to the Microlensing effect. It is given by:

$$A(t) = \frac{[u(t)]^2 + 2}{u(t) + \sqrt{[u(t)]^2 + 4}}$$

$$u(t) = u_0 + \left|\frac{t - t_0}{T_E}\right|$$

where u_0 is minimum impact parameter, T_E is the radius (i.e. determines how long the effect lasts), and t_0 determines shift in time (i.e. when the effect is at its peak). Figure 2 shows an example of Lens Magnification curve.

For Train Set, we generated 500 curves, each with 1000 time segments. 100 more such curves are generated for Test Set. For Dev Set, we generate 100 curves, each with 2000 time segments. Note that, in order to assess generalization of the model, we made a slight variation on the Dev Set.

Parameters to the graphs are experimentally determined to be the following:

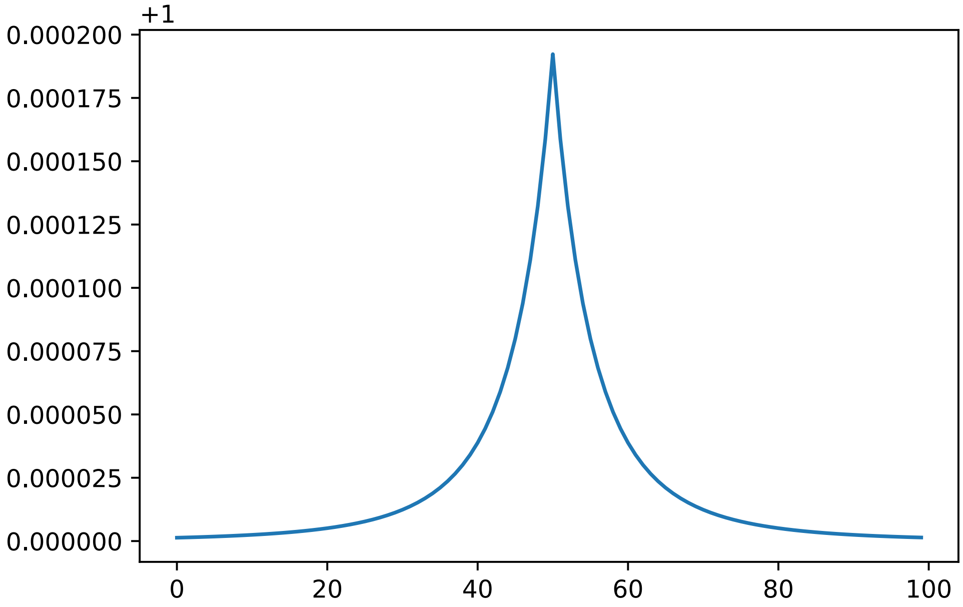

Figure 2: Lens Magnification Example

$$\begin{aligned}
w_t &\sim \mathcal{N}(0, 0.00003) \\
m_A &\sim \text{Uniform}[0.00008, 0.0002] \\
T &\sim \text{Uniform}[2, 5] \\
\phi &= 0 \\
u_0 &\sim \text{Uniform}[8, 12] \\
T_E &\sim \text{Uniform}[5, \frac{L_T}{20}] \\
t_0 &\sim \text{Uniform}[\frac{L_T}{5} + 10, L_T - \frac{L_T}{5} - 10]
\end{aligned}$$

where L_T is the number of time segments in the series i.e. 1000 for Train and Test Set, and 2000 for Dev Set.

Half of the data sets are generated with Lens Magnification curve. The other half are generated without one. First 20% of the series is used to normalize the data. Time interval $[t_0 - 1.5 \times T_E, t_0 + 1.5 \times T_E]$ is labeled to be 1. All other time segments are labeled 0.

Figure 3 shows samples of the train data generated.

2.2 Real Data

We prepared real-world microlensing light curve to analyze the performance of our models. In this subsection, we explain where we sourced the data and pre-processed the data.

2.2.1 MOA

Microlensing Observations in Astrophysics (MOA) is the set of observations made by joint New Zealand/Japan collaboration [4].

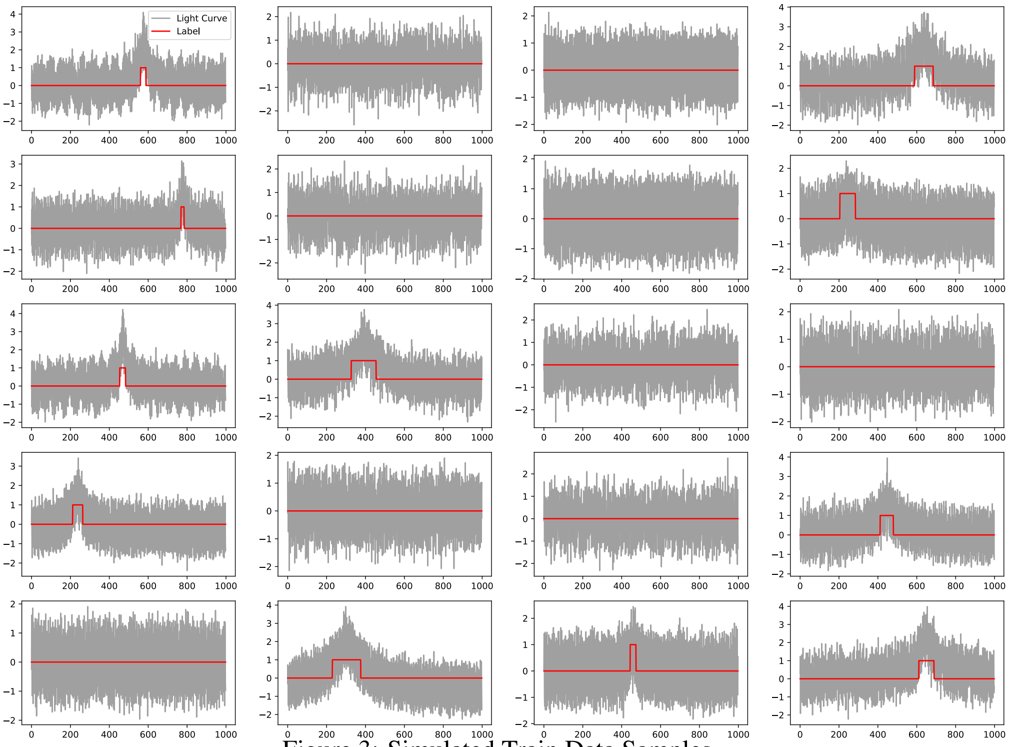

Figure 3: Simulated Train Data Samples

We read flux data from Field 18, convert to AB magnitude, invert the curve to get the light amplitude data, and filter for the known microlensing light curves by cross-referencing the alert data available alongside with the data set.

The formula we used is the following:

$$X = 2.5 \times \log\left(\frac{F}{691.8}\right)$$

where F is the flux data.

We then took the mean and the standard deviation of the first 20% of the graph to normalize the curve, and filled in zeroes for null data points. It turned out that filling in zeroes was enough to make the model work on the real data, so we didn't explore more advanced ways of handling null data points.

2.2.2 UKIRT

United Kingdom Infrared Telescope (UKIRT) located in Mauna Kea Observatory in Hawaii has been used to conduct microlensing studies.

We looked at magnitude data from Field 11 and applied the same normalization and pre-processing for missing data as for MOA.

UKIRT data did not come with event data so it was difficult to test our model to see if it would detect microlensing events. For future studies, UKIRT would be a good dataset where this model could be applied to label microlensing events. For our investigation, we proceeded with MOA as the real light curve.

2.3 Simulation to Real Data

We have explained that our choice of parameters for simulated graphs are informed by the performance of the models on the real data. To illustrate this point, take lens radius T_E which determines how long the effect lasts.

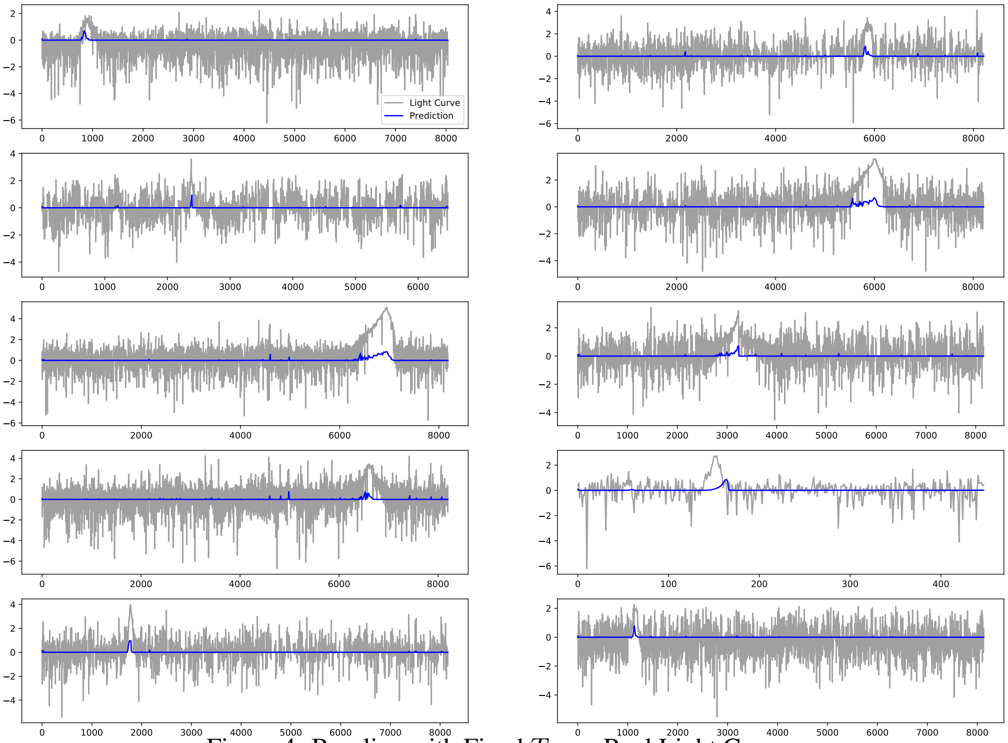

Figure 4: Baseline with Fixed T_E on Real Light Curve

Figure 4 is the prediction made by the baseline model trained with fixed T_E value, whereas Figure 9 is by the model trained with randomized T_E. When we have fixed T_E, the model is constrained to limit the length of alerts, whereas with randomized T_E the model learns to deal with the alerts with varying lengths.

2.4 Evaluation

We train the model with Train Set, and we keep track of F1 score on Train and Test Set as we progress in each epoch, to produce the learning curve.

We then look at the precision-recall and F1 graph on Dev Set with varying threshold, and examine at which threshold the F1 score is maximized.

Finally, we take the best model in each classes of models, and inspect the predictions made on the real data set.

3 Models and Results

3.1 Baseline RNN

We examine the performance of RNN models in predicting the event in four different settings:

- LSTM 1: Single-layer LSTM with 200 hidden units and tanh activation function, fed into fully-connected network with single sigmoid output function
- LSTM 2: Double-layer LSTM, first with 200 units and second with 100 units, both with tanh activation function, fed into fed into fully-connected network with single sigmoid output function
- GRU 1: Single-layer GRU with 200 hidden units and tanh activation function, fed into fully-connected network with single sigmoid output function
- GRU 2: Double-layer GRU, first with 200 units and second with 100 units, both with tanh activation function, fed into fed into fully-connected network with single sigmoid output function

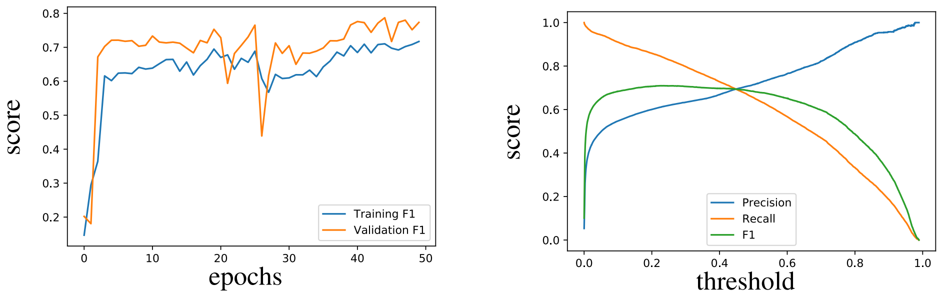

Figure 5: Learning Curve and Precision-Recall-F1 Curve for LSTM 1

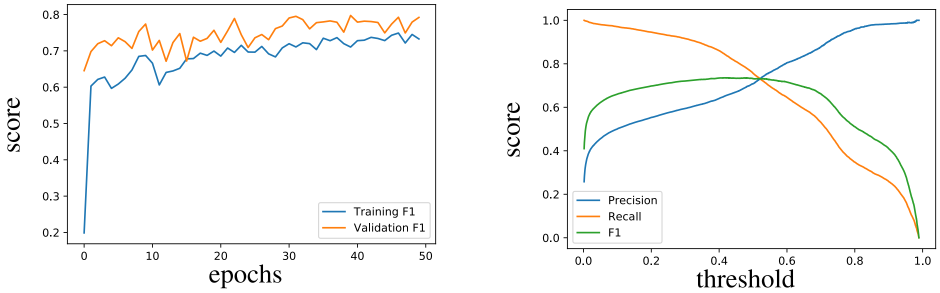

Figure 6: Learning Curve and Precision-Recall-F1 Curve for LSTM 2

Figures 5 and 6 show the performance of the LSTM networks. It is evident that having double-layer architecture is beneficial in terms of the training and performance. Figures 7 and 8 show the performance of the GRU networks.

Here are the results of the F1 scores on Dev Set at its best performing threshold:

- LSTM 1: F1: 0.7097, Threshold: 0.24
- LSTM 2: F1: 0.7358, Threshold: 0.41
- GRU 1: F1: 0.7428, Threshold: 0.26
- GRU 2: F1:: 0.7592, Threshold: 0.34

GRU 2 is the best performing baseline. Going forward, we will use GRU 2 as the reference baseline model.

Figure 9 shows predictions of GRU 2 on the real light curve. Overall, the model does the good job of predicting the events. However, predictions can be a little noisy in places where the curves are jumpy.

3.2 ARIMA-RNN

We tried to improve on the baseline GRU model by training an ARIMA model and feeding the predictions or residuals along with the original data into a two-channel GRU model. The ARIMA models were trained using 20% of the data and the remaining 80% were in-sample rolling predictions. We trained the following models:

- ARIMA (0,0,1)
- ARIMA (0,0,2)
- ARIMA (1,0,0)
- ARIMA (1,0,1)

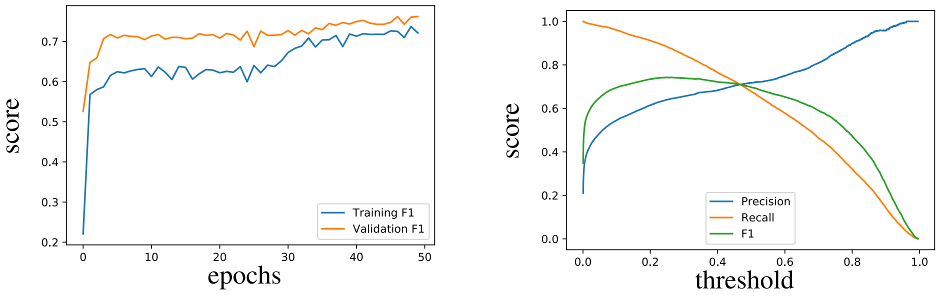

Figure 7: Learning Curve and Precision-Recall-F1 Curve for GRU 1

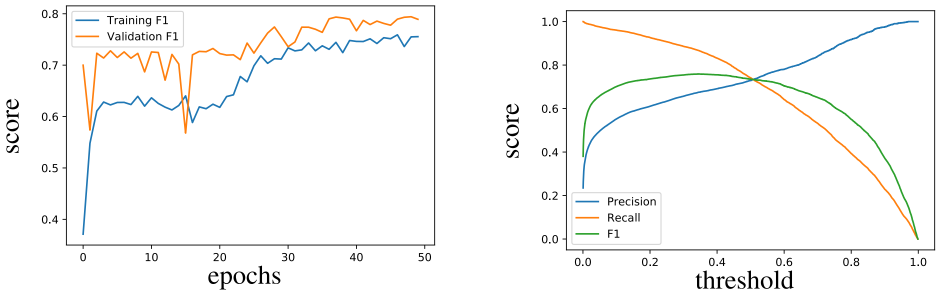

Figure 8: Learning Curve and Precision-Recall-F1 Curve for GRU 2

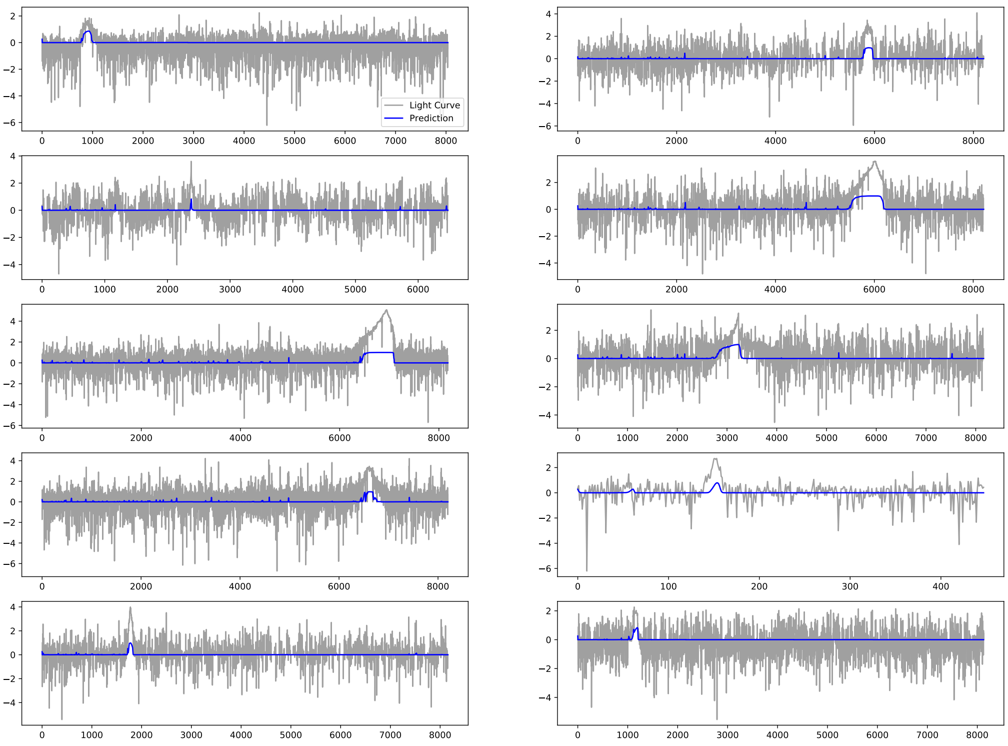

Figure 9: GRU 2 Prediction on Real Light Curve

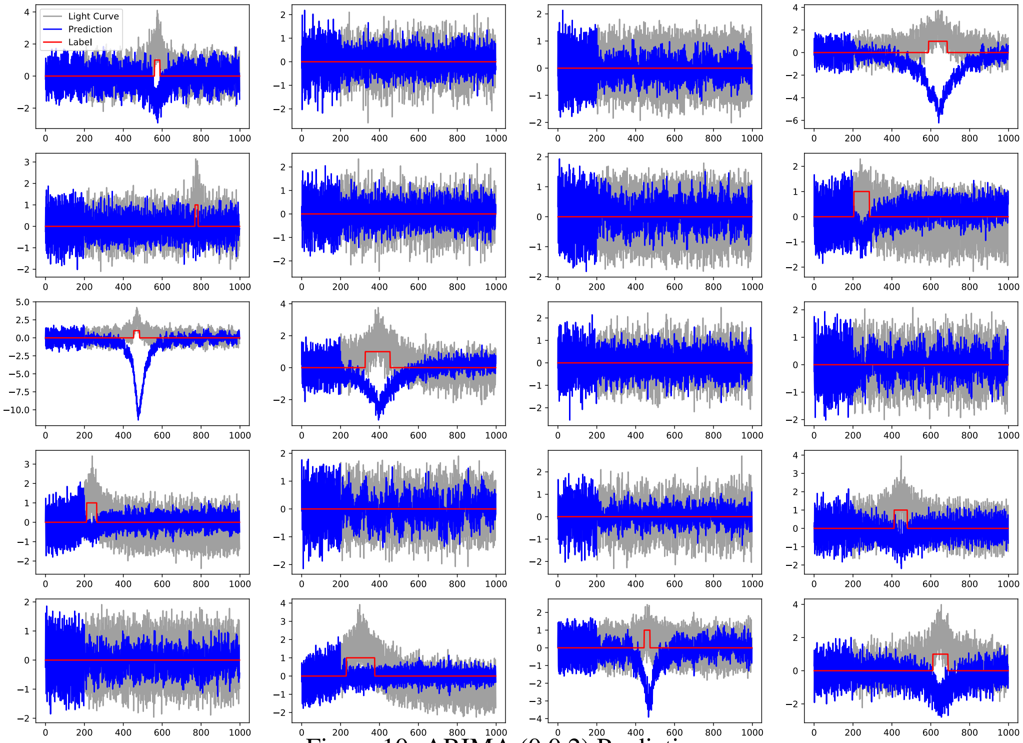

Figure 10: ARIMA (0,0,2) Predictions

- ARIMA (1,1,1)

Figure 10 show the predictions of ARIMA (0,0,2) model. The ARIMA predictions are inverted to the data because it was fit only on the first 20% where there aren't any events.

Figure 11 show the residuals of ARIMA (0,0,2) model. The residuals did not make much of a difference when it came to performance. This is most likely due to the model not having any new information from data and predictions, since the residual is just the difference of the two.

Here are the results of the F1 scores on Dev Set at its best performing threshold:

- GRU-ARIMA (0,0,1) Predictions: F1: 0.7556 Threshold: 0.37
- GRU-ARIMA (0,0,2) Predictions: F1: 0.7888 Threshold: 0.31
- GRU-ARIMA (1,0,0) Predictions: F1: 0.7631 Threshold: 0.37
- GRU-ARIMA (1,0,1) Predictions: F1: 0.7441 Threshold: 0.31
- GRU-ARIMA (0,0,1) Residuals: F1: 0.7711 Threshold: 0.32
- GRU-ARIMA (0,0,2) Residuals: F1: 0.7742 Threshold: 0.28
- GRU-ARIMA (1,0,0) Residuals: F1: 0.7727 Threshold: 0.38
- GRU-ARIMA (1,0,1) Residuals: F1: 0.7472 Threshold: 0.30

We found that GRU-ARIMA (0,0,2) predictions performed the best and proceeded to apply this model to the real light curve, MOA dataset. ARIMA (1,1,1) results were not included because predictions would explode for certain data.

Figure 12 show the performance of the GRU-ARIMA (0,0,2) model.

Figure 13 show the predictions of GRU-ARIMA (0,0,2) prediction model on the real light curve. Although this model performed the best on simulated dataset, it did not perform the best on the real light curve. The eight graph shows that GRU-ARIMA (0,0,2) failing to identify the light curve.

Based on visual analysis, GRU-ARIMA(1,0,1) residual performs the best and appears to have a stronger signal than the baseline. The predictions also seem to have less noise. Figure 14 shows the predictions of GRU-ARIMA(1,0,1) residual model on the real light curve.

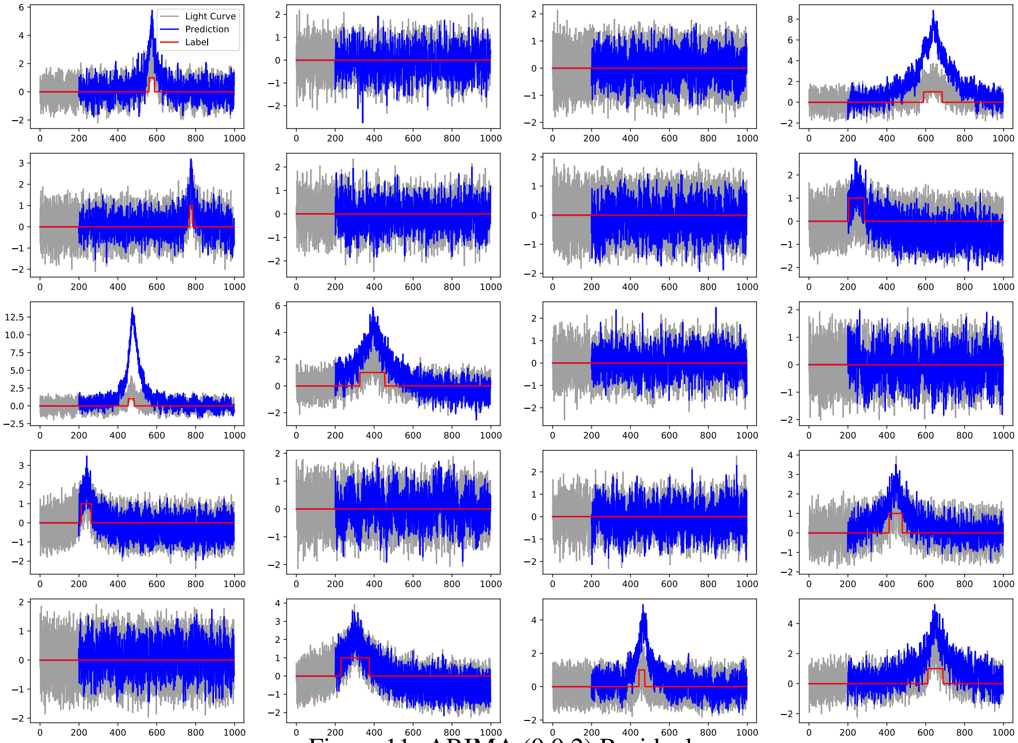

Figure 11: ARIMA (0,0,2) Residuals

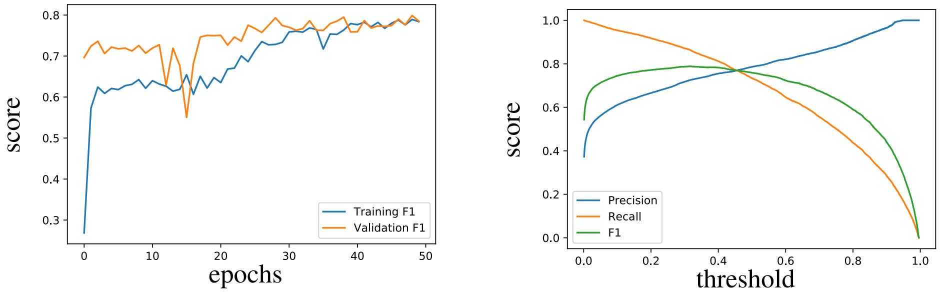

Figure 12: Learning Curve and Precision-Recall-F1 Curve for GRU-ARIMA (0,0,2)

3.3 GP-RNN

We also used GP(Gaussian Process)-RNN for prediction and comparing the result with ARIMA-RNN. Since GRU 2 performed best among the RNNs, we used it for this model. One of the challenges with Gaussian Process is the high computational cost associated with the training process therefore a subset of data associated with one light curve with 1000 time segments is used for training. Also there is a high storage cost associated with storing the covariance matrix therefore one light curve with 1000 time segments of test data is used for generating the covariance matrix although the entire test data is used for getting the other statistical properties.

The following kernels have been used for comparing different GP models:

- Exponential Sine Squared
- Mixed kernel (Constant + Matern(3/2) + White)
- Matern(3/2)
- Rational Quadratic
- RBF

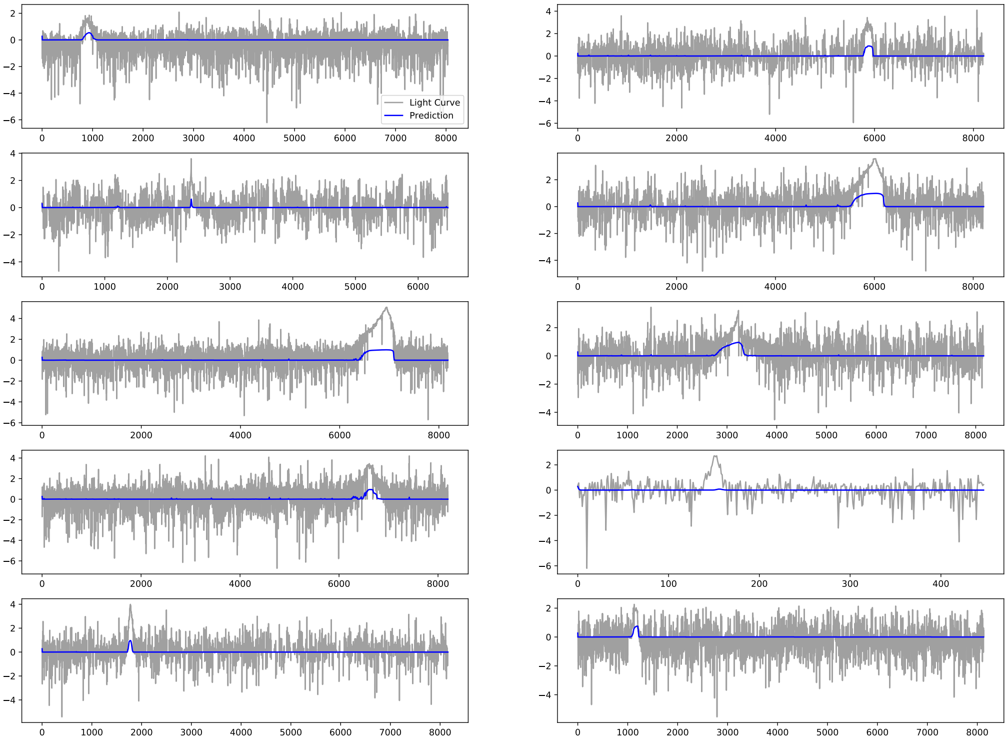

Figure 13: RNN-ARIMA (0,0,2) Prediction on Real Light Curve

- Spectral Mixture

By fitting the training data we obtained the statistical properties like mean, standard deviation, covariance, confidence bound and residual. These results are plotted in the following sections.

Mean
We found the mean to be constant irrespective of the kernel used. Figure 15.

Standard deviation
The standard deviation curves are similar for Mixed kernel, Matern(3/2), Rational Quadratic, RBF. Figure 15 has the curves of standard deviation for different kernels.

Covariance Matrix
The covariance matrix for Mixed kernel, Matern(3/2), and Spectral Mixture kernels found to be diagonal whereas for other kernels its quite different. It seems quite interesting for Rational Quadratic and RBF. See Figure 16.

Confidence bound
The confidence bound has been created at one standard deviation above and below the mean. It is very hard to detect the variation in confidence and variation in the test data values on a single scale as they vary at different precision. See Figure 16.

Residual / Standard deviation
Since the mean does not shows any variation with the different time segments therefore the residual has been divided by the standard deviation to get the deviation of GP output with the actual values. This deviation is expected to be higher in abnormal case i.e. in case of a microlensing event when the intensity of light goes too high. Figure 17 has the curve for residual / standard deviation.

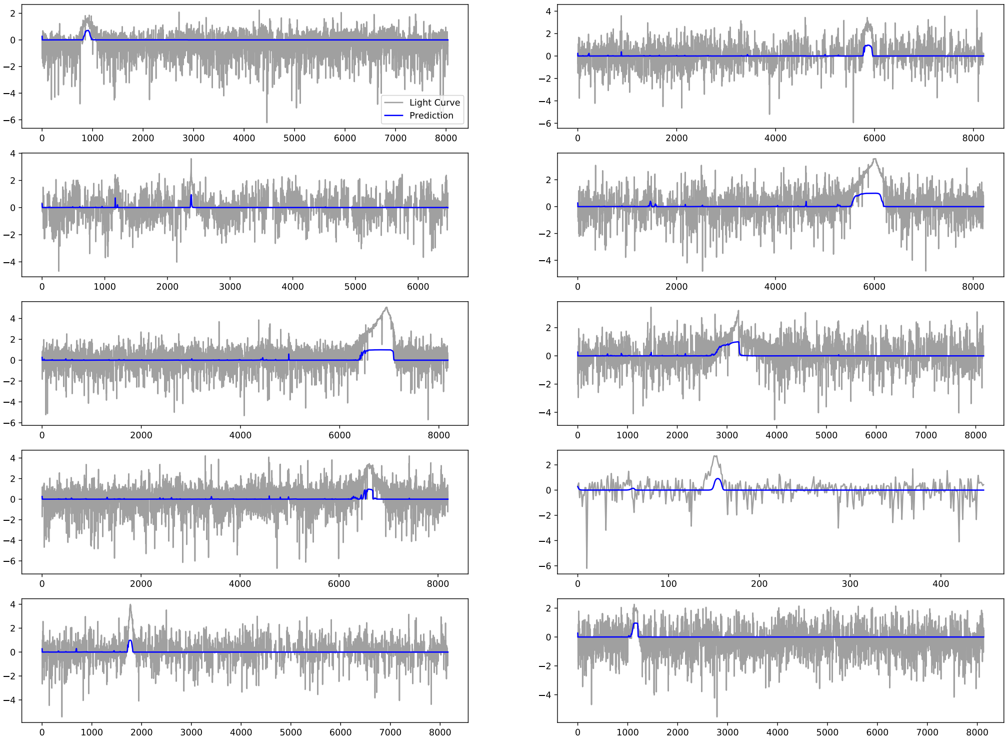

Figure 14: RNN-ARIMA (1,0,1) Residual on Real Light Curve

This value is used for feeding into the RNN to get the actual predictions. For feeding into the RNN the residuals and standard deviations given by the GP model with Rational Quadratic kernel have been used. See Figure 17.

Although the GP-RNN model provides a fair accuracy of around 90% but the F1 score is just 0.5258 which is nowhere close to that of ARIMA-RNN. But through the prediction curve it can be observe that this model predicts microlensing event at much early stage. Which is a useful trait to trigger an early warning system.

4 Conclusion

With simulated data sets to train the RNN based models, we have managed to produce models that makes reasonable predictions on the real light curves.

We found that the predictions generated by ARIMA provide additional information to the original data and boosts the F1 scores over the baseline model. On the real light curve, RNN-ARIMA model displayed stronger and cleaner signals than the baseline RNN model.

Although the GP-RNN model provides a low F1 score which is a very important evaluation ground since microlensing is a very rare phenomenon and a false negative would cost a lot but it provides a very early detection which can be helpful as ultimately an early warning system can be triggered in case of a microlensing event.

Appendix

MOA Data Used

Here is the list of MOA light curves shown in this paper:

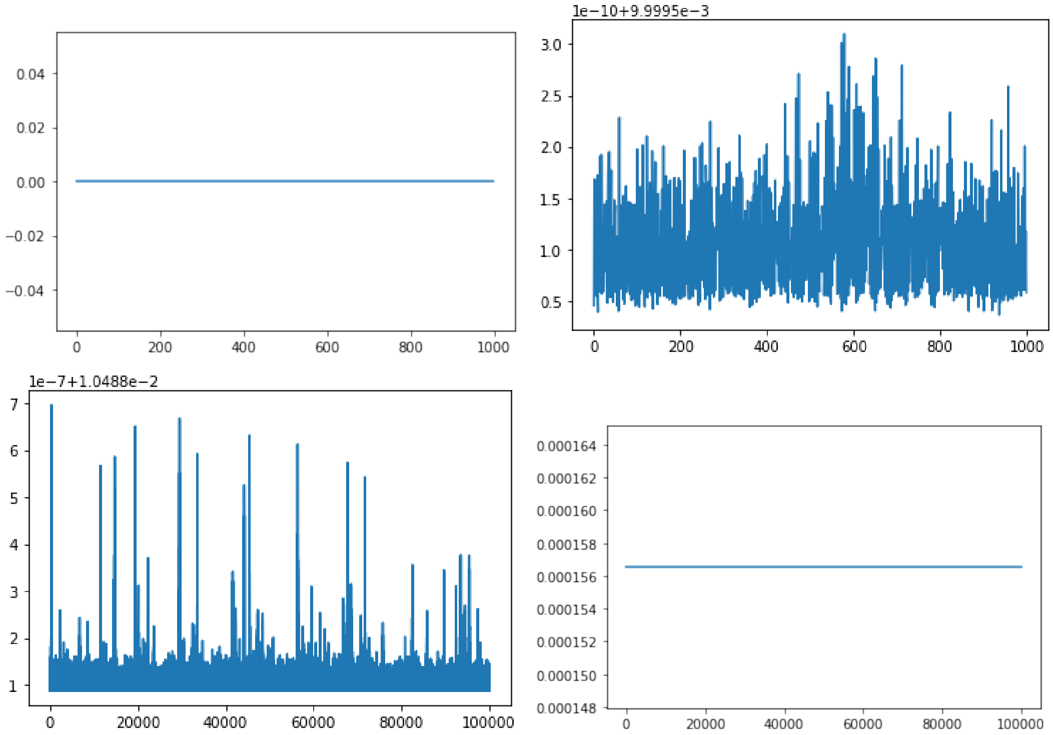

Figure 15: Top left: Mean corresponding to all kernels, Top right: Standard deviation curve for Exponential Sine Squared kernel, Bottom left: Standard deviation curve for Mixed , Matern, Rational Quadratic, RBF kernels Bottom right: Standard deviation curve for Spectral Mixture kernel

- gb18/R/9/gb18-R-9-4-30462.phot
- gb18/R/9/gb18-R-9-6-137778.phot
- gb18/R/9/gb18-R-9-3-87282.phot
- gb18/R/9/gb18-R-9-2-135512.phot
- gb18/R/9/gb18-R-9-5-156745.phot
- gb18/R/9/gb18-R-9-0-85169.phot
- gb18/R/2/gb18-R-2-7-215114.phot
- gb18/R/2/gb18-R-2-4-119901.phot
- gb18/R/2/gb18-R-2-3-72409.phot
- gb18/R/2/gb18-R-2-2-45059.phot

Acknowledgement

This paper makes use of data obtained by the MOA collaboration with the 1.8 metre MOA-II telescope at the University of Canterbury Mount John Observatory, Lake Tekapo, New Zealand. The MOA collaboration is supported by JSPS KAKENHI grant and the Royal Society of New Zealand Marsden Fund. These data are made available using services at the NASA Exoplanet Archive, which is operated by the California Institute of Technology, under contract with the National Aeronautics and Space Administration under the Exoplanet Exploration Program.

References

[1] Ying Sun, Zijun Zhao, Xiaobin Ma, and Zhihui Du. Short-timescale gravitational microlensing events prediction with arima-lstm and arima-gru hybrid model. In Jianhui Li, Xiaofeng Meng,

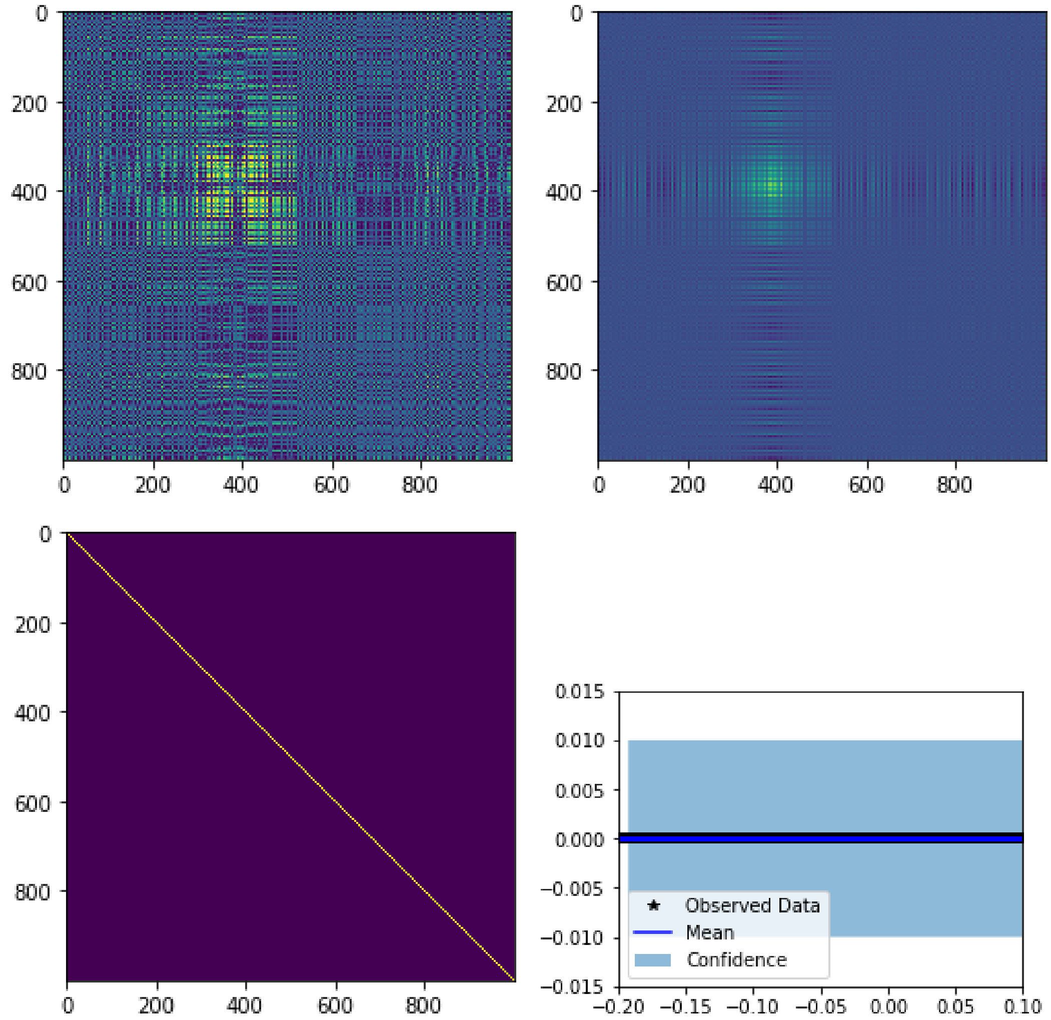

Figure 16: Top left: Covariance matrix structure for Exponential Sine Squared kernel, Top right: Covariance matrix structure for Rational Quadratic and RBF kernels, Bottom left: Covariance matrix structure for Mixed, Matern(3/2), and Spectral Mixture kernels, Bottom right: Confidence bound of the GP predictions along with true values kernels

Ying Zhang, Wenjuan Cui, and Zhihui Du, editors, *Big Scientific Data Management*, pages 224–238, Cham, 2019. Springer International Publishing.

[2] Zhihui Du. Private Communication, 2019.

[3] Jiaming Qiu, Yankui Sun, Chao Wu, Zhihui Du, and Jianyan Wei. Nfd: Toward real-time mining of short-timescale gravitational microlensing events. *Publications of the Astronomical Society of the Pacific*, 130:104504, 10 2018.

[4] Orlon Petterson and Canterbury University. Moa, microlensing observations in astrophysics, 2019.

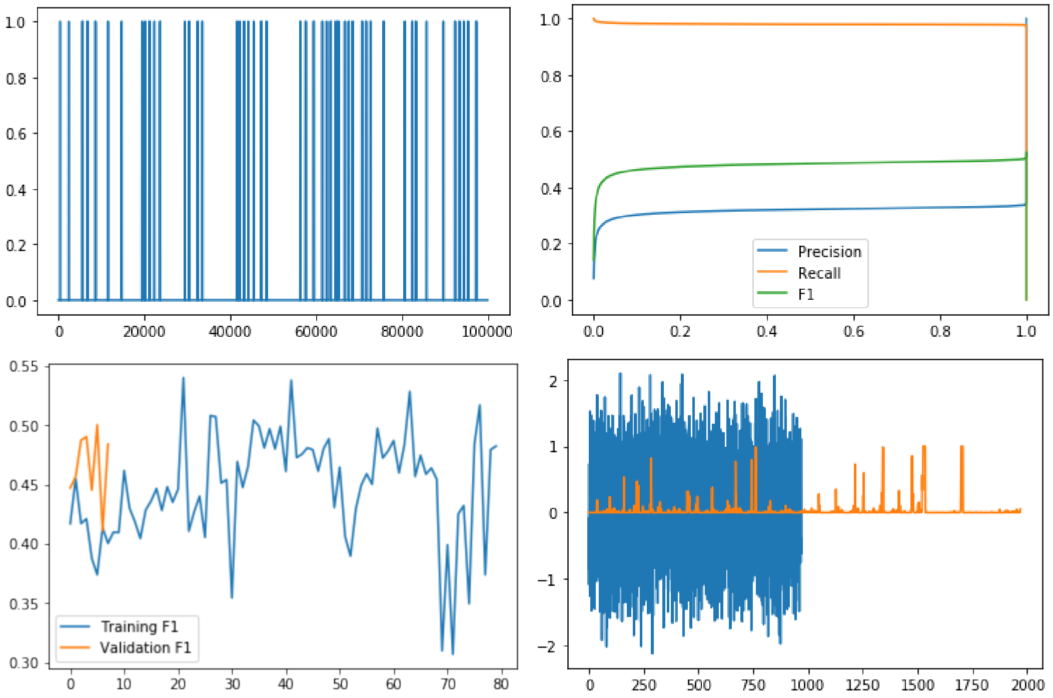

Figure 17: Top left: Residual/standard deviation curve against time-segments, Top right: Precision-Recall-F1 curve for GP-RNN with score against threshold, Bottom left: Learning curve for GP-RNN with score against epochs, Bottom right: GP-RNN prediction curve with the test data

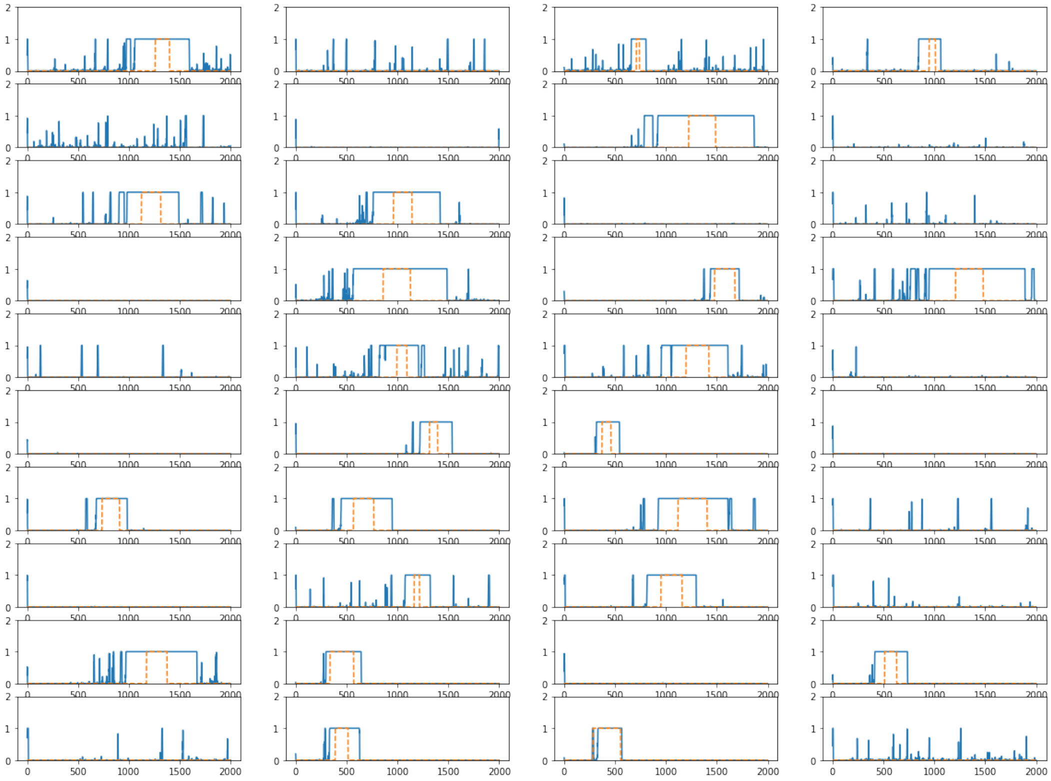

Figure 18: The evaluation of GP-RNN model

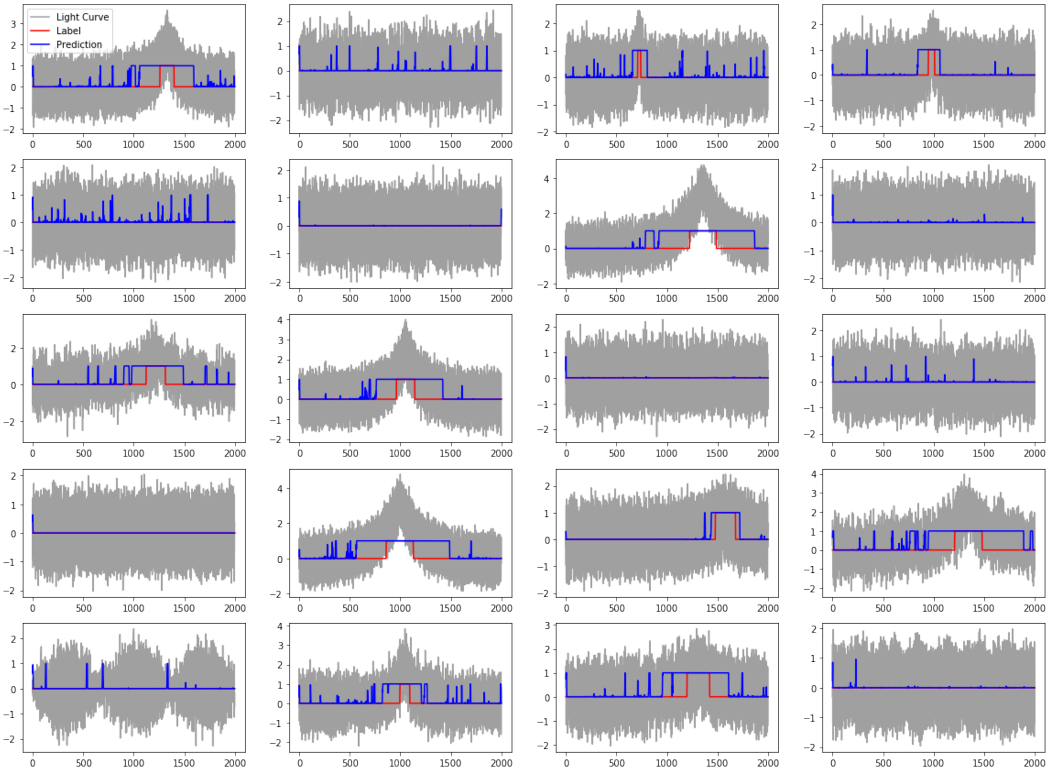

Figure 19: GP-RNN prediction vs the actual data