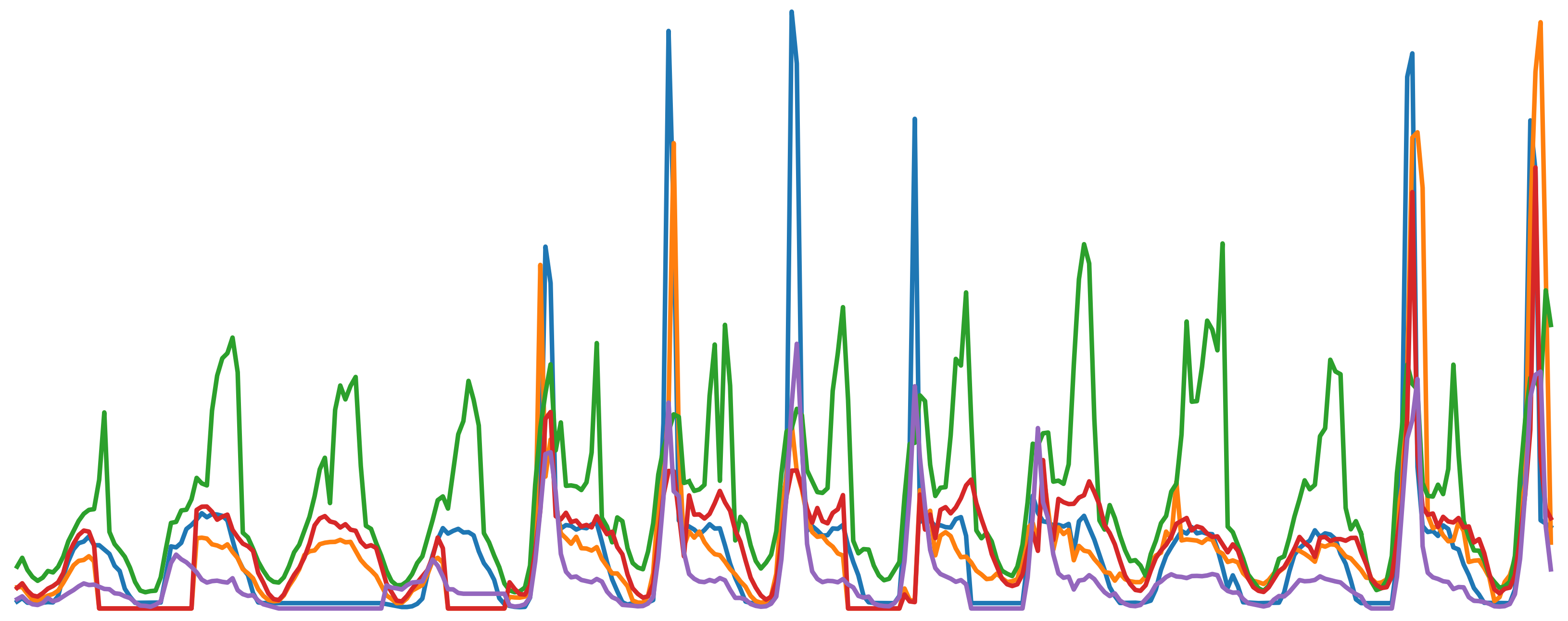
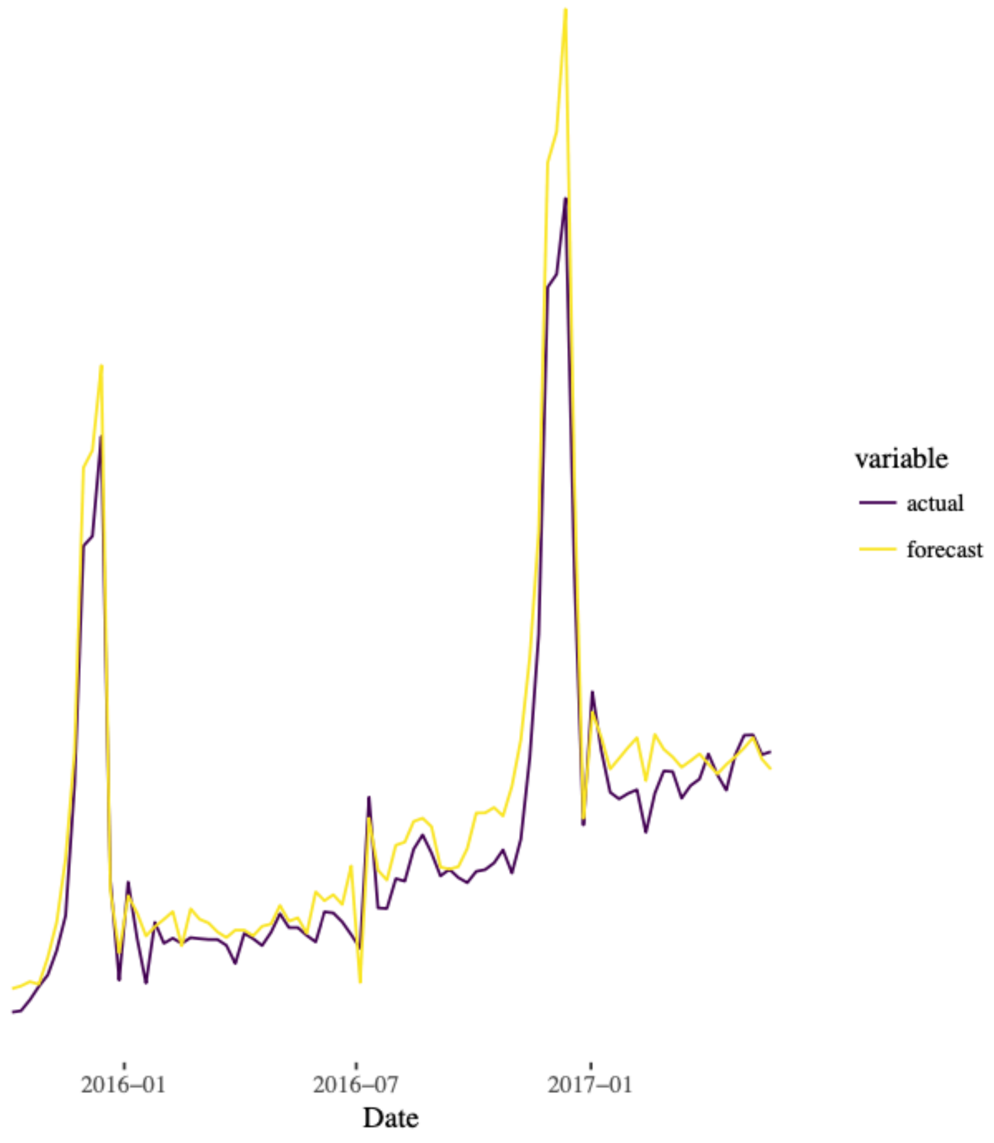
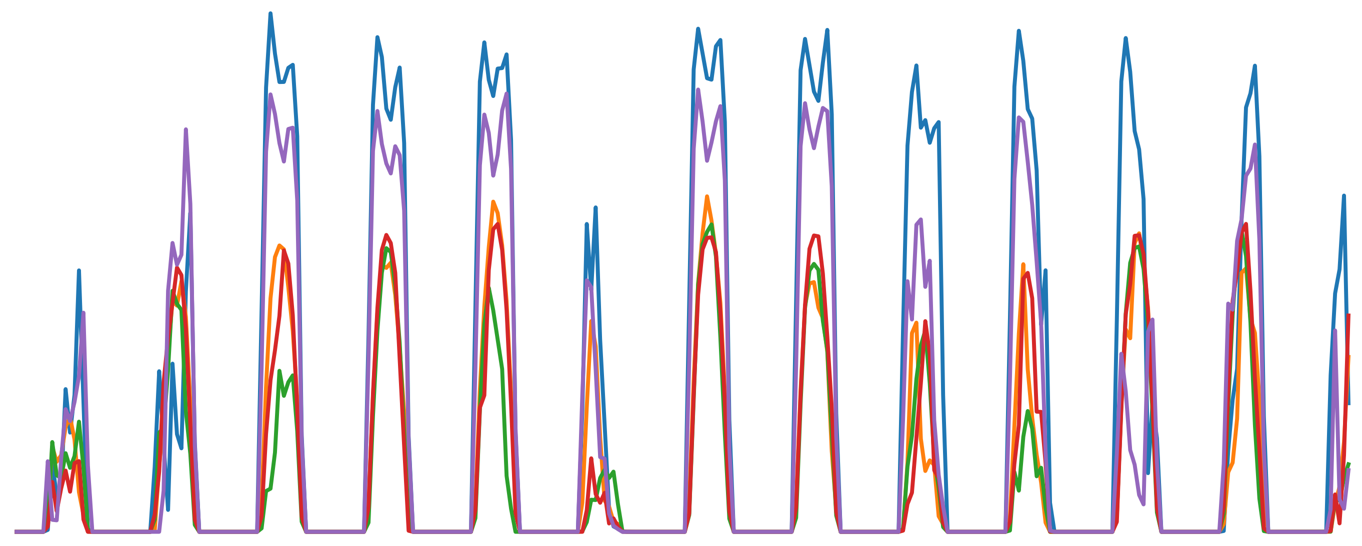
Introduction to Time Series Forecasting

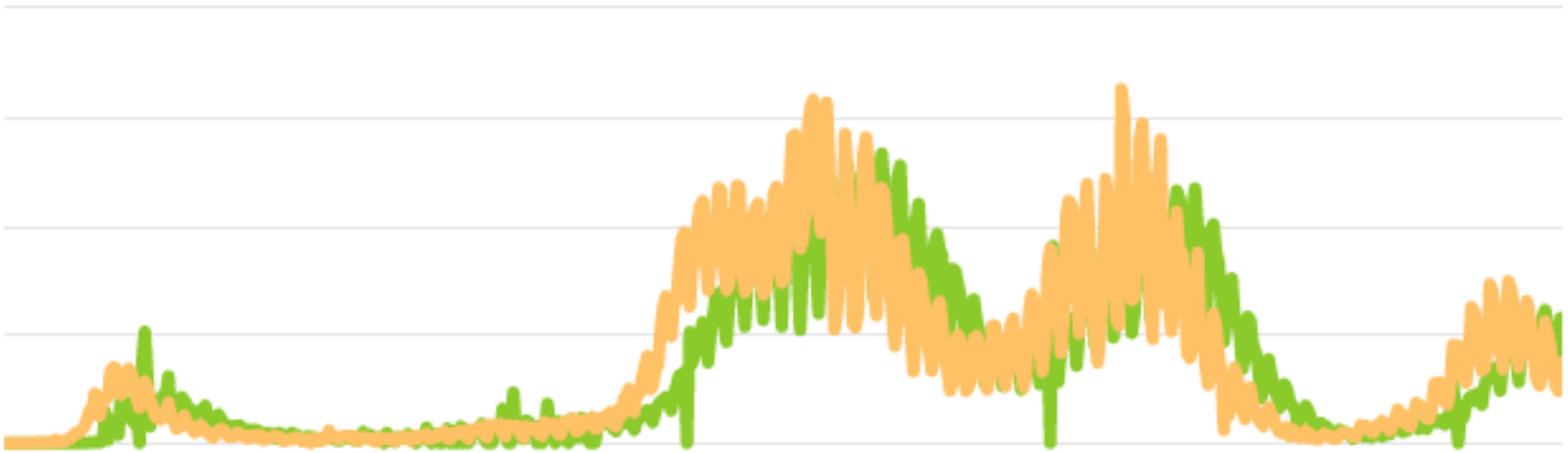
Rajbir-Singh Nirwan

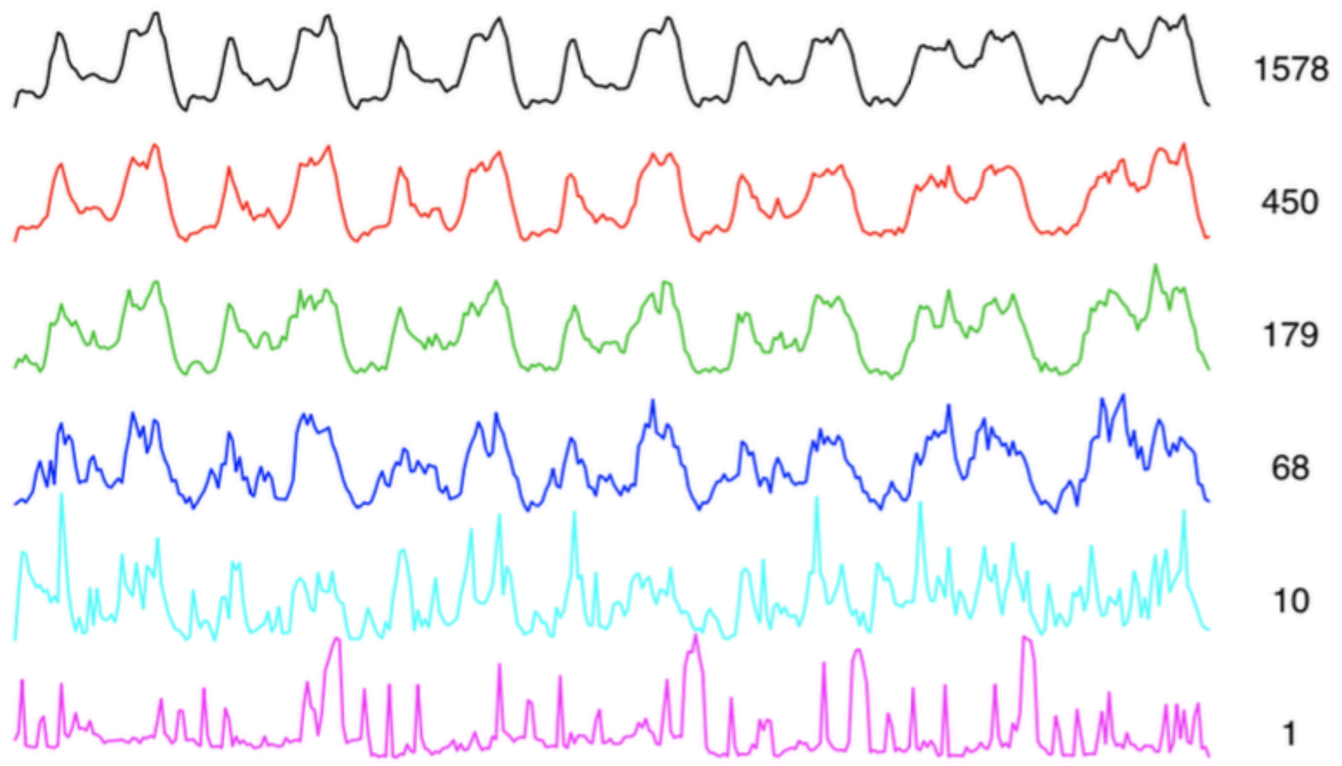
30.09.2021




variable
actual
forecast
2016–01
2016–07
2017–01
Date



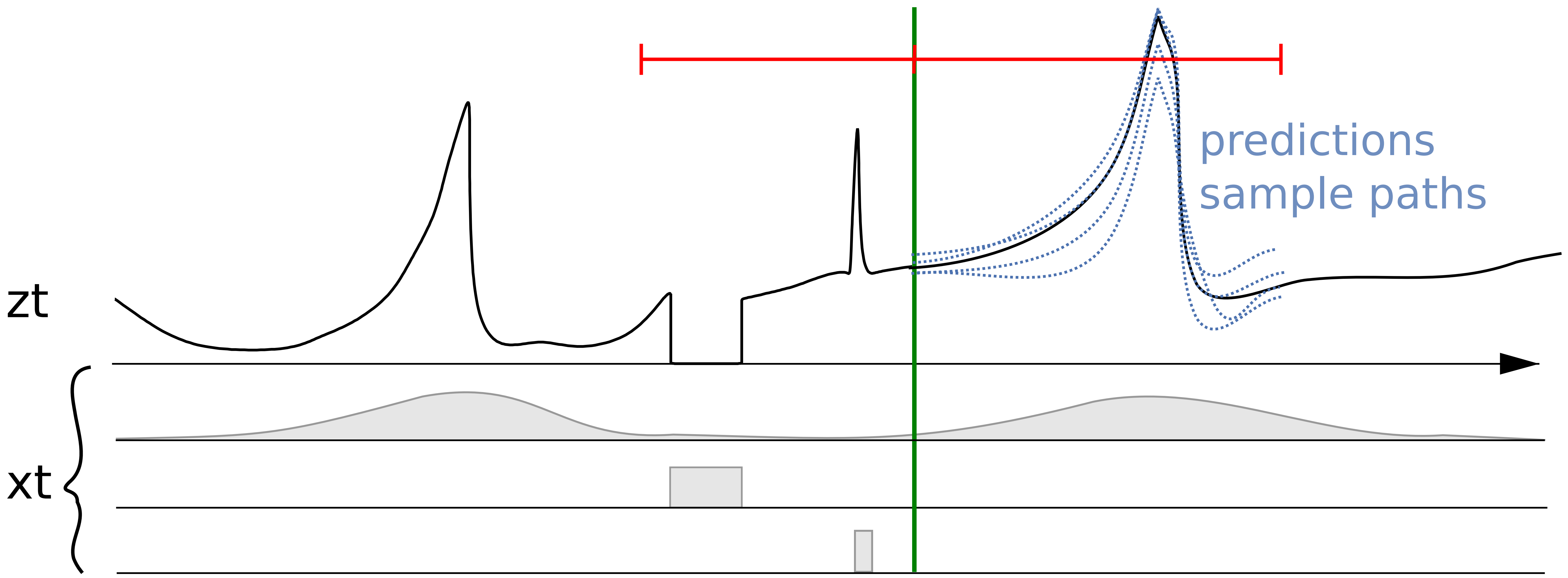



1578
450
179
68
10
1

Outline

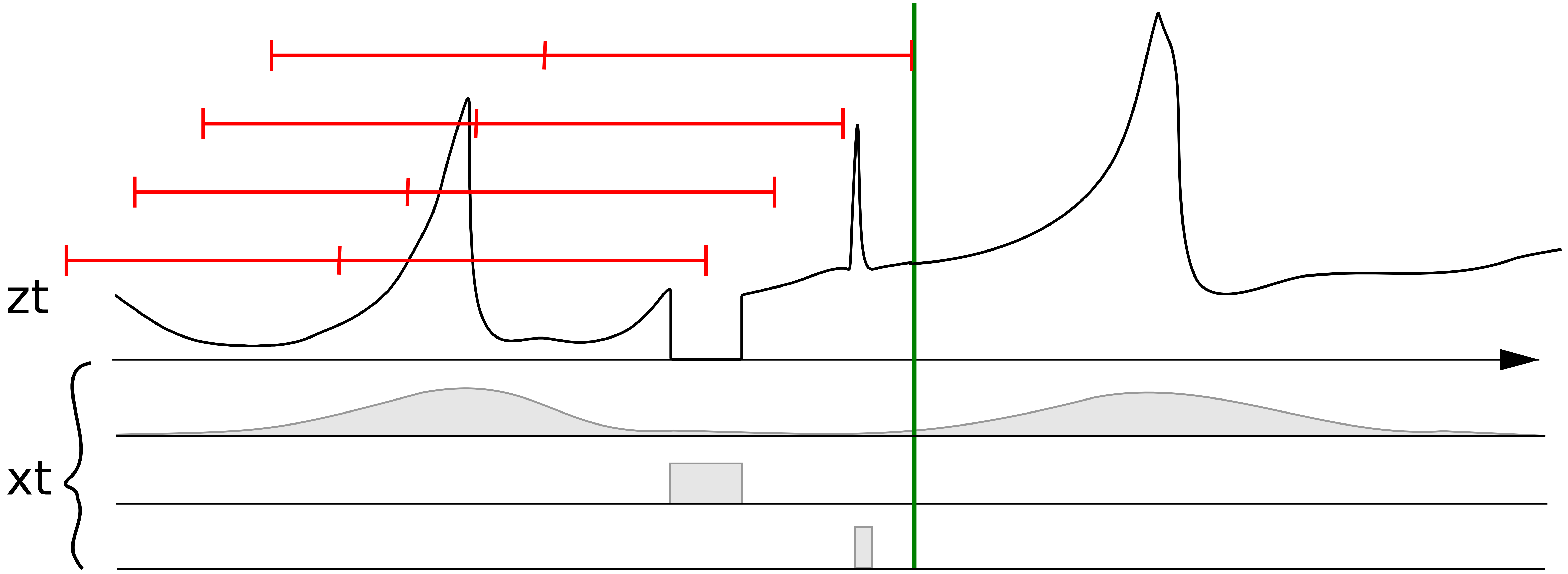
- Introduction
- Simple forecasting models
- Evaluation of forecasts
- Deep learning for time series
- Results

Time Series Forecasting



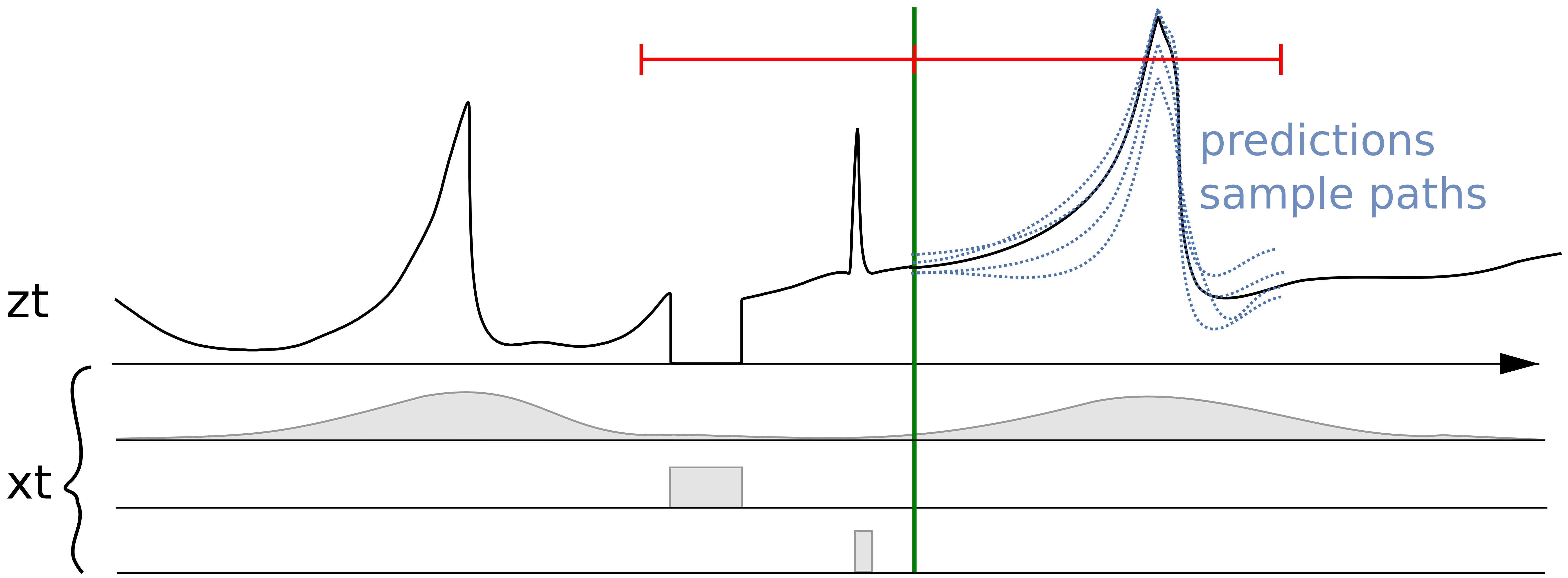
- Given the past we are asked to make predictions for the future
- Sequentially observed data
- Includes seasonality and cycles

Time Series Forecasting



- Given the past we are asked to make predictions for the future
- Sequentially observed data
- Includes seasonality and cycles

Time Series Forecasting



- Given the past we are asked to make predictions for the future
- Sequentially observed data
- Includes seasonality and cycles

Simple TS forecasting models

- Naive model $\tilde{y}_t = y_{t-1}$
- Naive seasonal model $\tilde{y}_t = y_{t-h}$
- Linear Model $y_t = wx_t + \epsilon_t$
- Moving average MA(Q) $y_t = \mu + \epsilon_t + \sum_{q=1}^{Q} \theta_q \epsilon_{t-q}$
- Autoregressive AR(P) $y_t = c + \epsilon_t + \sum_{p=1}^{P} \phi_p y_{t-p}$
- ARMA(P, Q) $y_t = c + \epsilon_t + \sum_{p=1}^{P} \phi_p y_{t-p} + \sum_{q=1}^{Q} \theta_q \epsilon_{t-q}$

Evaluation

- Mean square error (MSE) $e_t = |y_t - \tilde{y}_t|^2$
- Mean absolute deviation (MAD) $e_t = |y_t - \tilde{y}_t|$
- Mean absolute percentage (MAPE) $e_t = |y_t - \tilde{y}_t| / |y_t|$
- Symmetric MAPE (sMAPE) $e_t = |y_t - \tilde{y}_t| / (|y_t| + |\tilde{y}_t|)$
- Maximum likelihood (ML) $e_t = -\log p(y_t | \tilde{y}_t, \theta)$

Evaluation

- Probabilistic Forecasts
  - Instead of minimising MSE, MAD, MAPE, …
  - Minimise negative log-likelihood of data given a distribution

    Gaussian

    Poisson, Negative Binomial (Count data)

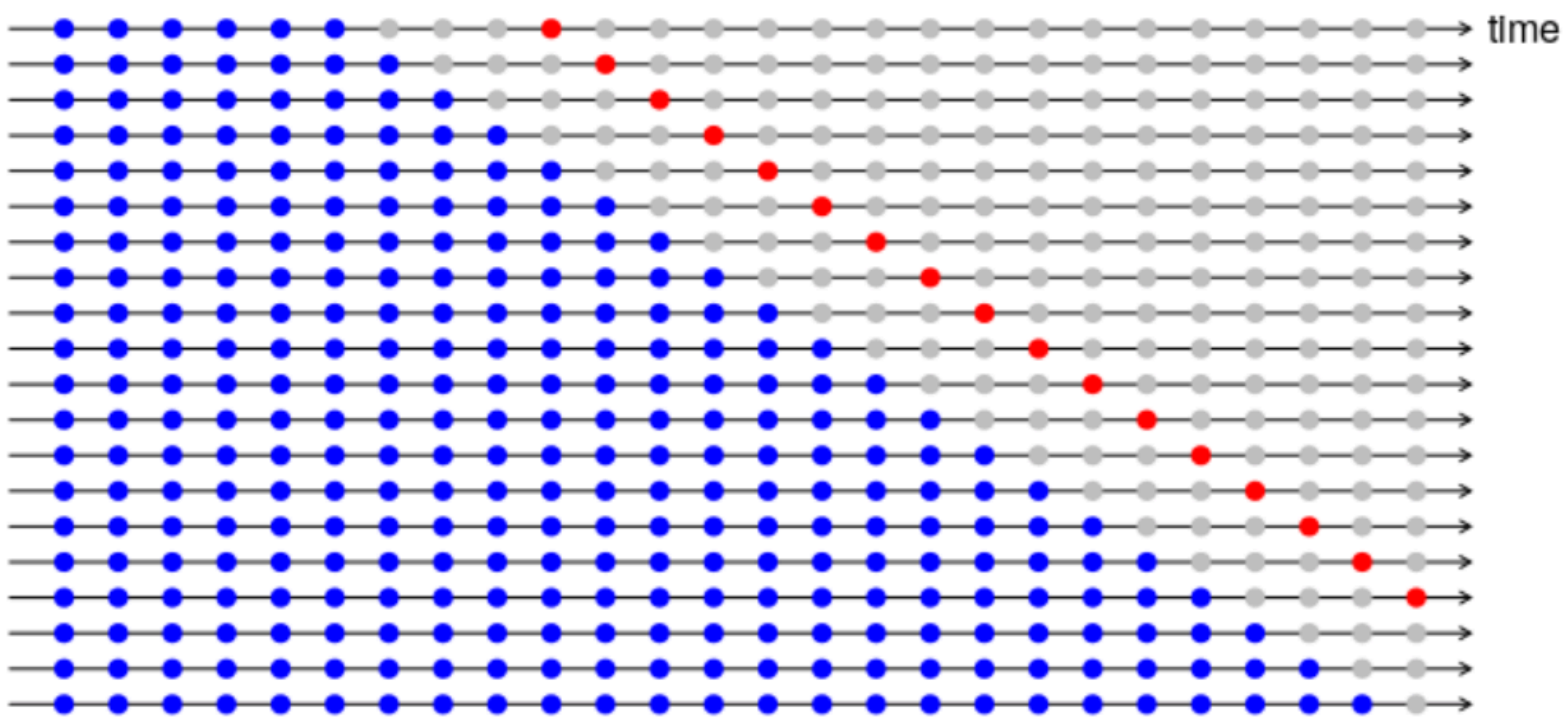
    Beta ([0,1])

    Bernoulli (Binary data)

    Student-t (Heavy tailed data)

Evaluation of time series data

- Cross validation
  - Roll out training and testing data forward in time
  - Make sure future information does not leak backwards in time

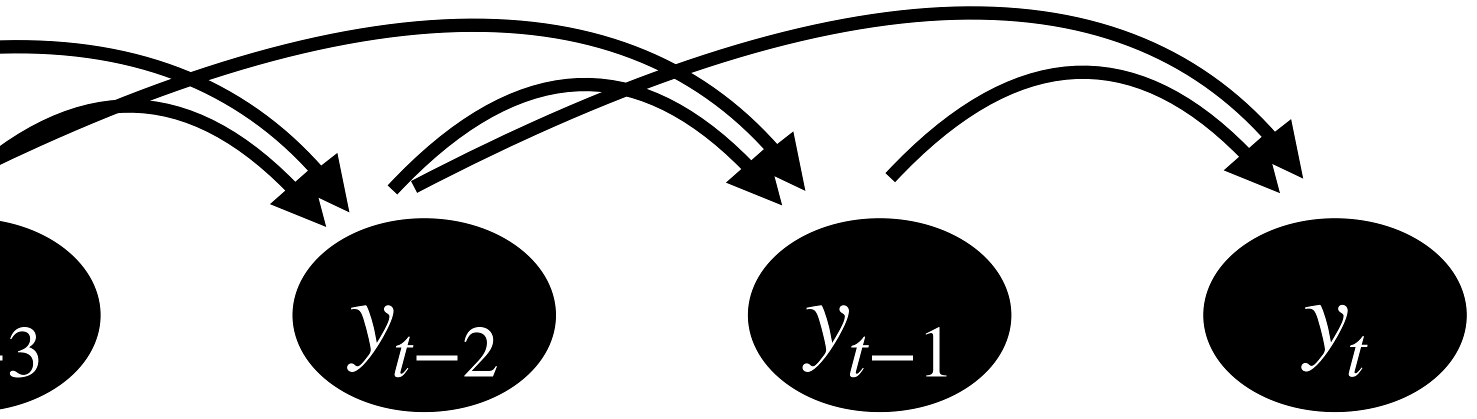


[Hyndman and Athanasopoulos, 2017]

Capturing sequential relationship of TS

- Autoregressive

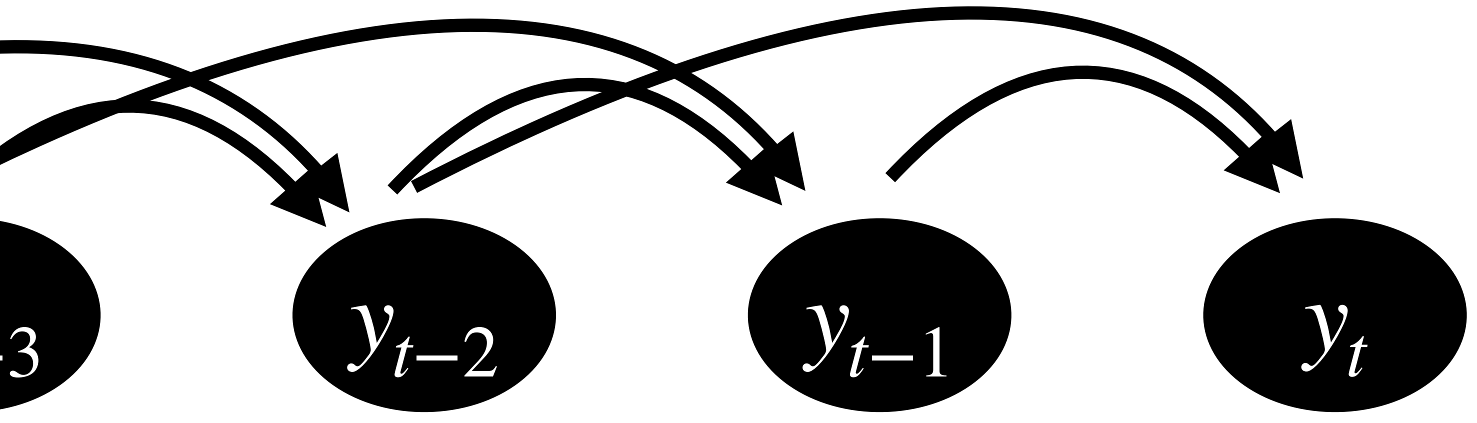
$$y_t = \sum_{n=1}^{N} \phi_n y_{t-n}$$



Capturing sequential relationship of TS

- Autoregressive

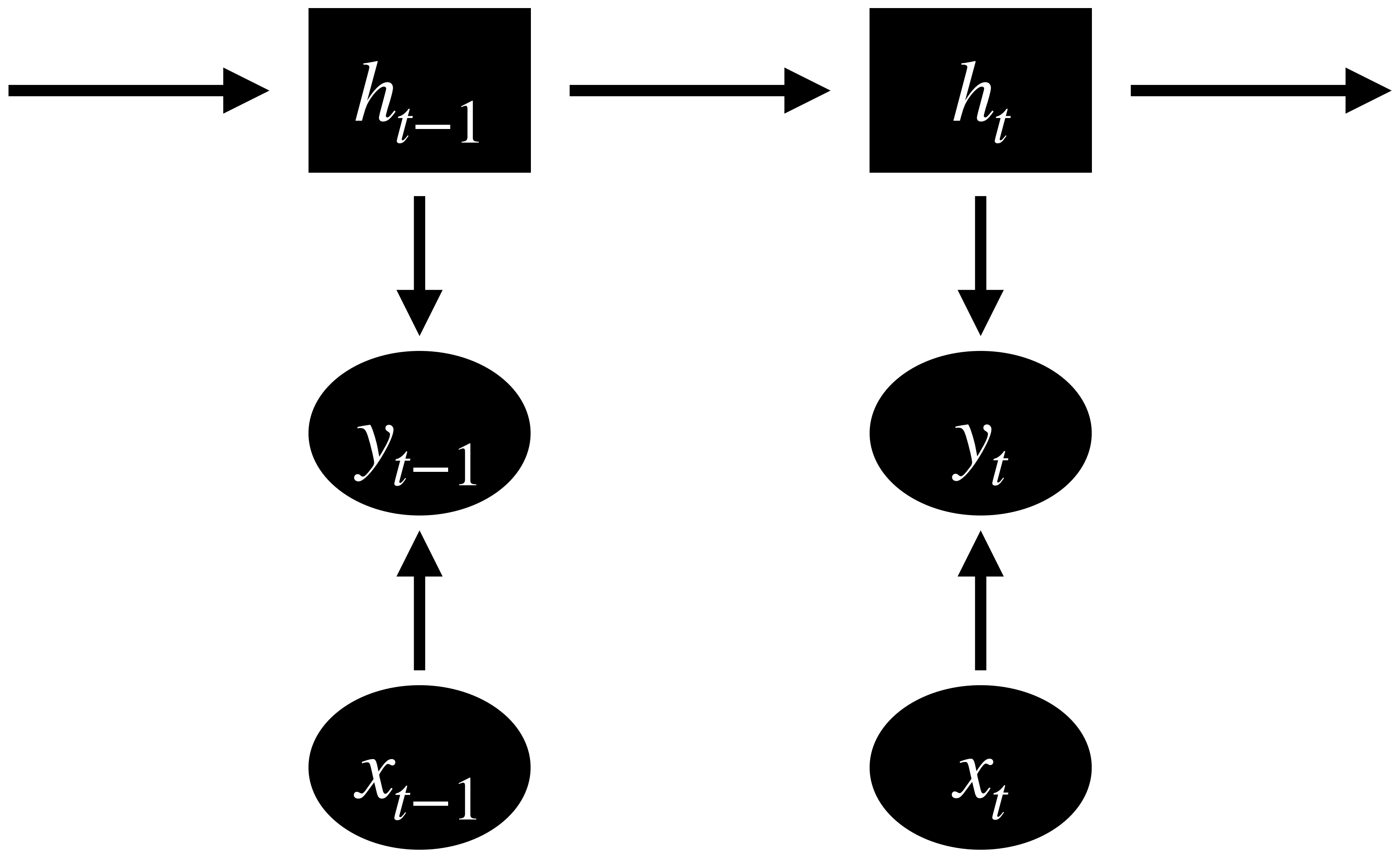
$$y_t = \sum_{n=1}^{N} \phi_n y_{t-n}$$



- State-space

$$h_t = Ah_{t-1} + \epsilon_t$$

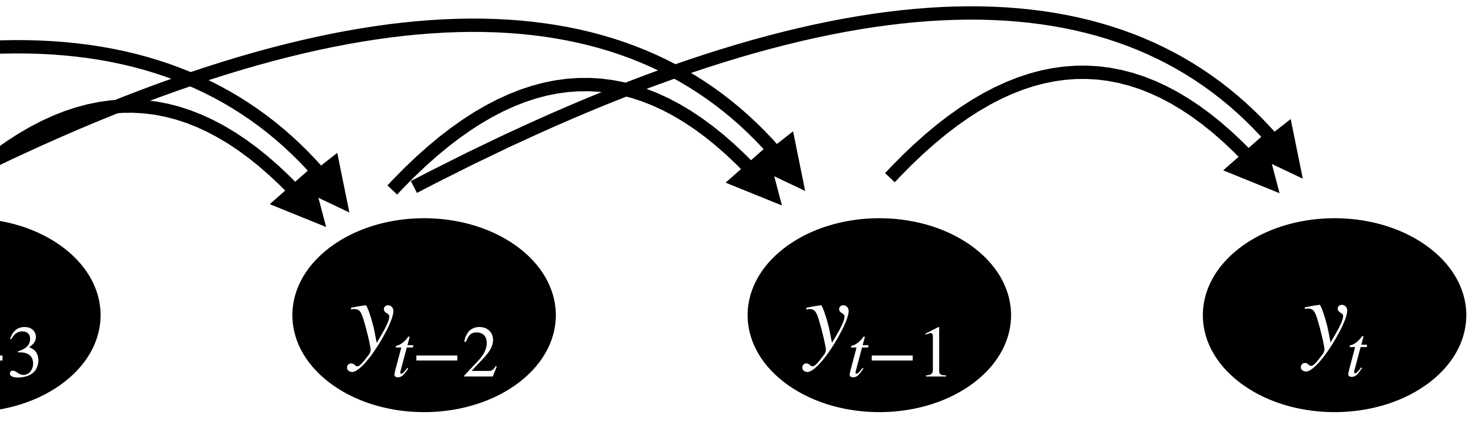
$$y_t = Bh_t + Wx_t + \epsilon_t$$



Capturing sequential relationship of TS

- Autoregressive

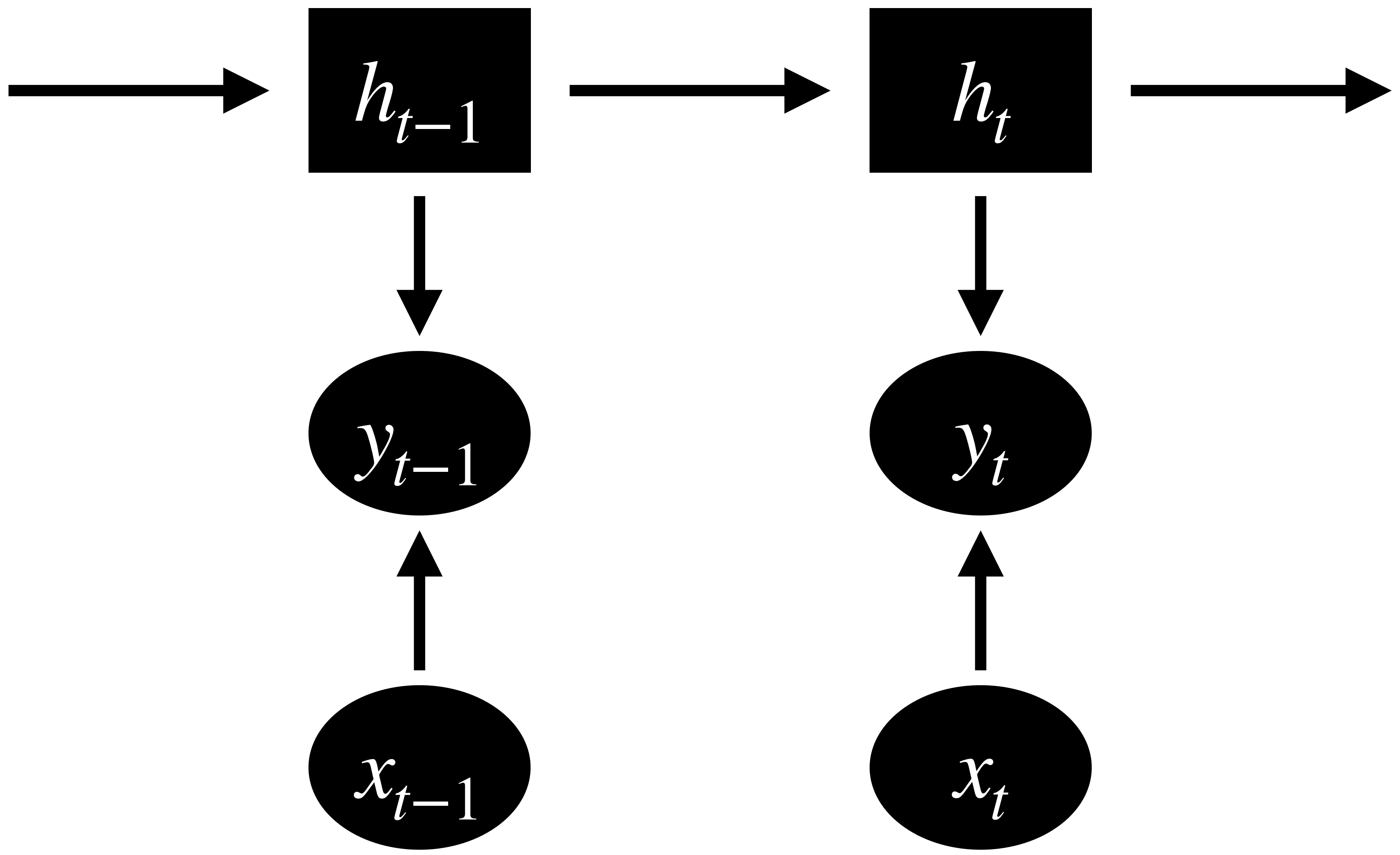
$$y_t = \sum_{n=1}^{N} \phi_n y_{t-n}$$



- State-space

$$h_t = Ah_{t-1} + \epsilon_t$$

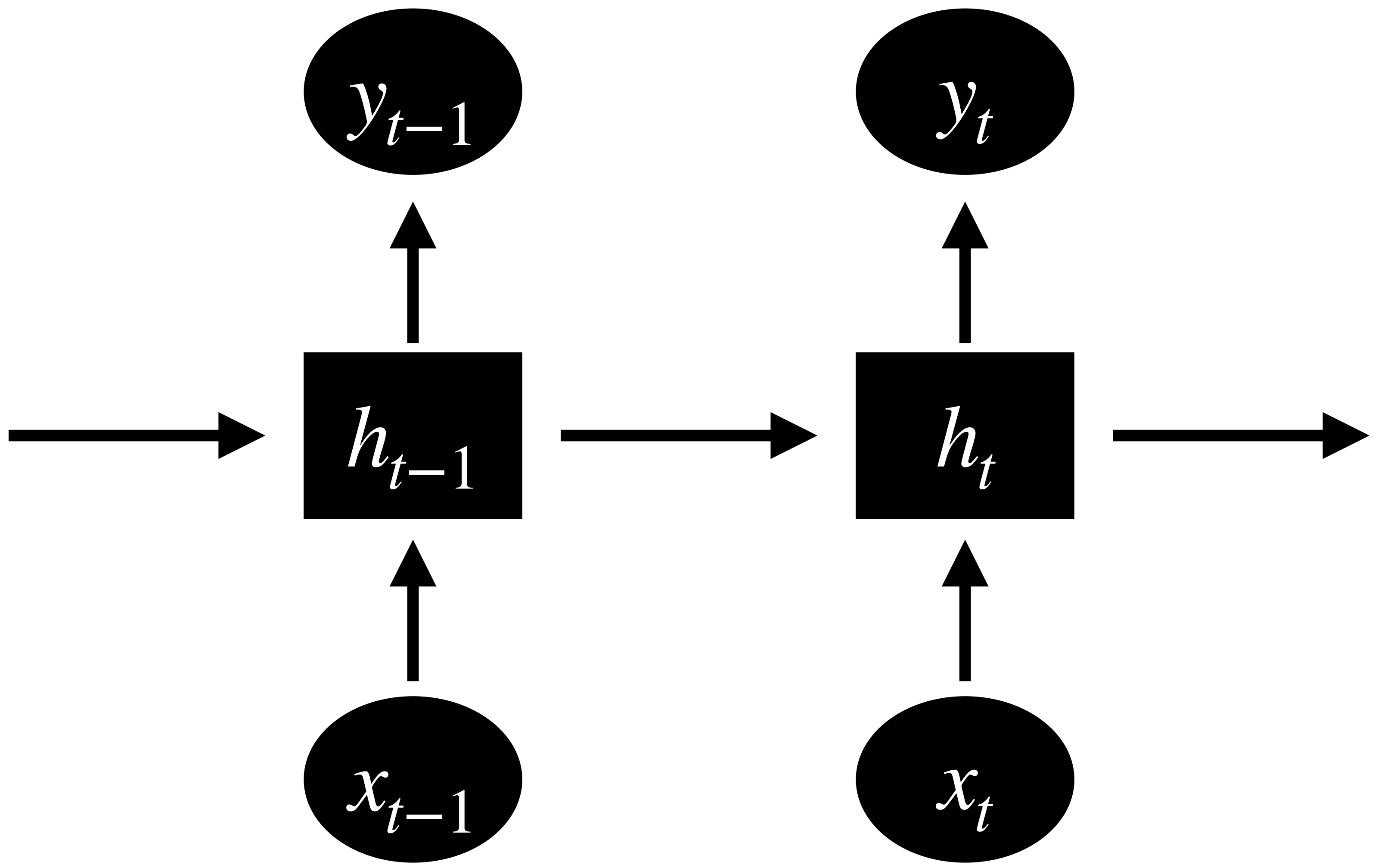
$$y_t = Bh_t + Wx_t + \epsilon_t$$



- NNs

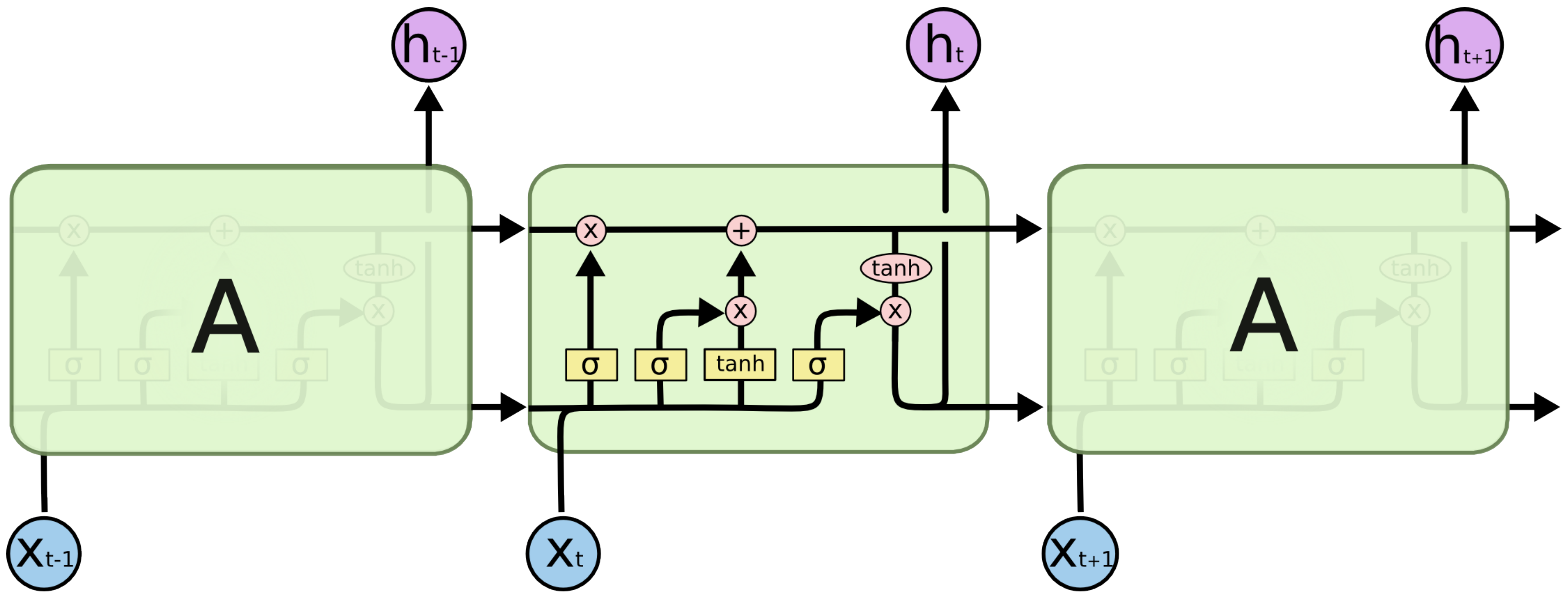
$$h_t = \sigma(\theta_0 h_{t-1} + \theta_1 x_t)$$

$$y_t = \sigma(\theta h_t)$$



Capturing sequential relationship of TS

RNN using LSTM units

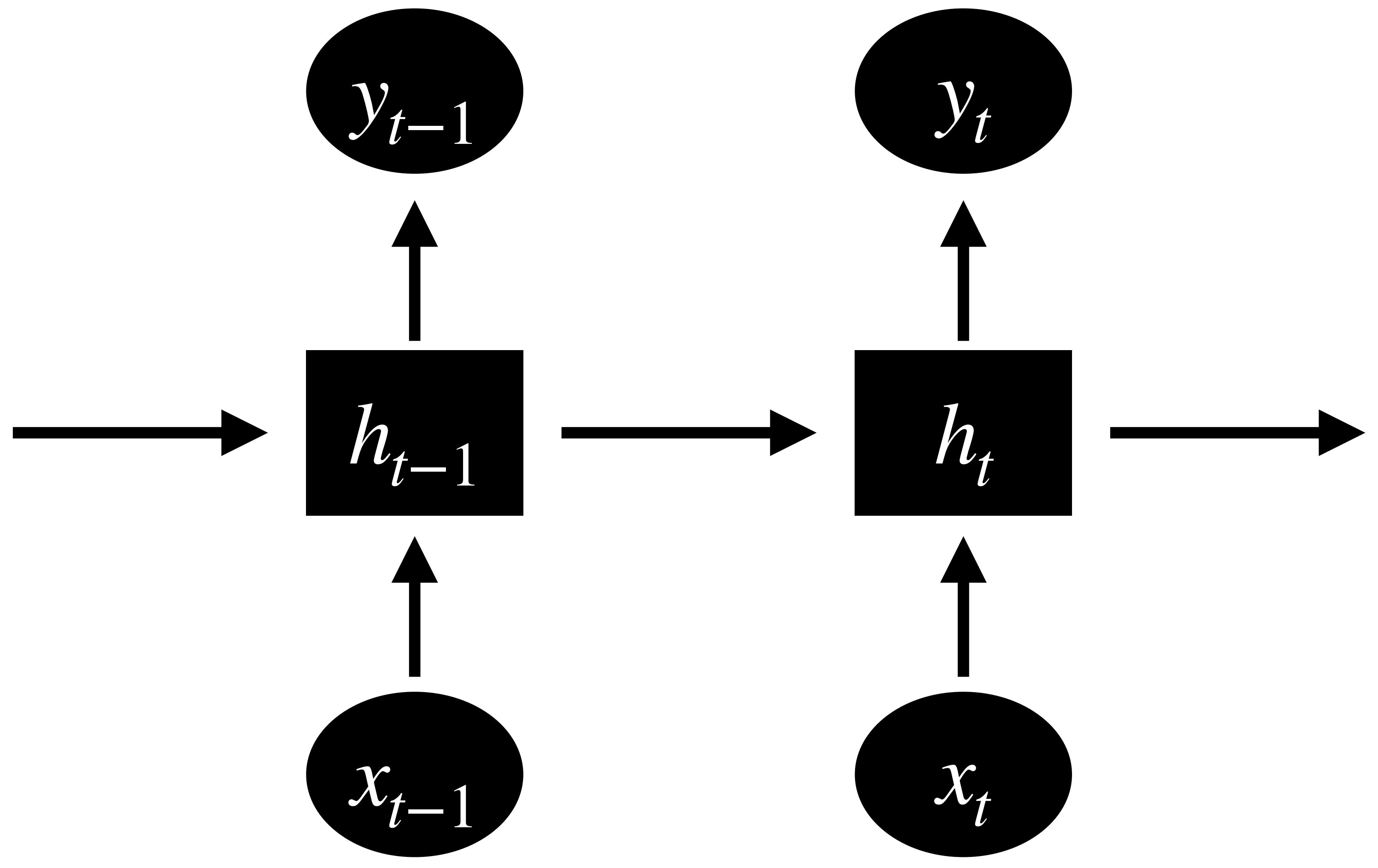


https://colah.github.io/posts/2015-08-Understanding-LSTMs/

- NNs

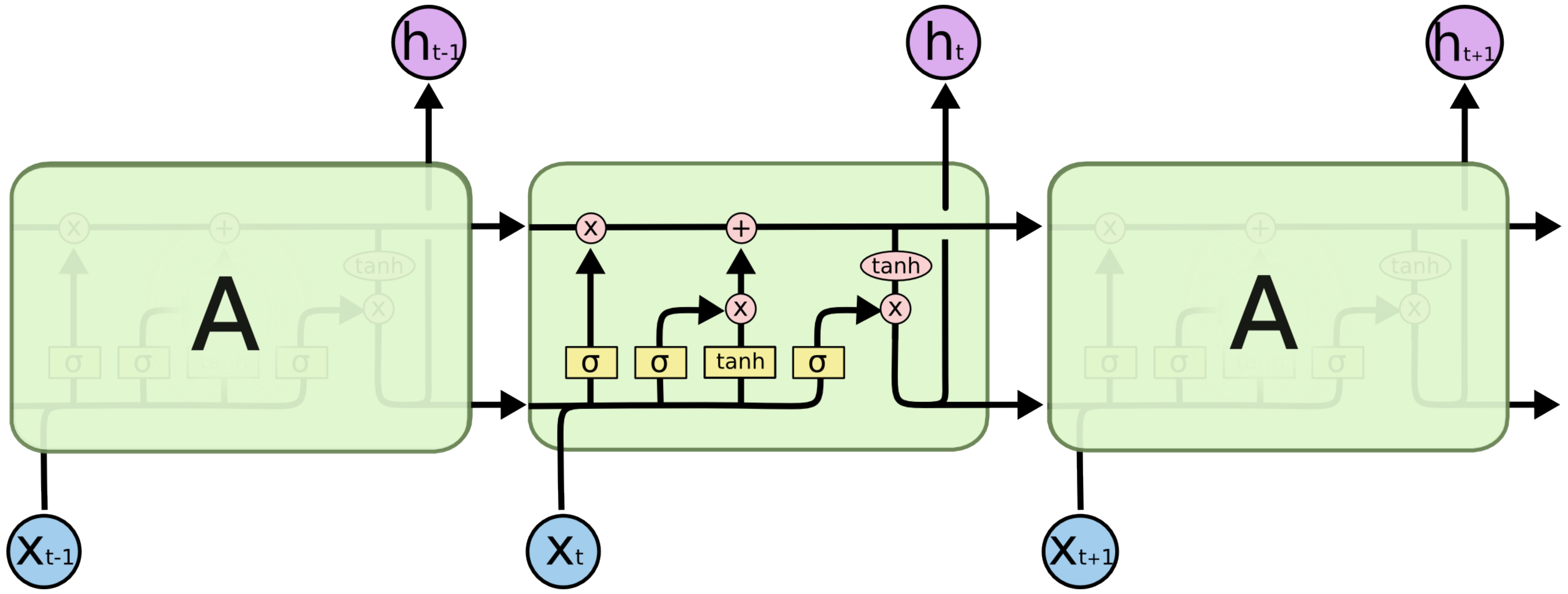
$$h_t = \sigma(\theta_0 h_{t-1} + \theta_1 x_t)$$

$$y_t = \sigma(\theta h_t)$$



Capturing sequential relationship of TS

RNN using LSTM units

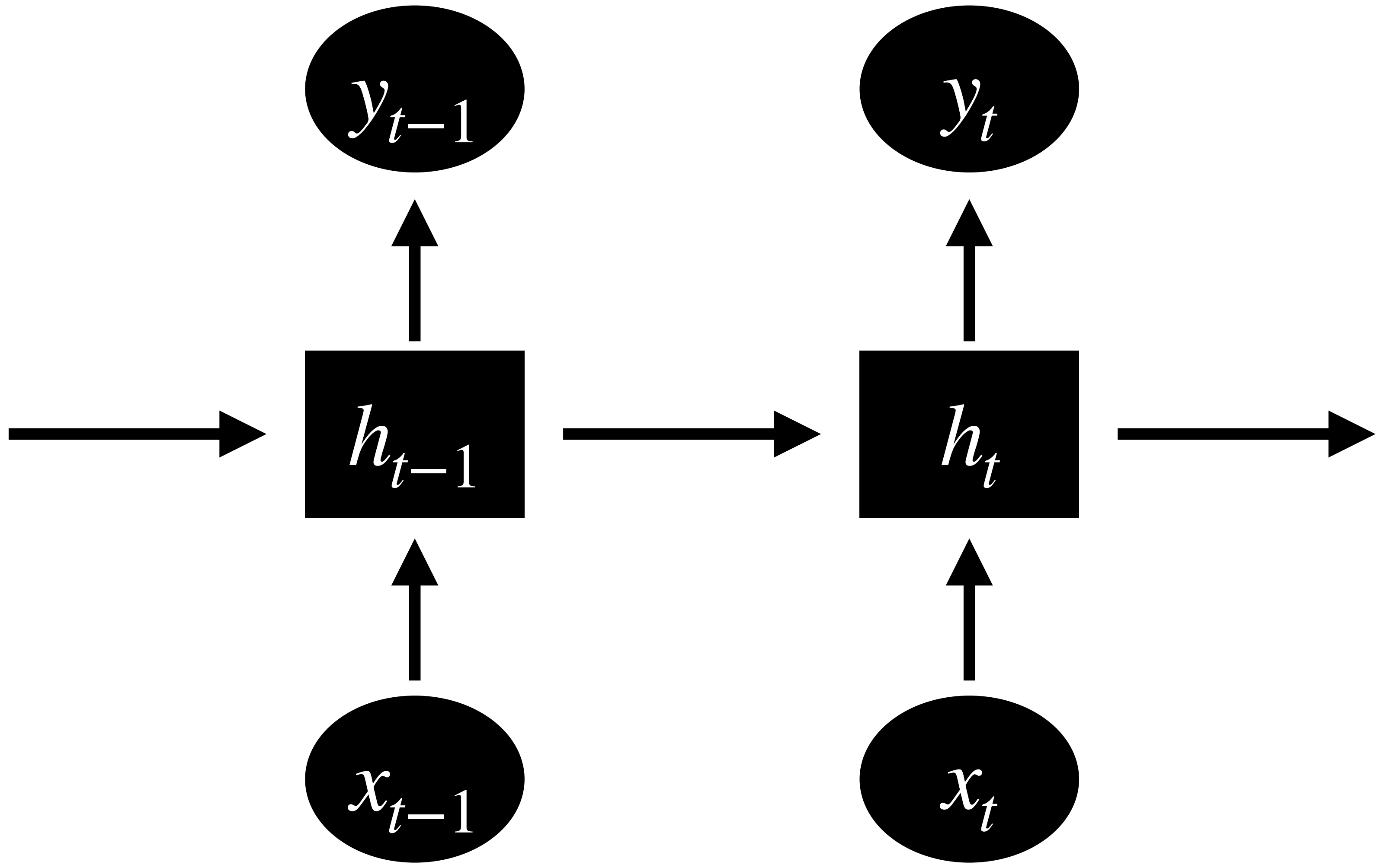


https://colah.github.io/posts/2015-08-Understanding-LSTMs/

- NNs

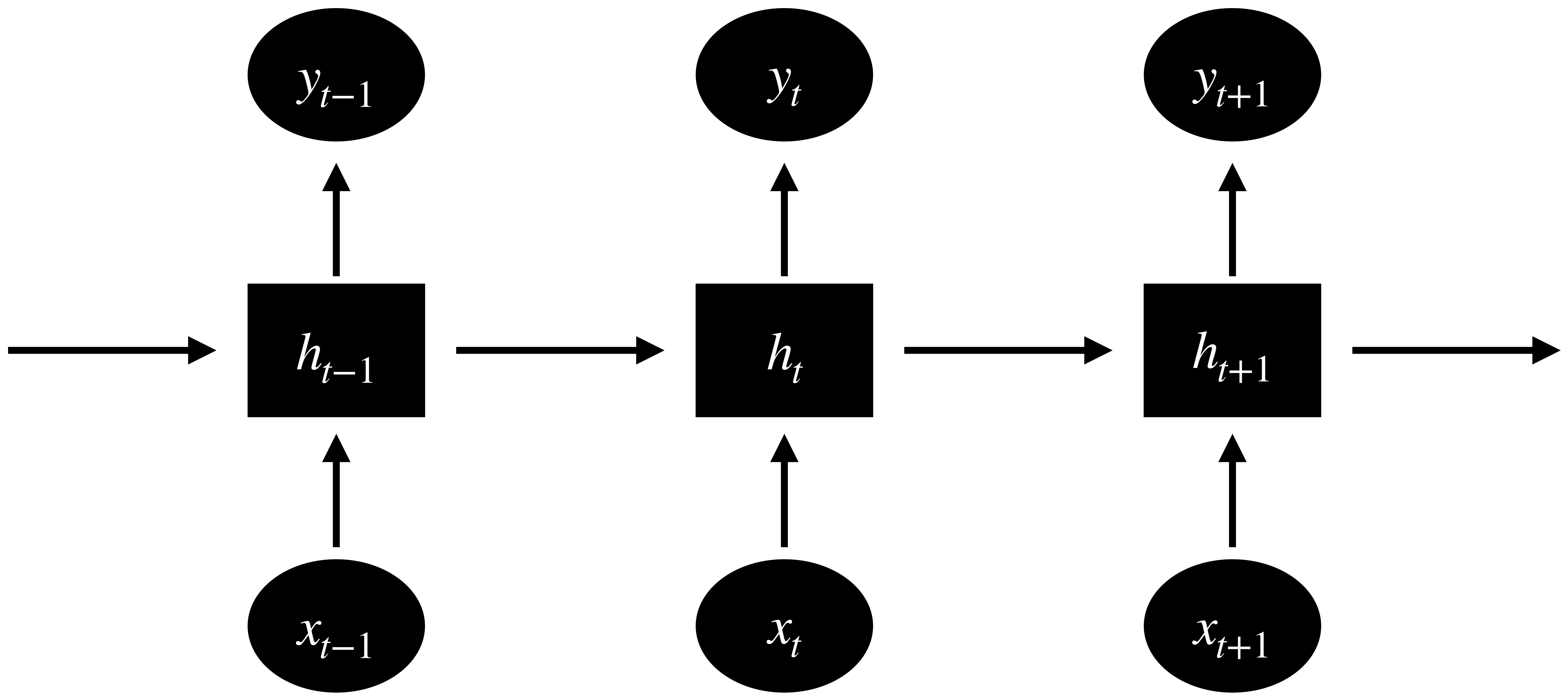
$$h_t = f_1(h_{t-1}, x_t)$$

$$y_t = f_2(h_t)$$



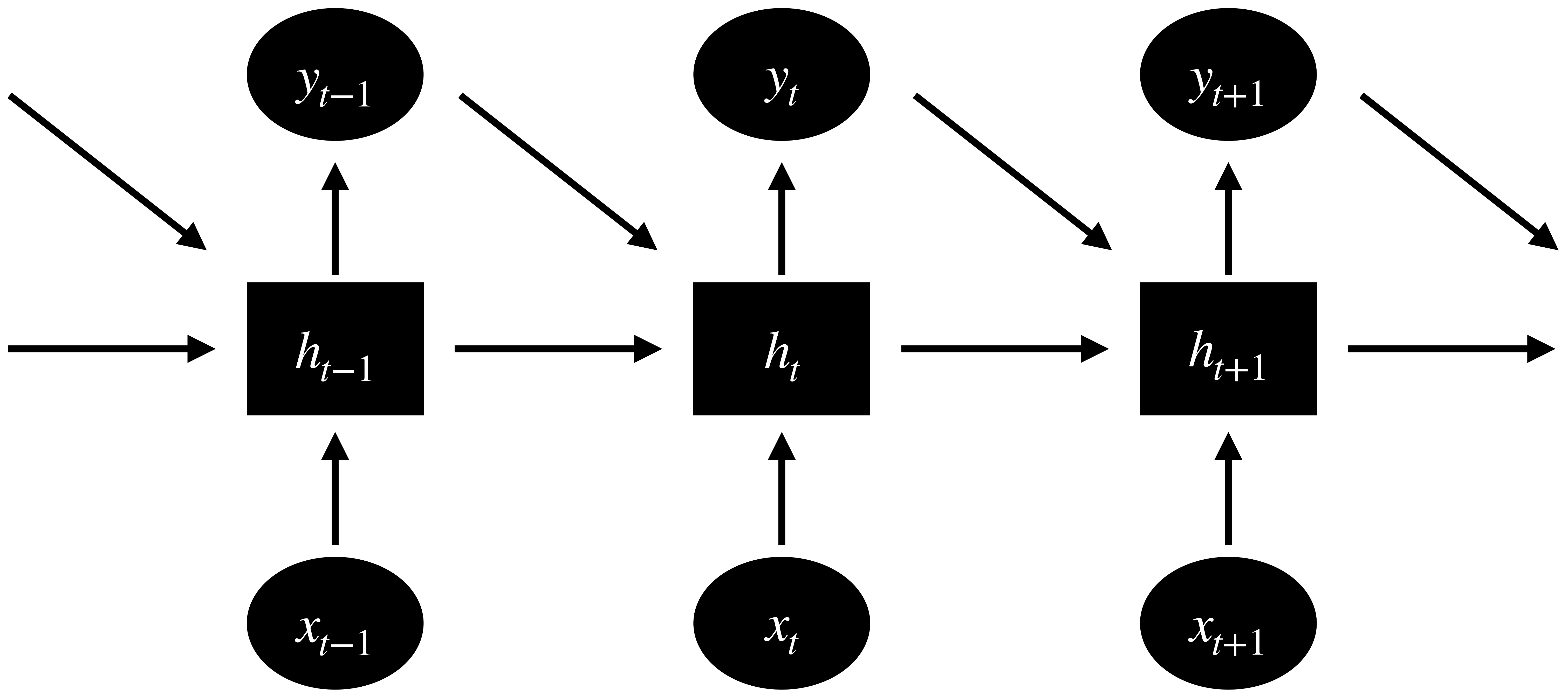
DeepAR [Flunkert et al. 2017]

One-to-One RNN combined with an autoregressive structure



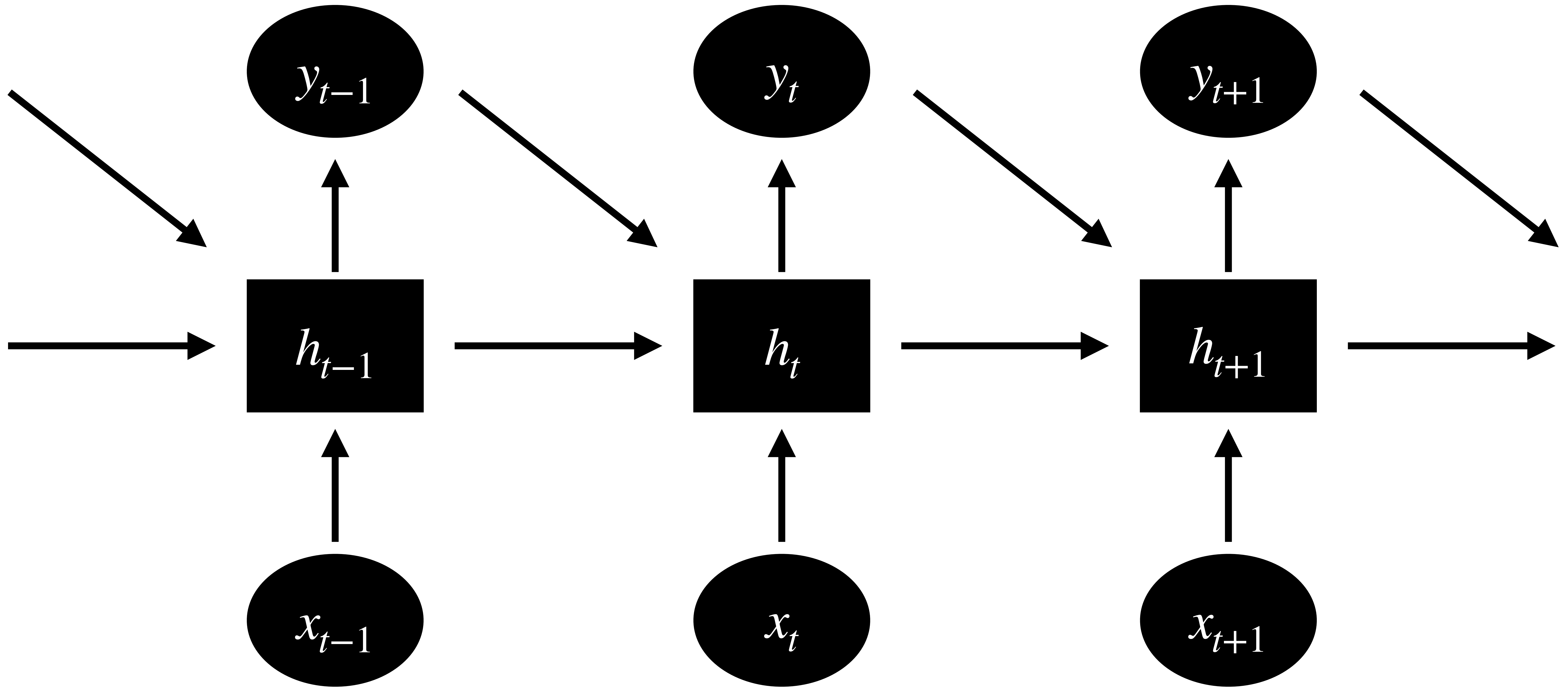
DeepAR [Flunkert et al. 2017]

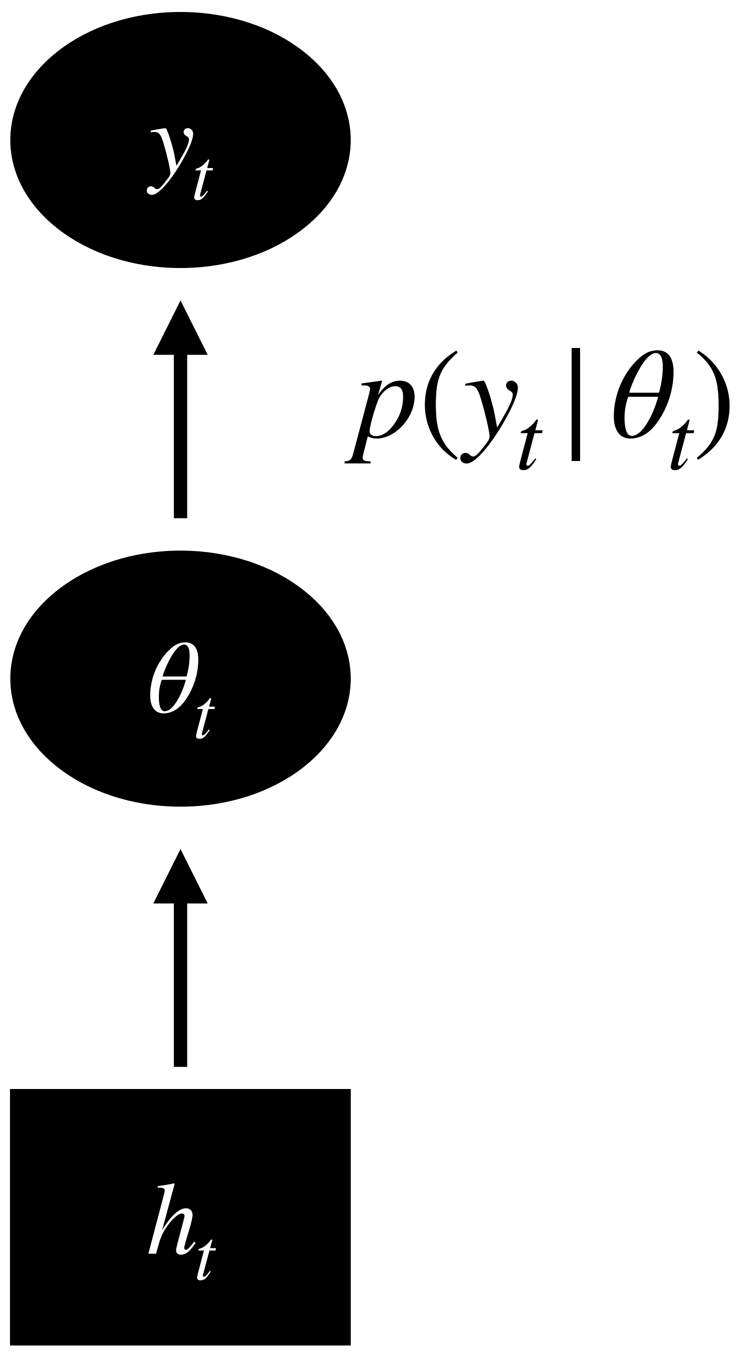
One-to-One RNN combined with an autoregressive structure



DeepAR [Flunkert et al. 2017]

One-to-One RNN combined with an autoregressive structure





Probabilistic Forecasts using DeepAR

- Embedder: $h_t = f_1(h_{t-1}, y_{t-1}, x_t) \quad \rightarrow \quad$ RNN
- Forecaster: $p(y_t \,|\, f_2(h_t))$

$\rightarrow$ e.g.: $f_2(h_t) = (w_\mu^T h_t + b_\mu, \text{softplus}(w_\sigma^T h_t + b_\sigma)) = (\mu, \sigma)$

Probabilistic Forecasts using DeepAR

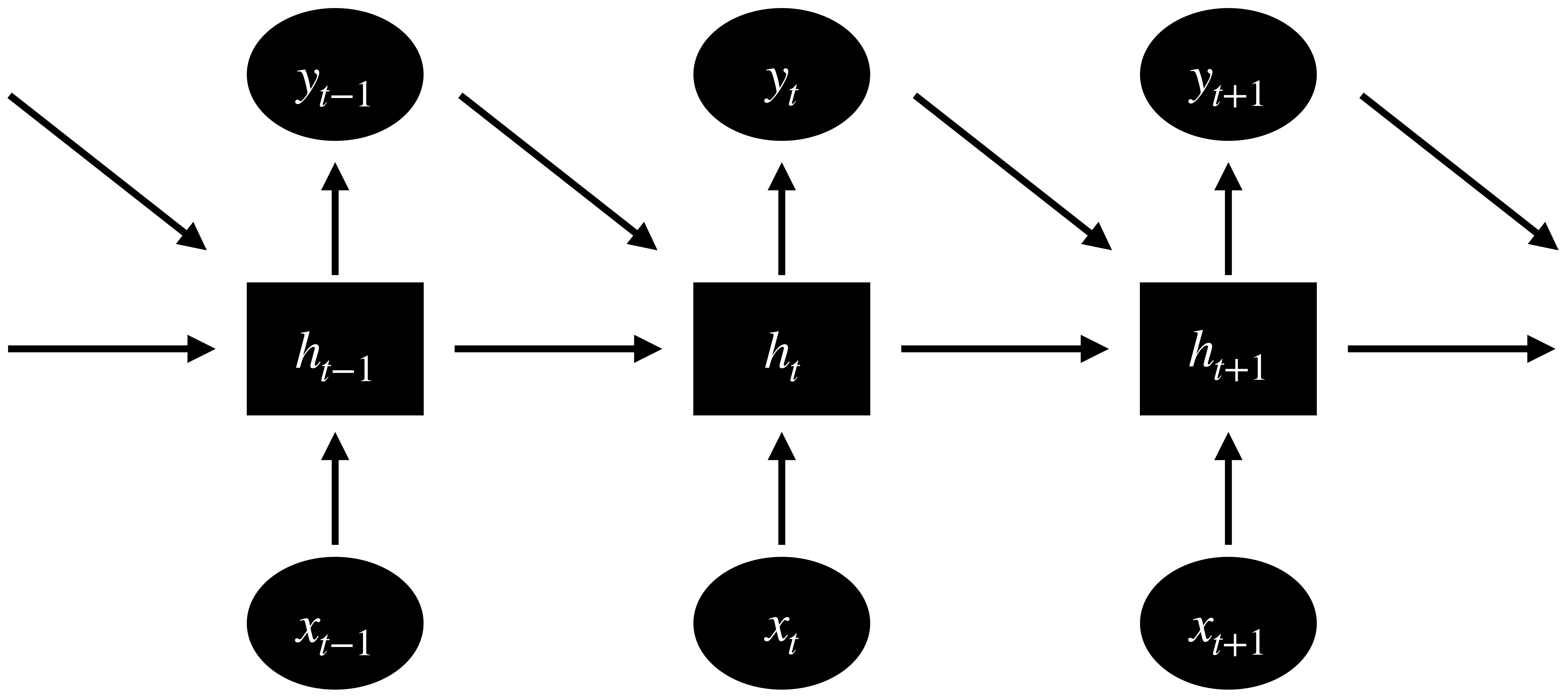
- Embedder: $h_t = f_1(h_{t-1}, y_{t-1}, x_t) \quad \rightarrow \quad$ RNN
- Forecaster: $p(y_t \,|\, f_2(h_t))$

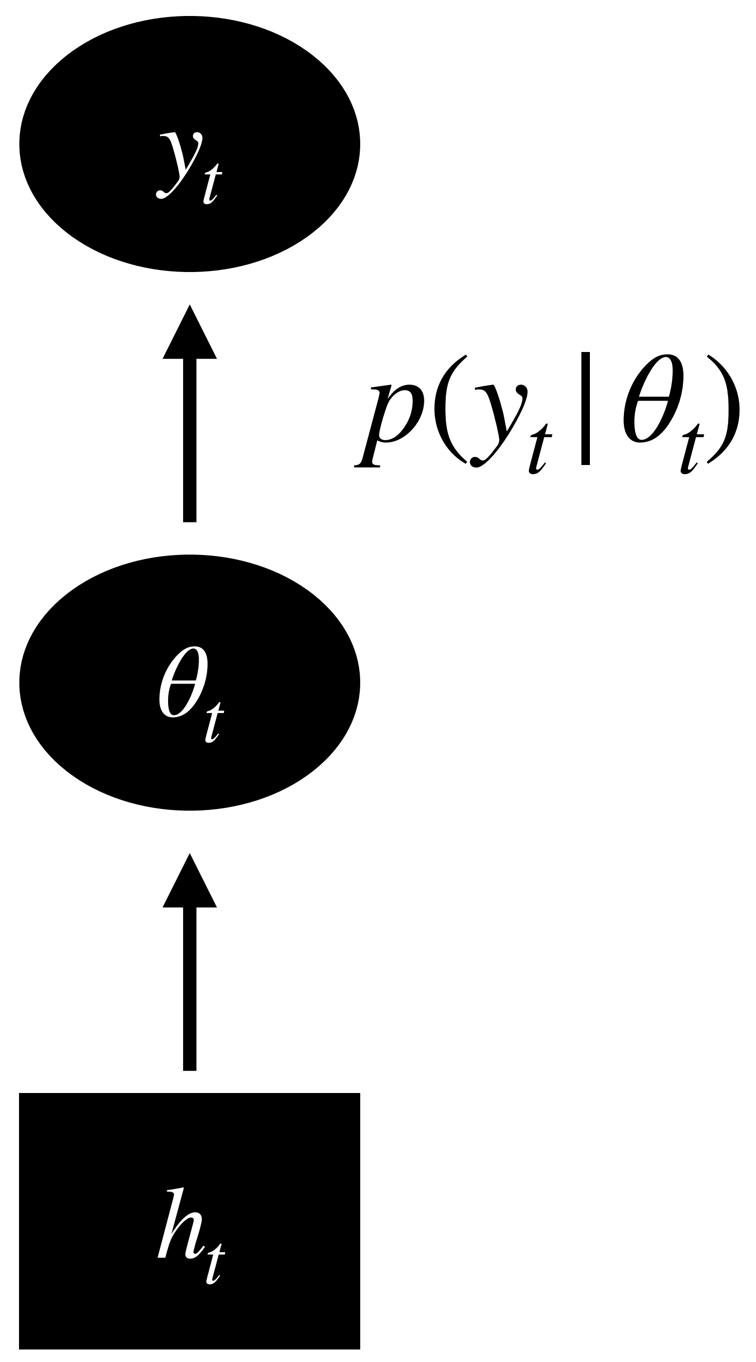
$\rightarrow$ e.g.: $f_2(h_t) = (w_\mu^T h_t + b_\mu, \text{softplus}(w_\sigma^T h_t + b_\sigma)) = (\mu, \sigma)$

Objective: $L = -\sum_{n=1}^{N} \sum_{t=1}^{T} \log p(y_{nt} \,|\, f_2(h_{nt}))$

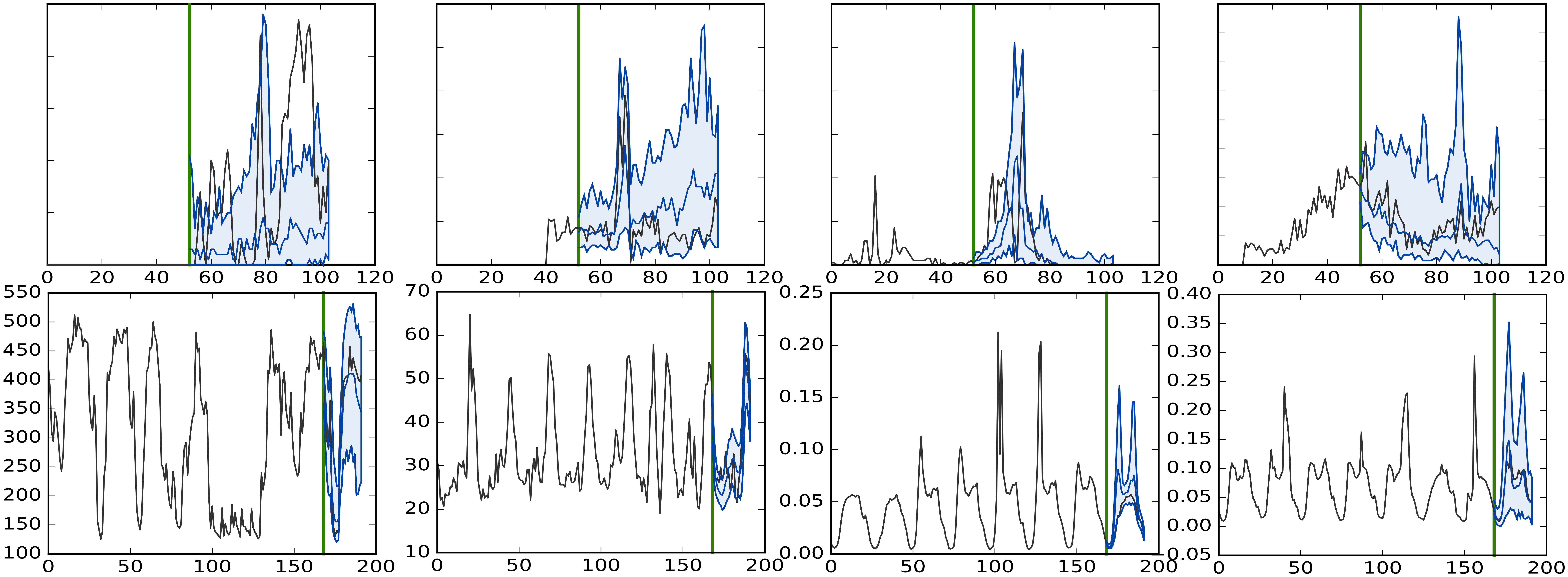
DeepAR [Flunkert et al. 2017]

One-to-One RNN combined with an autoregressive structure

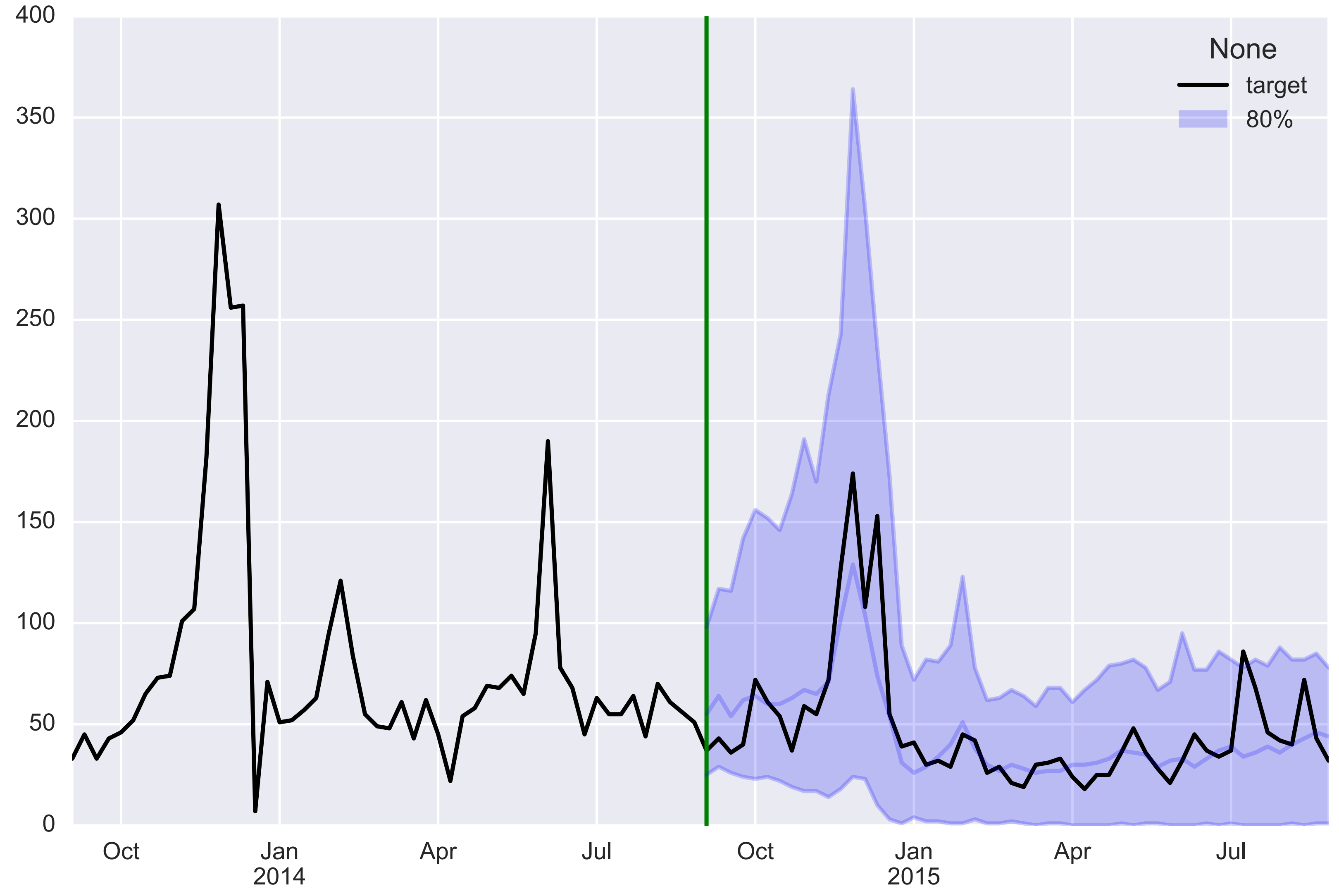


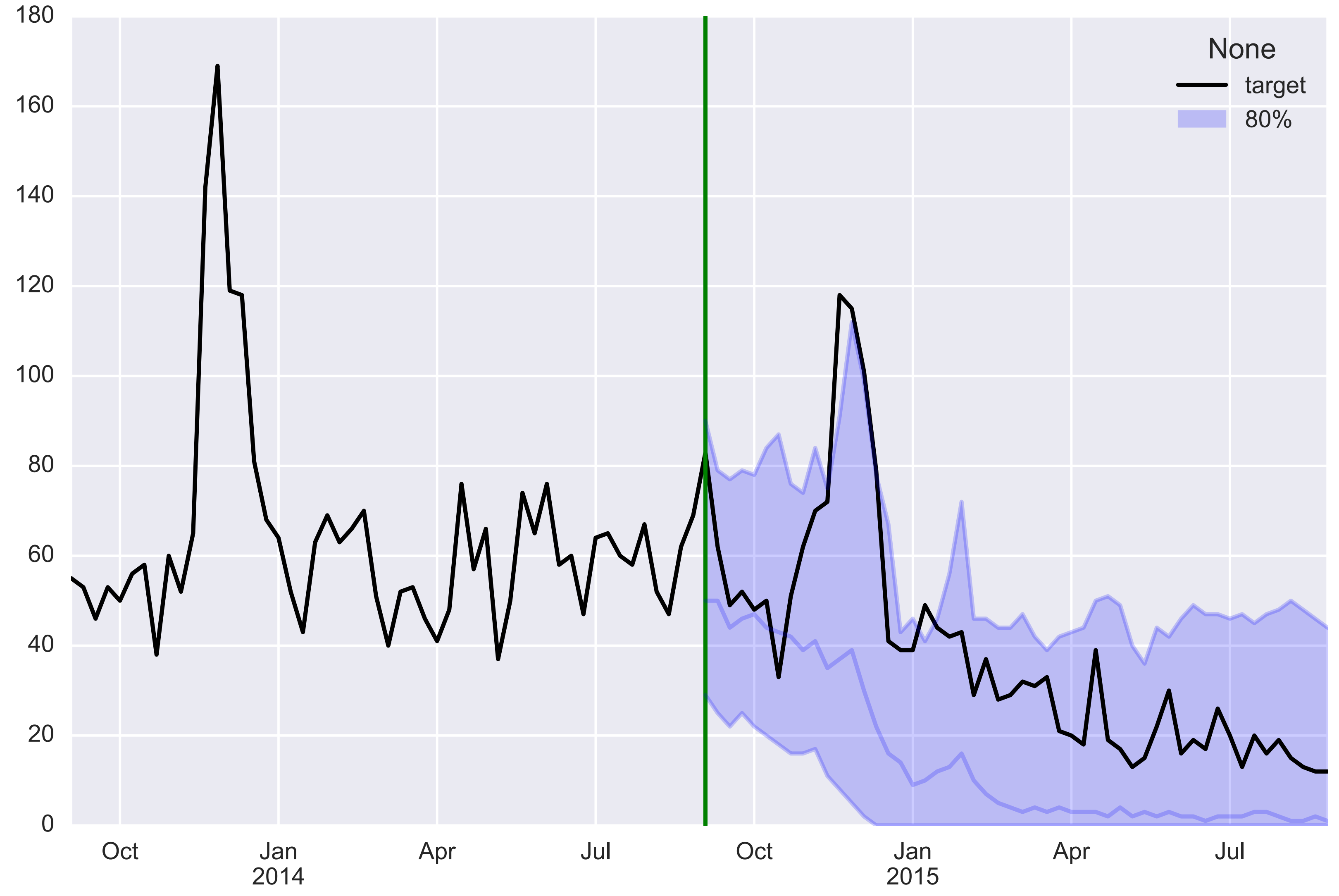


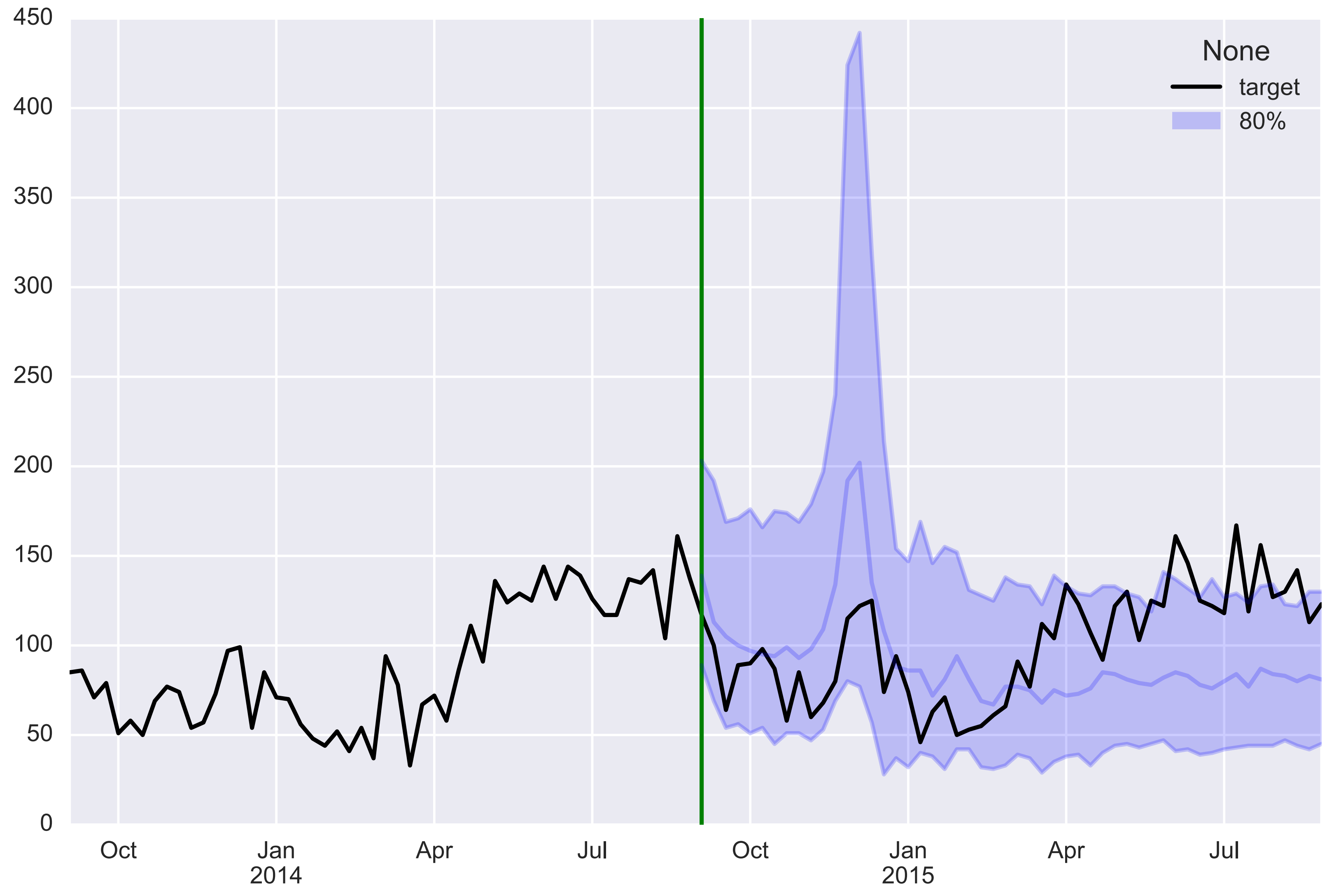
Probabilistic predictions on real world data

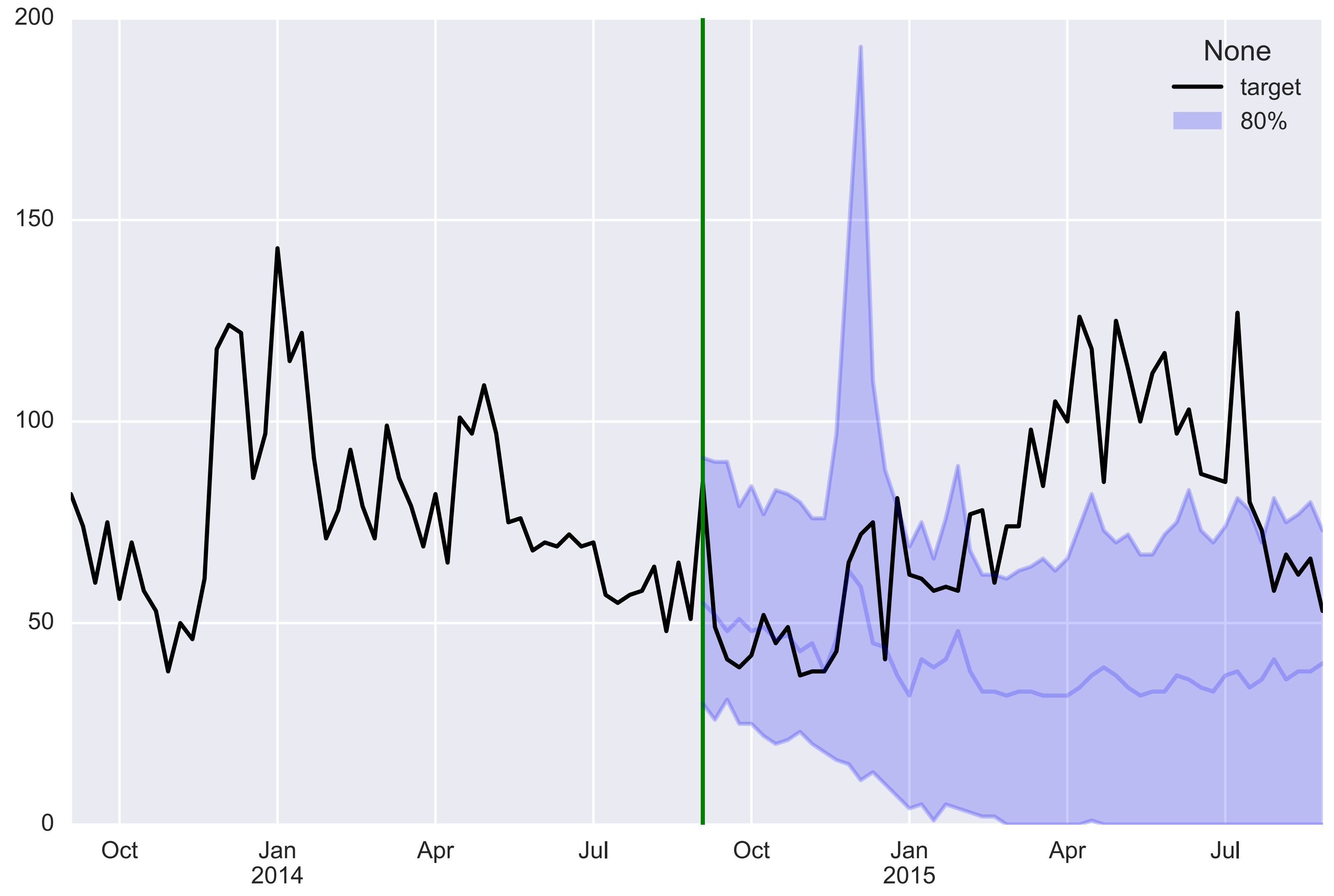


Probabilistic predictions on real world data









GluonTS - python package for TS analysis

https://github.com/awslabs/gluon-ts

Thanks for you attention!