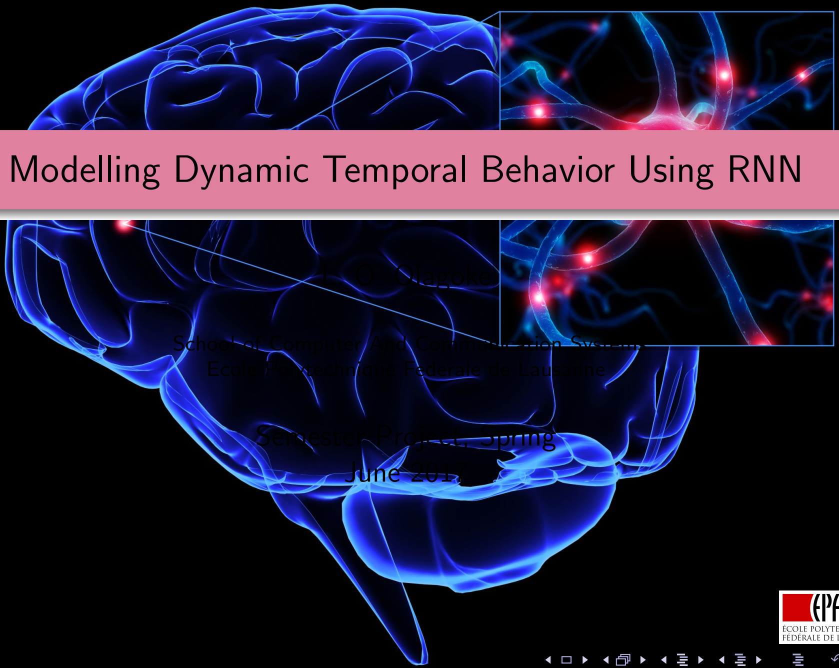


Modelling Dynamic Temporal Behavior Using RNN

L. O. Olagoke

School of Computer And Communication Systems
Ecole Polytechnique Federale de Lausanne

Semester Project, Spring
June 2017

Table of Contents

Motivation

We have got to pause and ask ourselves: How much clean air do we need?

Lee Lacocca,CEO/Chairman, Chrysler Corporation, 1979-1992

- London Disaster, 1952.
- Beijing smog: pollution red alert declared in China capital and 21 other cities, 16 Dec 2016[1]
- Air Pollution Grips Macedonian Capital, 7 Feb , 2017[2]
- Health Issue: asthma, bronchitis and chronic obstructive pulmonary disease (COPD).
- How do we model the dynamics of air pollutant ?

[1] https://www.theguardian.com/world/2016/dec/17/beijing-smog-pollution-red-alert-declared-in-china-capital-and-21-other-cities

[2] http://www.balkaninsight.com/en/article/air-pollution-grips-macedonian-capital-02-07-2017

Air pollution and Time Series

- Time series analysis can be used to capture pollutant trends and dynamics. How?
- Air pollution data is obtained at regular intervals from a number of fixed site monitors located throughout a region (6 regions of Macedonia)
- Pollutants reported are CO_2, PM_{10}, $PM_{2.5}$, O_3, SO_2,$N0_2$ and CO. Weather data (Temperature,Humidity etc) were also collected.
- We analyze the historical data and try to forecast the pollutant concentration some few time steps ahead (prediction horizon).
- What strategy shall we adopt in for multi-time step prediction?
- How many steps or horizons can we predict with confidence?

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Recursive Prediction Strategy

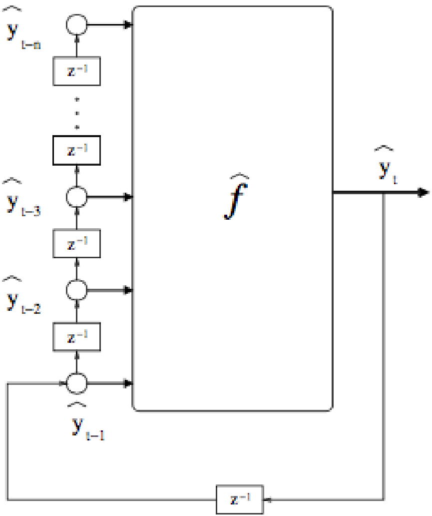


Figure: Recursive Prediction

- The recursive strategy :
  - trains first a one step model

Recursive Prediction Strategy

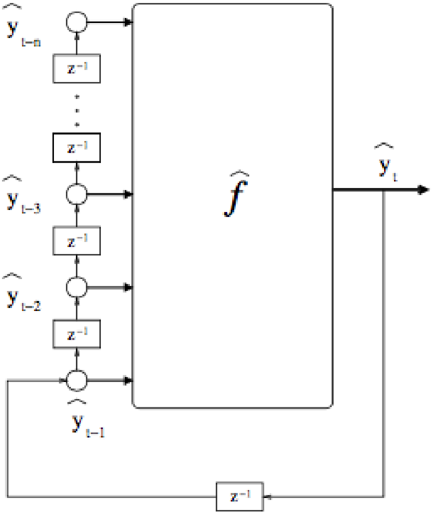


Figure: Recursive Prediction

- The recursive strategy :
  - trains first a one step model
  - then uses it recursively for returning a multistep prediction

Recursive Prediction Strategy

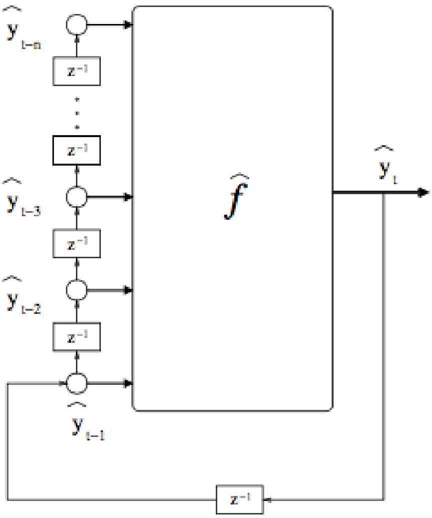


Figure: Recursive Prediction

- The recursive strategy :
  - trains first a one step model
  - then uses it recursively for returning a multistep prediction
  - The approximator $\hat{f}$ returns the prediction value at the time step $t+1$ by iterating the predictions obtained in the previous steps

A look at the Data set

- Data sets from 6 regions (Eastern Region, Western Region and Skopje Region)
- Each region contains multiple location
- We will focus on one Region (Skopje) and one location (Rektorat) for a target pollutant (PM_{10})
- We will consider influence of other pollutant on prediction accuracy
- Then we consider influence of one location data on the chosen locatio
- And also influence of data from other regions on the chosen pollutant trends.

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Bellman equation for Markov Reward Process (MRP)

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 626500 entries, 0 to 626499
Data columns (total 19 columns):
date                  626500 non-null datetime64[ns]
PM10                  370787 non-null float64
NAME                  626500 non-null object
PM10_null_pointers    626500 non-null int64
CO                    352851 non-null float64
CO_null_pointers      626500 non-null int64
NO2                   222714 non-null float64
NO2_null_pointers     626500 non-null int64
O3                    291676 non-null float64
O3_null_pointers      626500 non-null int64
PM25                  79371 non-null float64
PM25_null_pointers    626500 non-null int64
time                  626500 non-null object
month                 626500 non-null int32
day                   626500 non-null int32
hour                  626500 non-null int64
daysInterval          626500 non-null timedelta64[ns]
days_interval         626500 non-null int64
hour_interval         626500 non-null int64
dtypes: datetime64[ns](1), float64(5), int32(2), int64(8), object(2), timedelta64[ns](1)
memory usage: 86.0+ MB
```

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Lag Plot And Time series plot of data

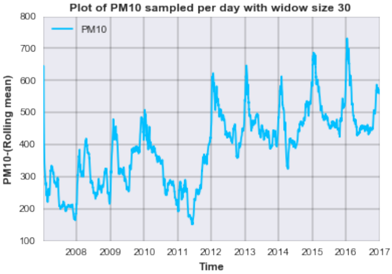


Figure: Time series plot of sampled data for PM10

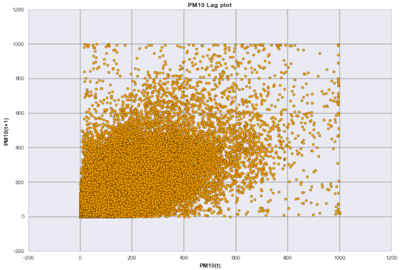


Figure: Lag plot for PM10

Dealing With Missingness

Missingness Taxonomy:

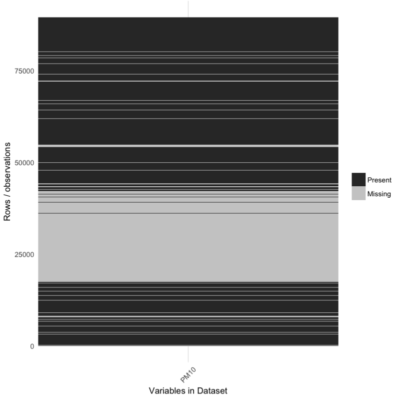
- Missing Completely at Random (MCAR)
- Missing not at Random (MNAR)
- Missing at Random (MAR)

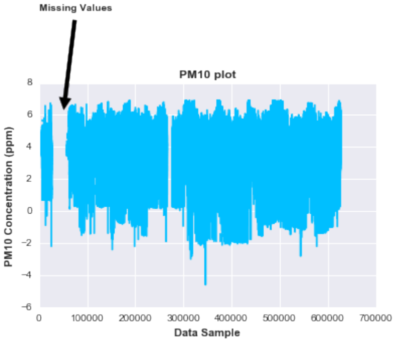
Solutions:

- Deletion
- Single Imputation method (Mean, mode)
- Model based Method(Multiple Imputation, Maximum Likelihood)

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Missingness Plot





Partial Autocorrelation

For time series, the partial auto-correlation between $PM10_t$ and $PM10_{t-h}$ is defined as the conditional correlation between $PM10t$ and $PM10_{t-h}$ conditional on $PM10_{t-h+1}, \ldots, PM10_{t-1}$: the set of observations that come between the time points t and $t-h$.

Partial Autocorrelation

For time series, the partial auto-correlation between $PM10_t$ and $PM10_{t-h}$ is defined as the conditional correlation between $PM10t$ and $PM10_{t-h}$ conditional on $PM10_{t-h+1}, \ldots, PM10_{t-1}$: the set of observations that come between the time points t and $t-h$.

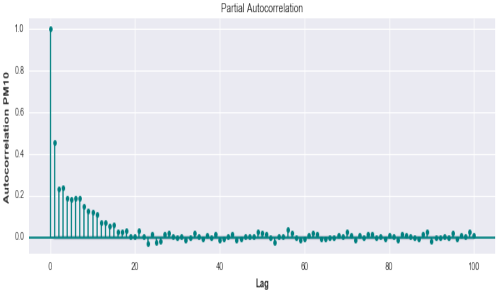


Figure: Partial Autocorrelation Plot PM10

With this we can approximately choose an horizon of 24 time-steps ahead.

Box-Cox Transformation

From Skewed to Symmetric Distribution:

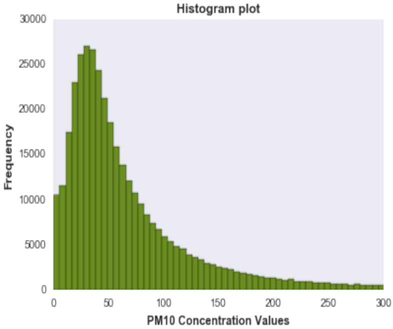


Figure: Histogram plot PM10

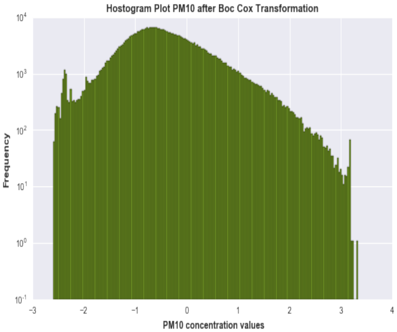


Figure: After Box Cox Transformation

Correlation Between PM_{10} And Other Pollutants

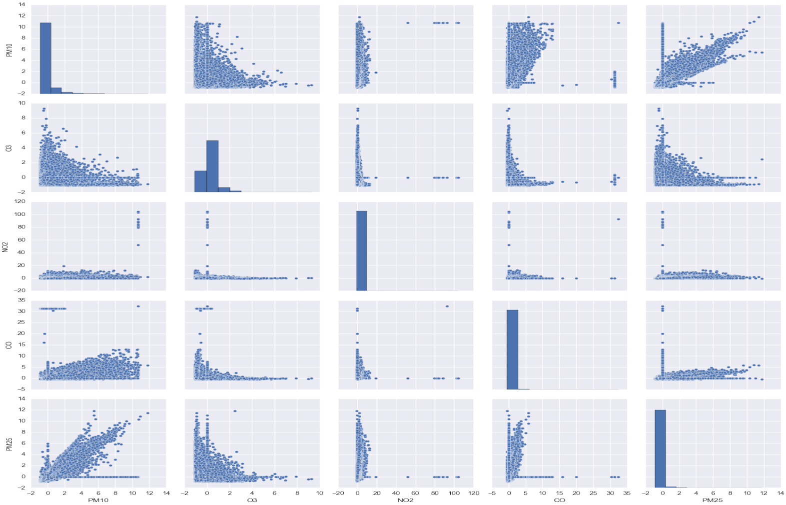


Figure: Pair Plot

Function Approximator: LSTM

LSTM : for modelling time series sets and their Long Time Dependencies accurately.

Function Approximator: LSTM

LSTM : for modelling time series sets and their Long Time Dependencies accurately.

- It has memory cell using linear and logistic unit using

Function Approximator: LSTM

LSTM : for modelling time series sets and their Long Time Dependencies accurately.

- It has memory cell using linear and logistic unit using
- Memory cells have multiplicative interaction

Function Approximator: LSTM

LSTM : for modelling time series sets and their Long Time Dependencies accurately.

- It has memory cell using linear and logistic unit using
- Memory cells have multiplicative interaction
- Information gets into cell when write gate is open

Function Approximator: LSTM

LSTM : for modelling time series sets and their Long Time Dependencies accurately.

- It has memory cell using linear and logistic unit using
- Memory cells have multiplicative interaction
- Information gets into cell when write gate is open
- Information stays in cell so long as keep gate is on.

Function Approximator: LSTM

LSTM : for modelling time series sets and their Long Time Dependencies accurately.

- It has memory cell using linear and logistic unit using
- Memory cells have multiplicative interaction
- Information gets into cell when write gate is open
- Information stays in cell so long as keep gate is on.
- Information can be read by turning on the read
- linear unit that has a self link. And retains information when weight is one
- We can backpropagate through the circuit because logistic have nice derivative.

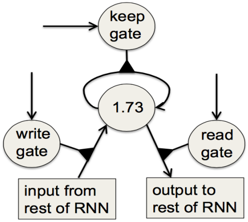


Figure: Simple RNN

Simple LSTM and Backpropagation

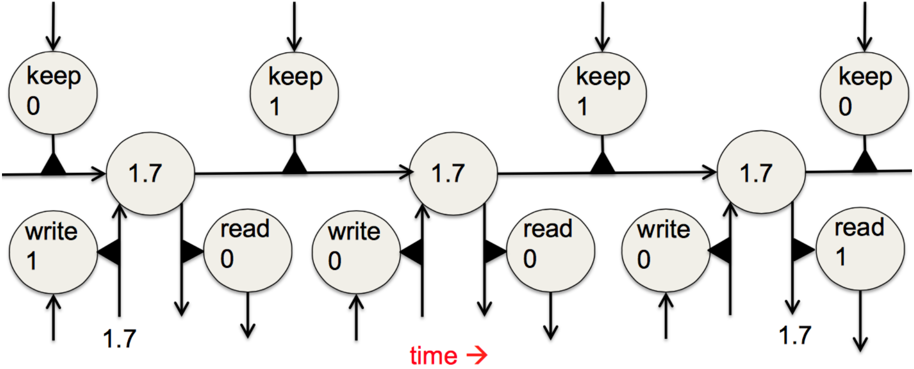


Figure: LSTM

Base Models: AR,MA and ARIMA.

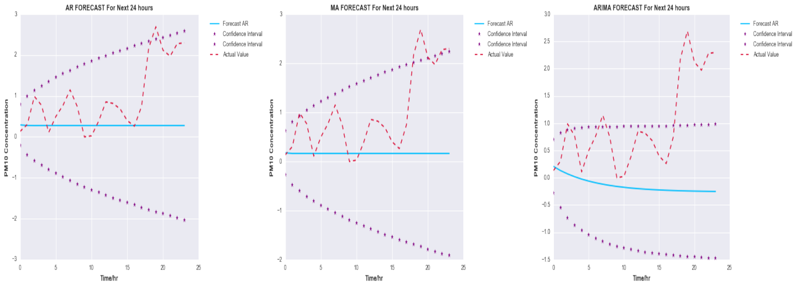
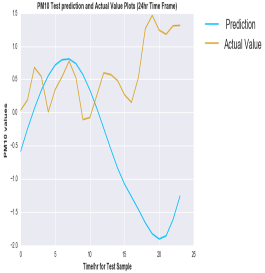


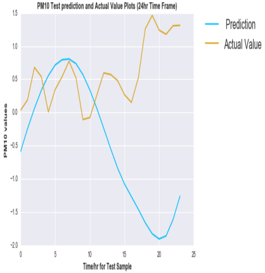
Figure: AR, MA, and ARIMA models

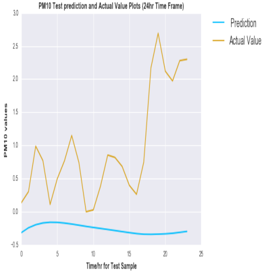
Prediction With LSTM

LSTM Models: From left to right, using only PM10, using PM10 with local mean filling, PM10 with other impurities from same region



LSTM Models: From left to right, using only PM10, using PM10 with local mean filling, PM10 with other impurities from same region





Prediction With LSTM

LSTM Models: From left to right, using only PM10, using PM10 with local mean filling, PM10 with other impurities from same region

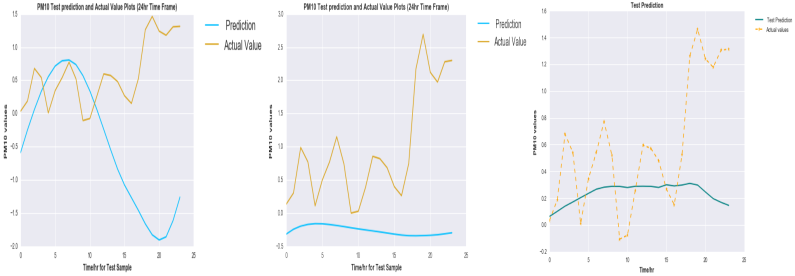


Figure: AR, MA, and ARIMA models

Table Of Prediction Score

Model	RMSE	R^2	Input	Nan Mechanism
MA	1.061809	0.0496	PM10	Mean Filling
AR	1.139356	0.05798	PM10	Mean Filling
ARIMA	1.11608	0.2330	PM10	Mean Filling
$LSTM_{pm10}$	1.6754	0.39751	PM10	Zero Filling
$LSTM_{pm10}$	1.48916	0.2210	PM10	Local Mean Filling
$LSTM_{pm10}$	0.5790	3.6e-5	PM10 plus Other Impurities	Local Mean Filling
$LSTM_{pm10}$	0.994	0.65	PM10 plus Other Impurities	Using Mask

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

LSTM Implementation

```
import tensorflow as tf;
lstm_layer = rnn_cell.LSTMCell(input_dim*312,state_is_tuple
    ↪ =False)
lstm_state = tf.Variable(tf.zeros([1, lstm_layer.state_size
    ↪ ]), trainable=False,name="initial_state")
lstm_output, lstm_state_output = lstm_layer1(features,
    ↪ lstm_state)
lstm_update_op= lstm_state.assign(lstm_state_output)
```

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

LSTM Implementation

```
import tensorflow as tf;
lstm_layer = rnn_cell.LSTMCell(input_dim*312,state_is_tuple
    ↪ =False)
lstm_state = tf.Variable(tf.zeros([1, lstm_layer.state_size
    ↪ ]), trainable=False,name="initial_state")
lstm_output, lstm_state_output = lstm_layer1(features,
    ↪ lstm_state)
lstm_update_op= lstm_state.assign(lstm_state_output)
```

Once in while set state to zero in the computational graph:

```
sess.run(lstm_state1.assign(tf.zeros([1, lstm_layer1.
    ↪ state_size])))
```

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

LSTM Implementation Continued

When using regularisation parameter, do not optimise "bias term":

```
lambda_l2_reg=0.5
l2 = lambda_l2_reg * sum(
    tf.nn.l2_loss(tf_var)
        for tf_var in tf.trainable_variables()
        if not ("bias" in tf_var.name)
)
# Minimizing error
error = tf.reduce_sum(tf.pow(final_output - y_input, 2)) +
    ↪ l2
```

- Average over Many Different Models (e.g LSTM, ARIMA and Gaussian processes)
- Use LSTM architecture but average over predictions made by many different weight vectors (Use different Hidden layer, different number or types of units per layer).

Improving Model And New Model Proposal

- Use Hidden Markov Models
- Describe a probabilistic distribution over the states (Good,Moderate, Unhealthy for sensitive groups, Unhealthy and Harzadous).
- Use LSTM for belief state prediction.
- Use Veterbi algorithm, Forward-Backward Algorithm and Expectation maximization as necessary.

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Conclusions

- LSTM offers a very promising way to predict air quality
- Missing data can also be imputed using predictions from LSTM.
- Using Mixture of models can offer considerable improvement.
- Data from neighboring locations and meteorological data can be very useful in prediction for a location of interest.
- Using Markov Models can also be an alternative.

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Conclusions

Thank You for Your Attention.
Questions ?