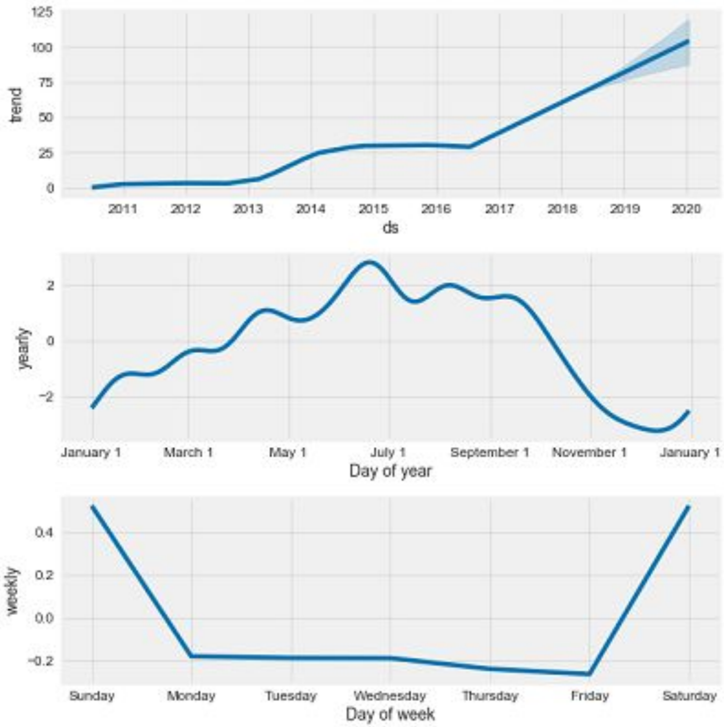
Time Series Data Visualization and Prediction using Python

Chantana Chantrapornchai
Dept. of Computer Engineering
Faculty of Engineering
Kasetsart University
Thailand

What is time series data?

- Data with time: Financial prices, weather, home energy usage, etc.
- Scale of data: seconds, minute, day, month, quarter, year etc.



https://towardsdatascience.com/time-series-analysis-in-python-an-introduction-70d5a5b1d52a

Let's see our case study: Energy data

Monitoring inefficiency of energy consumption in data center.

Chantana Chantrapornchai

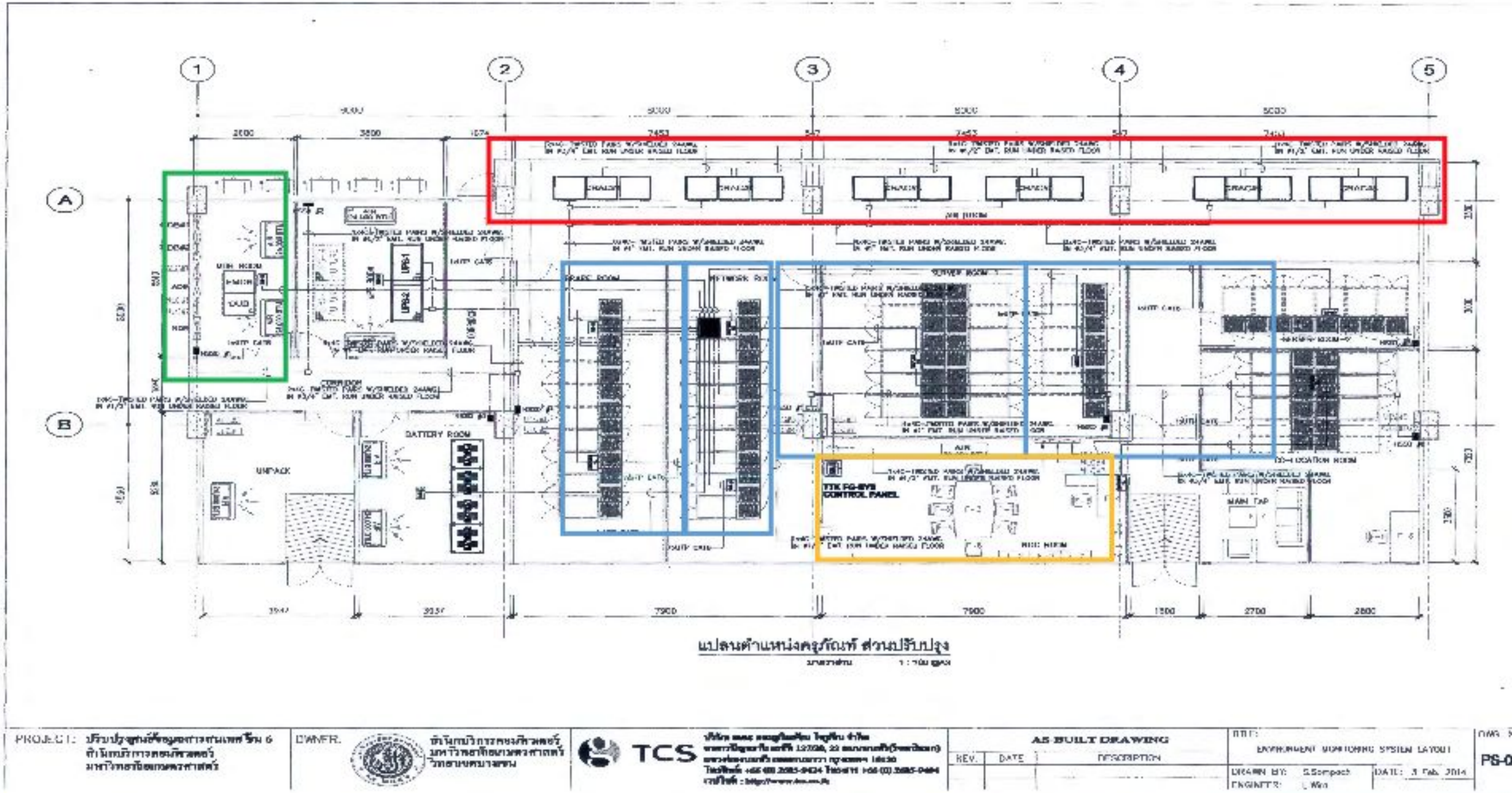
HPCNC Laboratory, Dept. of Computer Engineering

Kasetsart University

Withit Chatlatanakulchai, et.al.

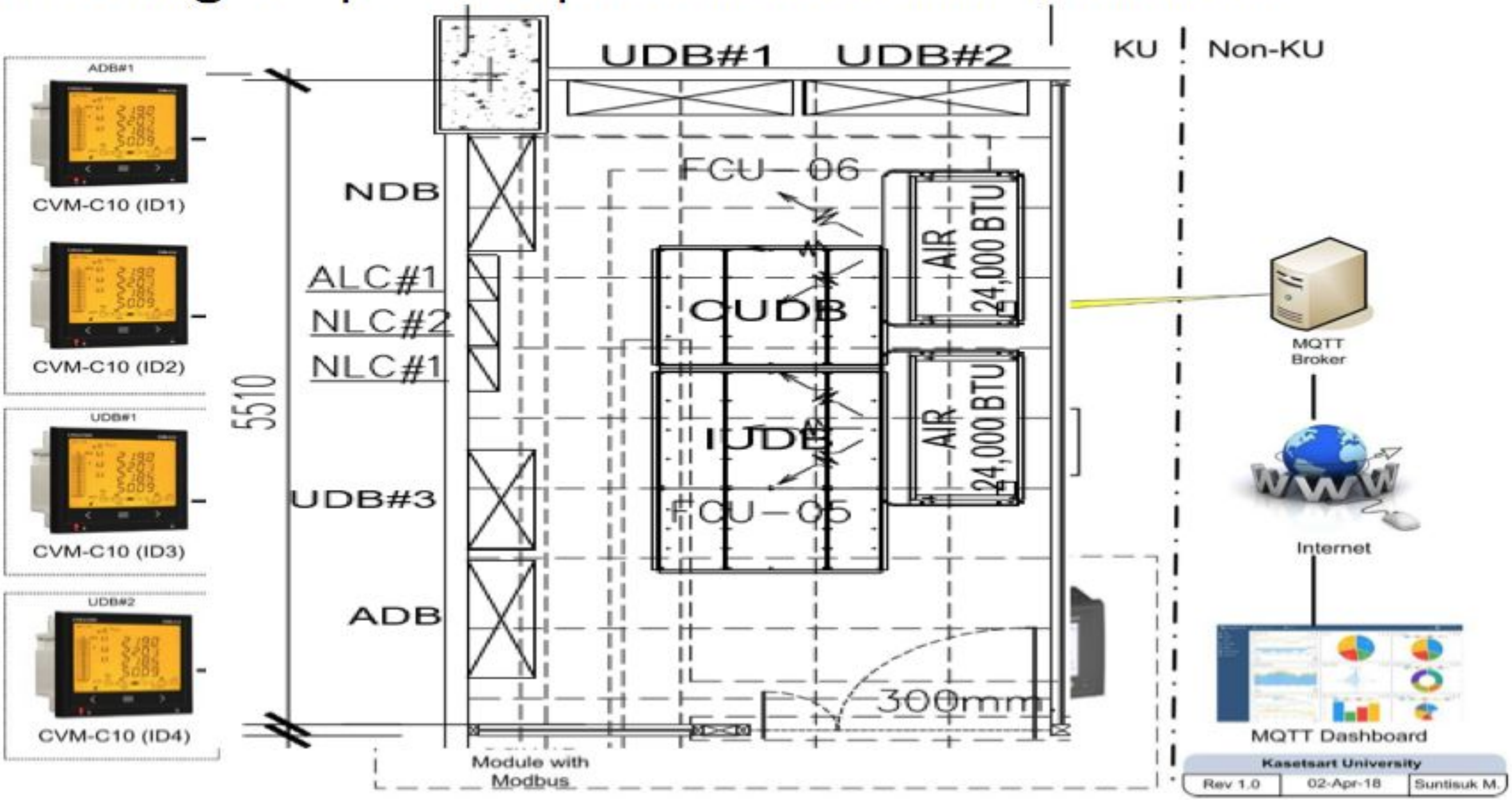
CRVLAB , Dept. of Mechanical Engineering

Kasetsart University

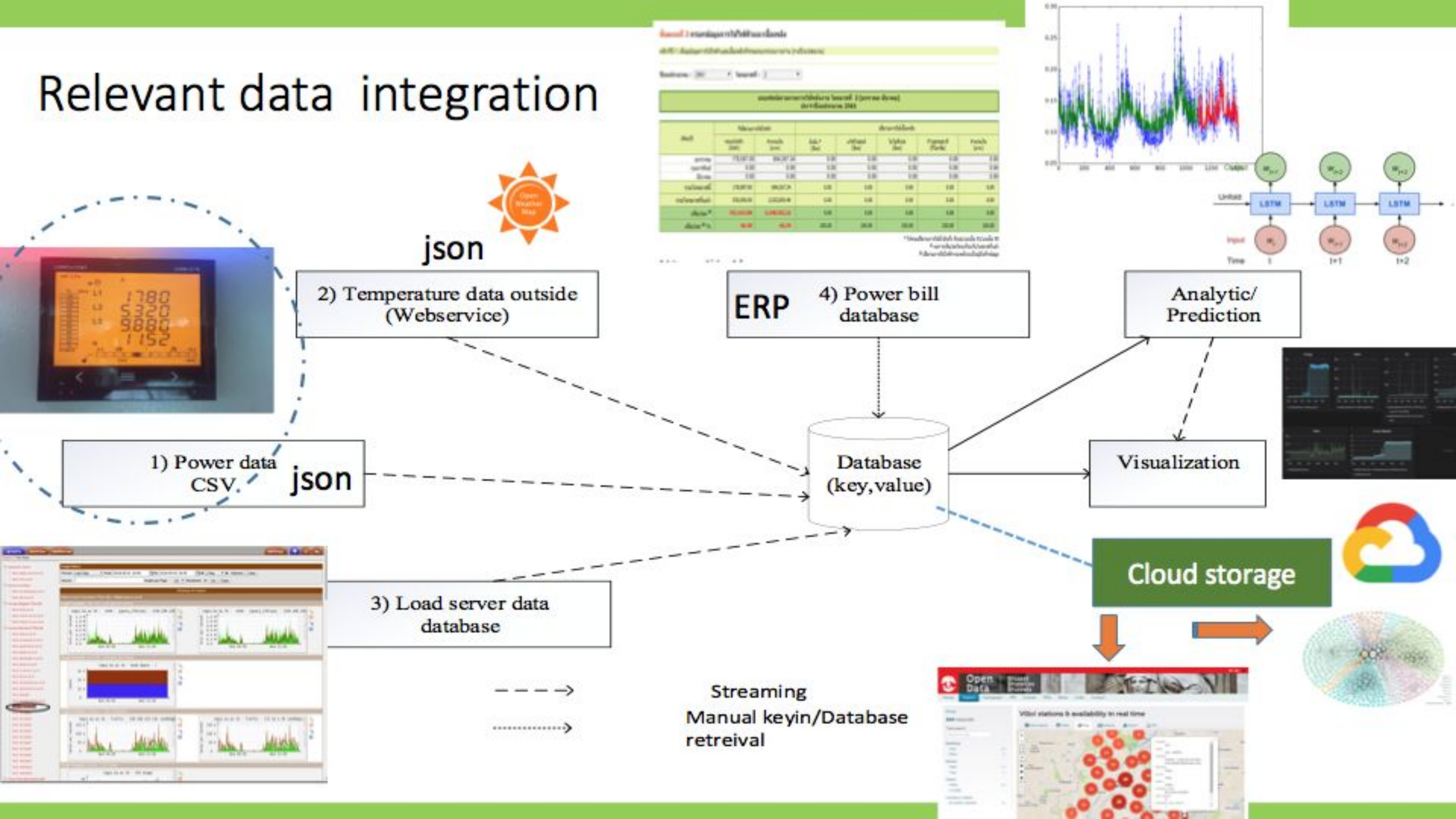


A
B
1
2
3
4
5
AIR ROOM
SPARE ROOM
NETWORK ROOM
SERVER ROOM
BATTERY ROOM
UNPACK
CO-LOCATION ROOM
NOC ROOM
MAIN TAP
UPS-1
UPS-2
FIK FG-SYS CONTROL PANEL
แปลนตำแหน่งครุภัณฑ์ ส่วนปรับปรุง
PROJECT:
OWNER:
TCS
AS BUILT DRAWING
REV.
DATE
DESCRIPTION
TITLE:
ENVIRONMENT MONITORING SYSTEM LAYOUT
DRAWN BY:
S.Sompech
DATE: 3 Feb. 2014
ENGINEER:

Measuring scope and power meter installation.

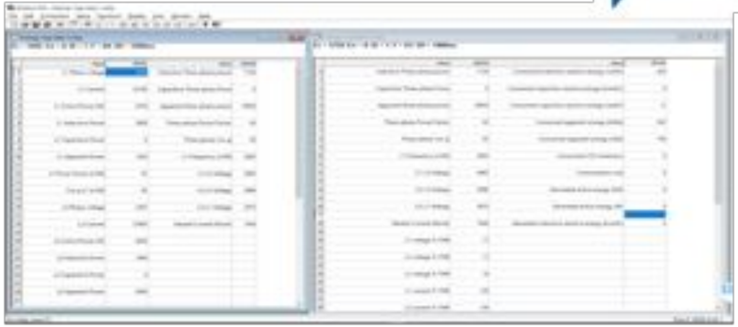


Relevant data integration

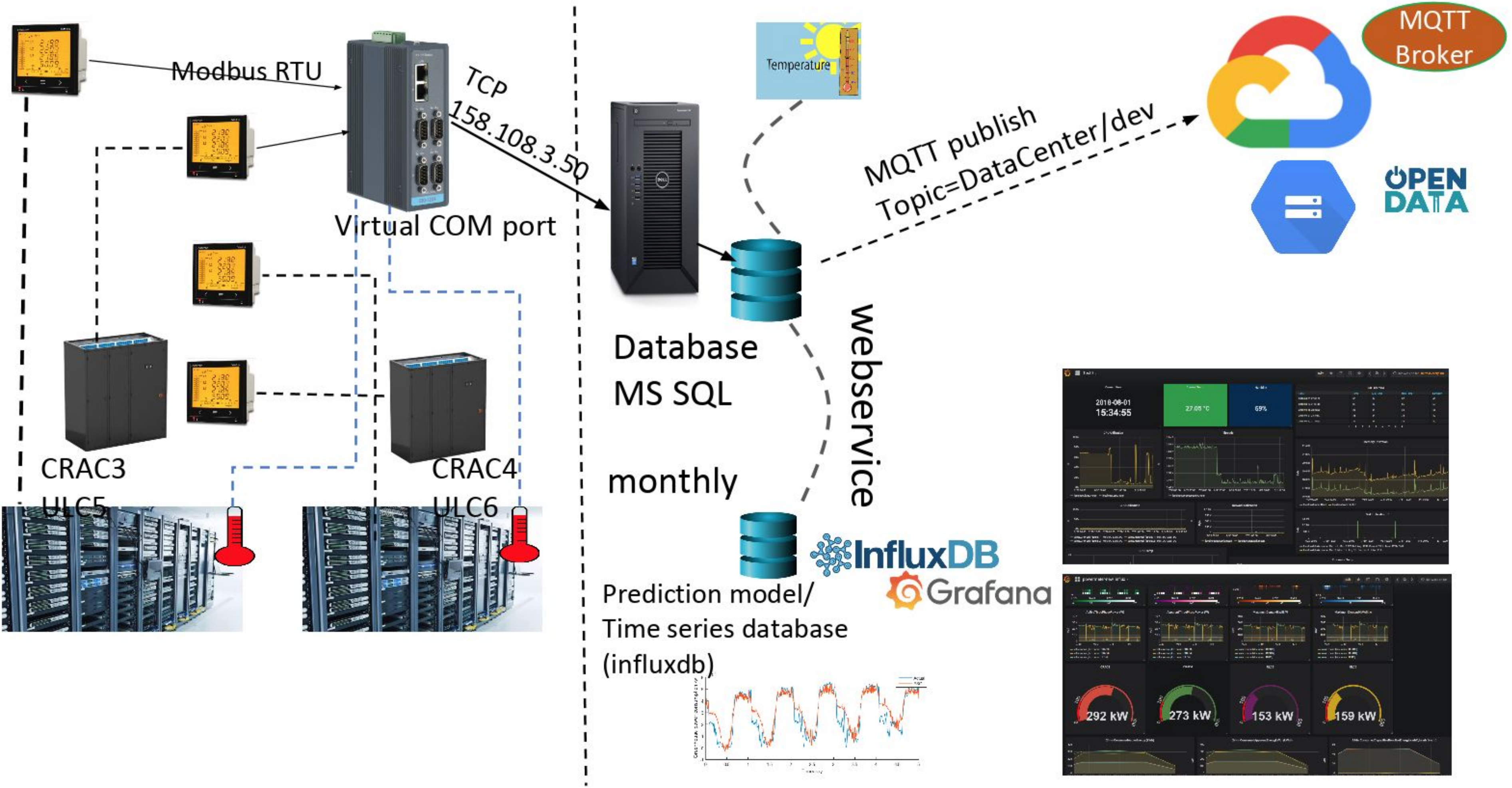


Meter installation

NATIONAL INSTRUMENTS



CIRCUTOR



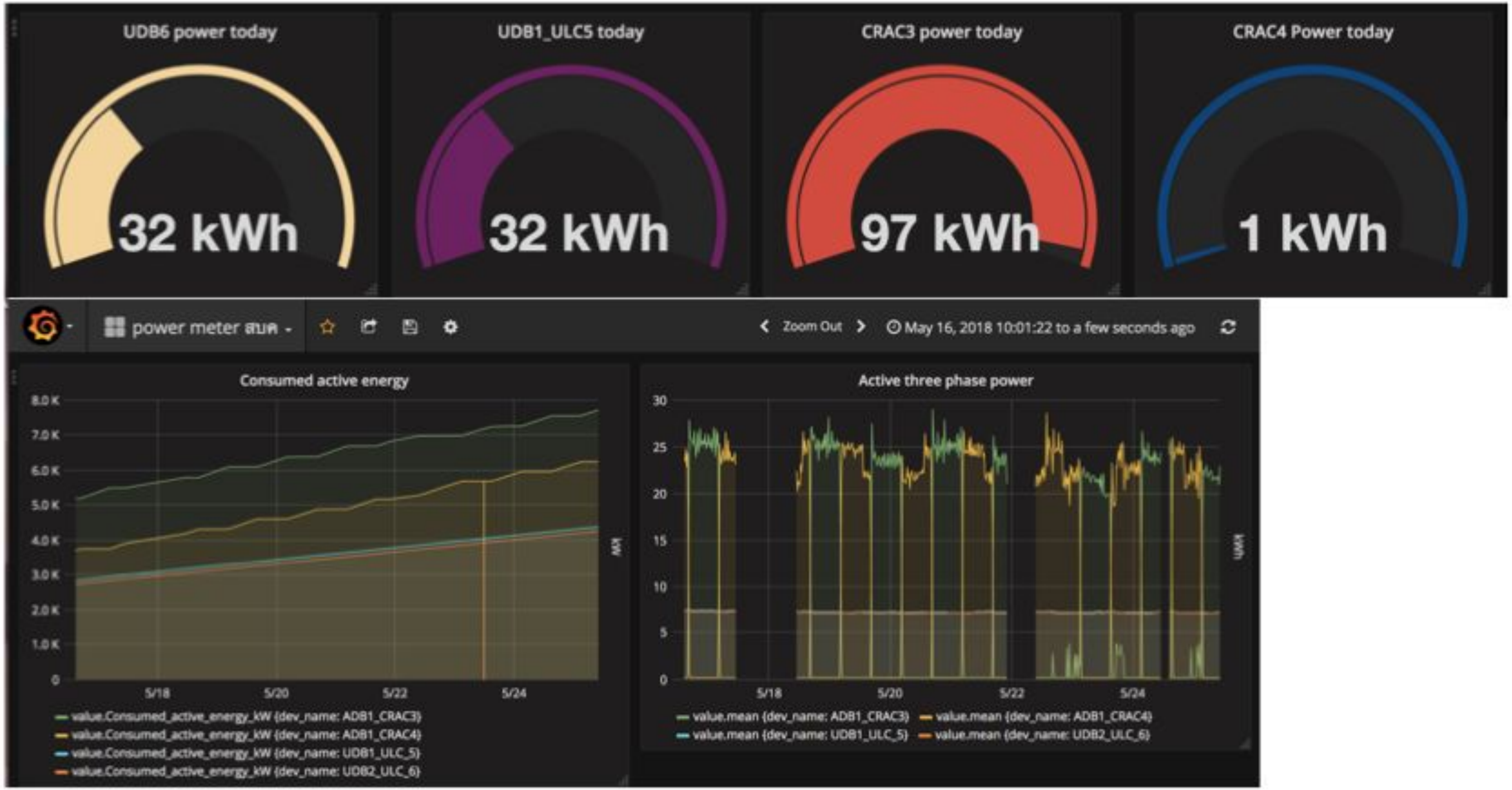
Modbus RTU
TCP
158.108.3.50
Virtual COM port
CRAC3
ULC5
CRAC4
ULC6
Temperature
Database
MS SQL
monthly
websservice
MQTT publish
Topic=DataCenter/dev
MQTT
Broker
OPEN
DATA
InfluxDB
Grafana
Prediction model/
Time series database
(influxdb)
2018-08-01
15:34:55
27.05 °C
69%
292 kW
273 kW
153 kW
159 kW

How to do visualization

Several tools: web tools.

Grafana

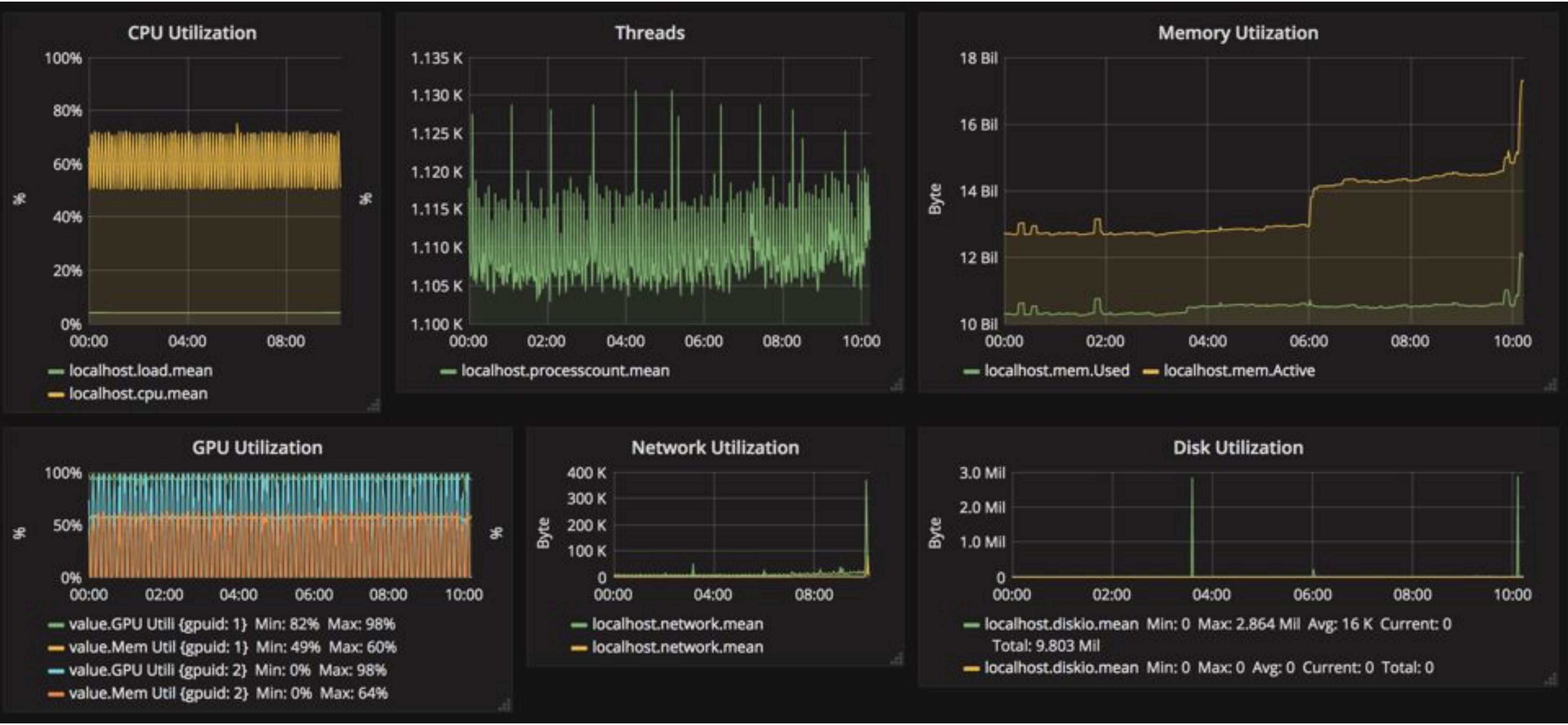
Chronograf



UDB6 power today
32 kWh
UDB1_ULC5 today
32 kWh
CRAC3 power today
97 kWh
CRAC4 Power today
1 kWh
power meter สบค
Zoom Out
May 16, 2018 10:01:22 to a few seconds ago
Consumed active energy
8.0 K
7.0 K
6.0 K
5.0 K
4.0 K
3.0 K
2.0 K
1.0 K
0
5/18
5/20
5/22
5/24
kW
value.Consumed_active_energy_kW {dev_name: ADB1_CRAC3}
value.Consumed_active_energy_kW {dev_name: ADB1_CRAC4}
value.Consumed_active_energy_kW {dev_name: UDB1_ULC_5}
value.Consumed_active_energy_kW {dev_name: UDB2_ULC_6}
Active three phase power
30
25
20
15
10
5
0
5/18
5/20
5/22
5/24
kWh
value.mean {dev_name: ADB1_CRAC3}
value.mean {dev_name: ADB1_CRAC4}
value.mean {dev_name: UDB1_ULC_5}
value.mean {dev_name: UDB2_ULC_6}

Power Meter New
(ConsumedActiveEnergykW_kWh+ConsumedActiveEnergyW_Wh)
CRAC3 Max: 28.03 K
CRAC4 Max: 25.39 K
ULC5 Max: 15.84 K
ULC6 Max: 15.77 K
ActiveThreePhasePower_W
Watt
CRAC3
CRAC4
ULC5
ULC6
CRAC4Avg15min
CRAC3Avg15min
ULC5Avg15min
ULC6Avg15min
ConsumedCapacitiveReactiveEnergyvarhC_varh
CRAC3
CRAC4
ULC5
ULC6
MaximumDemandkWIII_W
MaximumDemandkVAIII
ThreePhasePowerFactorx100


Test0
Zoom Out
Today so far
Refresh every 30s
Current time
2018-05-25
10:11:18
Current Temp
26.00 °C
Humidity
79%
Past Temp Value
Time
Temp
Min Temp
Max Temp
humidity
2018-05-25 09:51:42
31
30
31
79
2018-05-25 09:27:13
29
29
29
70
2018-05-25 08:51:42
27
27
28
83
2018-05-25 08:27:13
27
27
27
88
2018-05-25 07:51:42
27
27
27
88
1 2 3 4 5 6 7 8 9
CPU Utilization
100%
80%
60%
40%
20%
0%
%
00:00
04:00
08:00
Threads
1.135 K
1.130 K
1.125 K
1.120 K
1.115 K
1.110 K
1.105 K
1.100 K
00:00
02:00
04:00
06:00
08:00
10:00
Memory Utiization
18 Bil
16 Bil
14 Bil
12 Bil
10 Bil
Byte
00:00
02:00
04:00
06:00
08:00
10:00



CPU Utilization
localhost.load.mean
localhost.cpu.mean
Threads
localhost.processcount.mean
Memory Utiization
localhost.mem.Used
localhost.mem.Active
GPU Utilization
value.GPU Utili {gpuid: 1} Min: 82% Max: 98%
value.Mem Util {gpuid: 1} Min: 49% Max: 60%
value.GPU Utili {gpuid: 2} Min: 0% Max: 98%
value.Mem Util {gpuid: 2} Min: 0% Max: 64%
Network Utilization
localhost.network.mean
localhost.network.mean
Disk Utilization
localhost.diskio.mean Min: 0 Max: 2.864 Mil Avg: 16 K Current: 0 Total: 9.803 Mil
localhost.diskio.mean Min: 0 Max: 0 Avg: 0 Current: 0 Total: 0

Let’s explore the data with data science process

https://medium.com/@chantrapornchai

Review : Data Science Process

Data exploration

-seeing noises

-finding relationships among variables

Data cleansing

Modeling: based on selected features

Checking accuracy: also try out other models

Results visualization

Data exploration

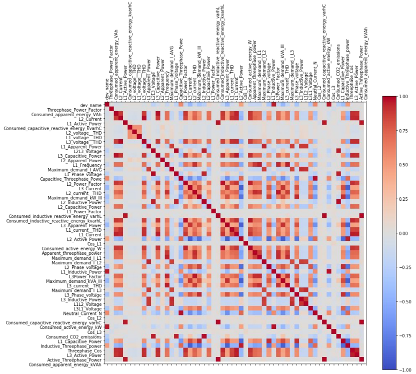
Graphical tools.

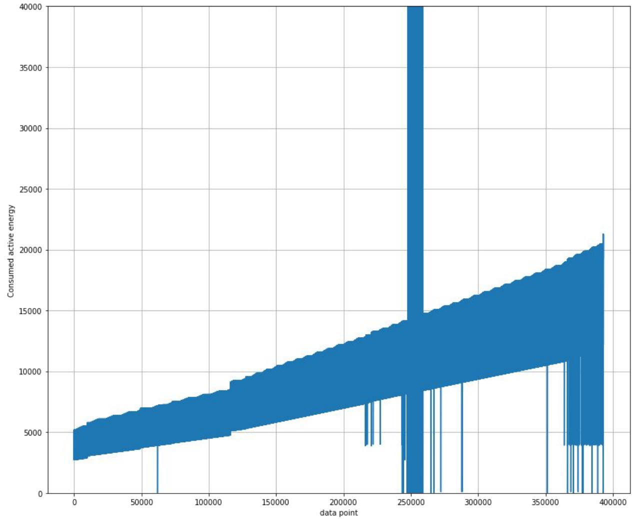
-matplotlib

Plot (x,y)

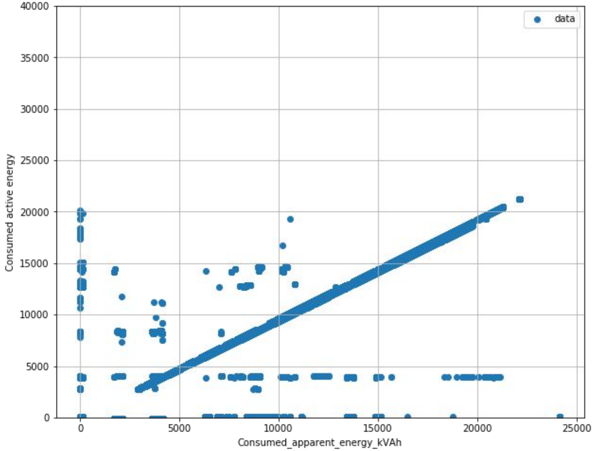
Scatter plot

Heatmap plot

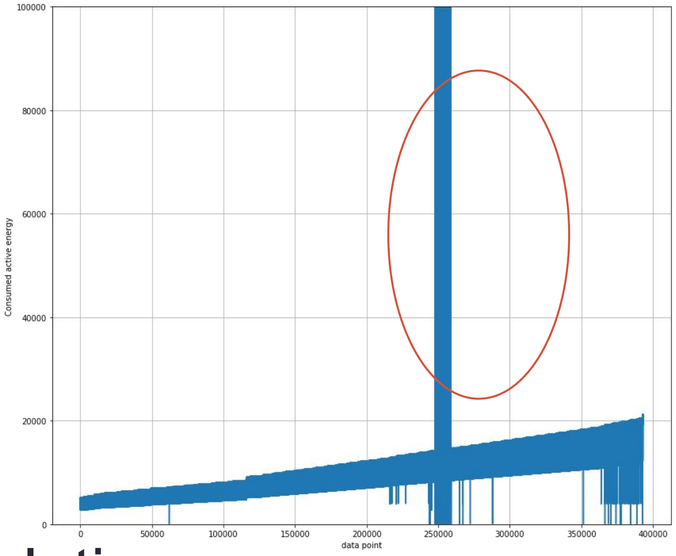




	dev_name	Threephase_Power_Factor	Consumed_apparent_energy_VAh	L2_Current	L1_Active_Power
Timestamp					
16/05/2018 14:41:02	UDB1_ULC_5	-0.93	964.0	7.56	4.12
16/05/2018 14:41:16	UDB1_ULC_5	-0.93	993.0	7.52	4.12
16/05/2018 14:41:16	UDB2_ULC_6	-0.93	693.0	7.96	3.76
16/05/2018 14:41:16	ADB1_CRAC3	1.00	756.0	0.00	0.16
16/05/2018 14:41:16	ADB1_CRAC4	0.94	619.0	34.16	8.32



Cleansing



Remove noises

-Constraints (Max,Min)

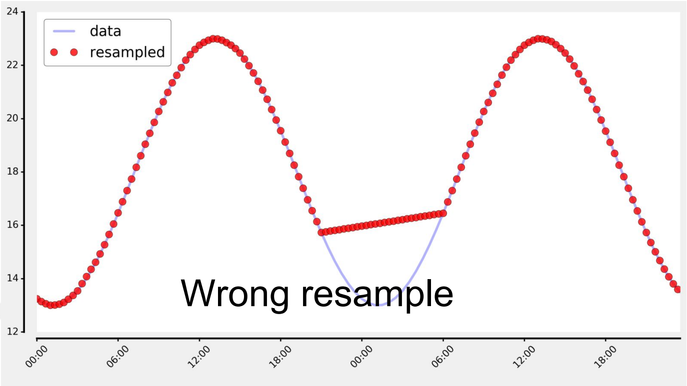
-monotonically increasing

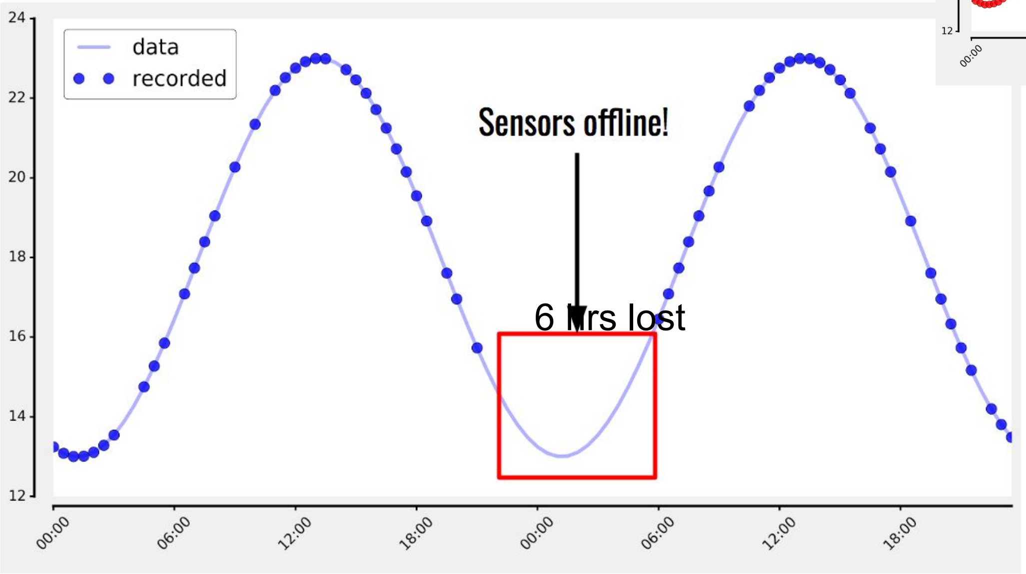
Missing data with resampling/interpolation

Adjusting periodic data

-Upsampling : Min -> Second

-Downsampling: Min -> Year


24
22
20
18
16
14
12
data
resampled
Wrong resample
00:00
06:00
12:00
18:00
00:00
06:00
12:00
18:00


24
22
20
18
16
14
12
data
recorded
Sensors offline!
6 hrs lost
00:00
06:00
12:00
18:00
00:00
06:00
12:00
18:00

original

```
                     Consumed_active_energy_kW  Consumed_apparent_energy_kVAh  \
Timestamp
2018-05-16 14:41:16                     5180.0                         5413.0
2018-05-16 14:41:46                     5180.0                         5413.0
2018-05-16 14:42:16                     5180.0                         5413.0
2018-05-16 14:42:46                     5180.0                         5413.0
```

Every 2 m

```
2018-07-06 05:48:00                          5254.000000
2018-07-06 05:50:00                          5254.000000
2018-07-06 05:52:00                          5254.000000
2018-07-06 05:54:00                          5255.000000
```

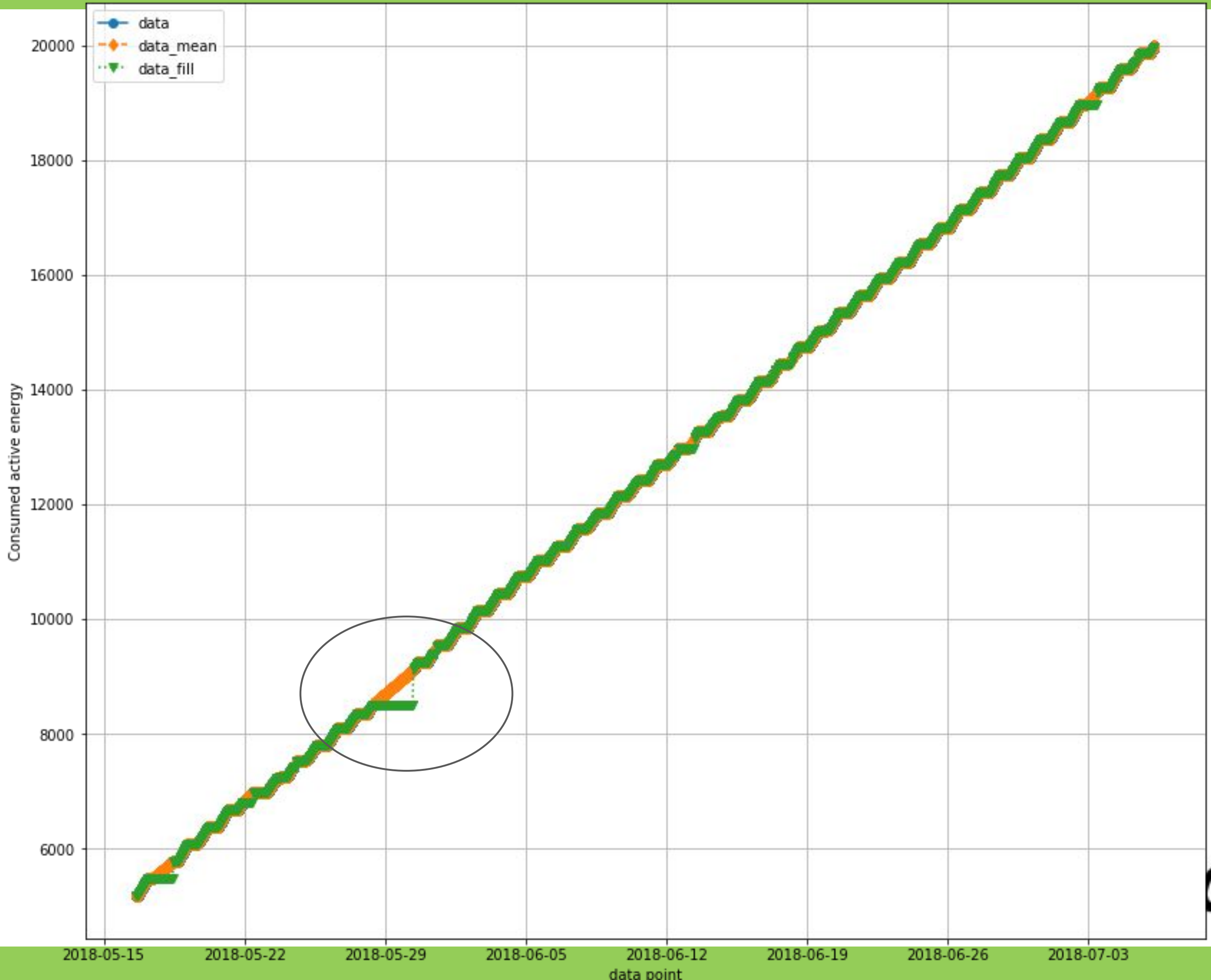
https://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.resample.html

Every 50min

```
2018-05-16 17:30:00                     5205.0                         5439.0
2018-05-16 18:20:00                     5227.0                         5461.0
2018-05-16 19:10:00                     5248.0                         5483.0
2018-05-16 20:00:00                     5269.0                         5504.0
2018-05-16 20:50:00                     5290.0                         5526.0
2018-05-16 21:40:00                     5312.0                         5549.0
```

KASETSART UNIVERSITY

HPCNC


data
data_mean
data_fill
Consumed active energy
20000
18000
16000
14000
12000
10000
8000
6000
2018-05-15
2018-05-22
2018-05-29
2018-06-05
2018-06-12
2018-06-19
2018-06-26
2018-07-03
data point

Finding relationships with existing attributes

Some analysis with statistics backgrounds

Why finding correlation?

It is a mutual relationship between quantities.

-Sales increases when increase marketing budget?

-Energy usages increase when temperature is high?

-Customer purchases more if there are packages promotions?

etc.

https://www.datascience.com/learn-data-science/fundamentals/introduction-to-correlation-python-data-science

KASETSART UNIVERSITY

HPCNC

Why we need correlation?

Help predict one quantity from another.

Indicate casual relationship.

Basic foundation to other modeling techniques.

Type of correlations

Covariances:

Covariance is a statistical measure of association between two variables *X* and *Y*.

Indicate the increase in the same direction.

$$Cov(X,Y) = E[(X - E[X])(Y - E[Y])]$$ or $$\text{cov}(X,Y) = \sum_{i=1}^{N} \frac{(x_i - \bar{x})(y_i - \bar{y})}{N}.$$

if both variables tend to move in the same direction, we expect the "average" rectangle connecting each point (X_i, Y_i) to the means (X_bar Y_bar) to have a large and positive diagonal vector, corresponding to a larger positive product in the equation above.

$$\mu = E(X) = \sum[xP(x)]$$

where

μ = mean

$E(X)$ = expected value

x = an outcome

$P(x)$ = probability of that outcome

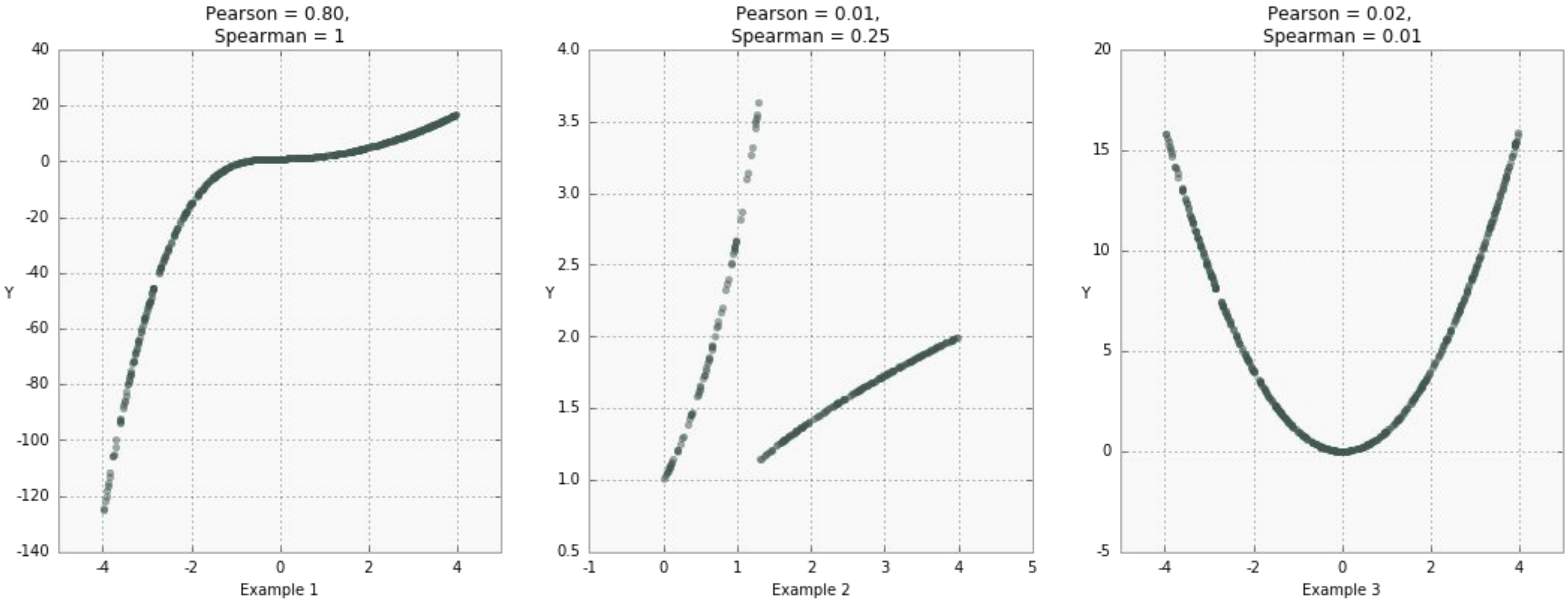
KASETSART UNIVERSITY

Kinds of correlation

Pearson: Linear association

Kendall: Linear association with ranking (monotonically increasing/decreasing)

Spearman: Linear association, directional agreement



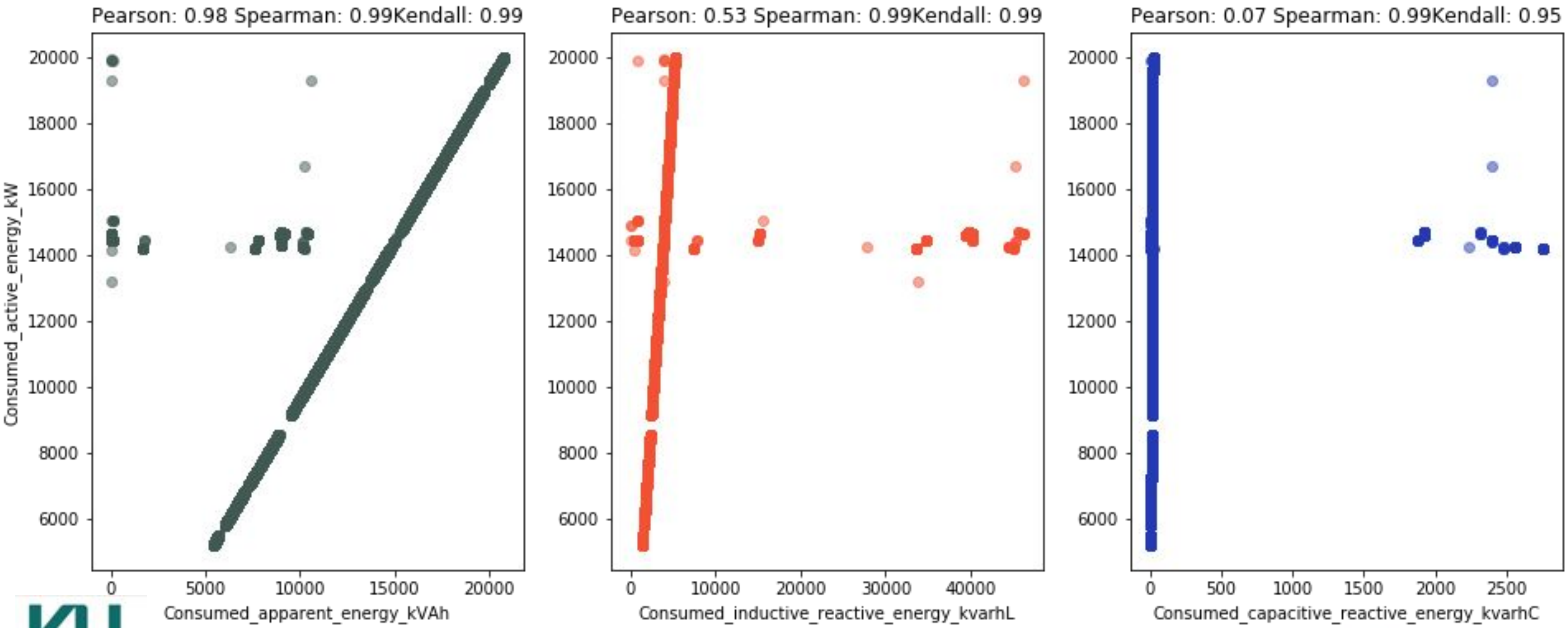
a clear monotonic (always increasing) and non-linear relationship.

clear groups in X and a strong, although non-monotonic, association for both groups with Y. Pearson correlation is almost 0 since the data is very non-linear. Spearman rank correlation shows a weak association since the data is non-monotonic.

a nearly perfect quadratic relationship centered around 0. However, both correlation coefficients are almost 0 due to the non-monotonic, non-linear, and symmetric nature of the data.

KASETSART UNIVERSITY

HPCNC



KU
KASETSART
UNIVERSITY

HPCNC

Select top (20) features from correlation using corr

Consumed_apparent_energy_kVAh	0.981747
Consumed_inductive_reactive_energy_kvarhL	0.528266
Consumed_capacitive_reactive_energy_kvarhC	0.068560
L2_voltage__THD	0.067717
Cos_L1	0.052542
L2_current__THD	0.050298
L1_current__THD	0.047905
Consumed_CO2_emissions	0.039708
Consumed_capacitive_reactive_energy_varhC	0.039587
L1_Power_Factor	0.035377
Threephase_Cos	0.034699
L1_Capacitive_Power	0.031062
L3_Phase_voltage	0.027613
Capacitive_Threephase_Powe	0.026782
Cos_L2	0.026488
Cos_L3	0.024519
L3Power_Factor	0.024410
L3_Apparent_Power	0.023013
Maximum_demand_I_L3	0.021808
L2_Power_Factor	0.021571

KASETSART UNIVERSITY

HPCNC

Perform simple linear regression

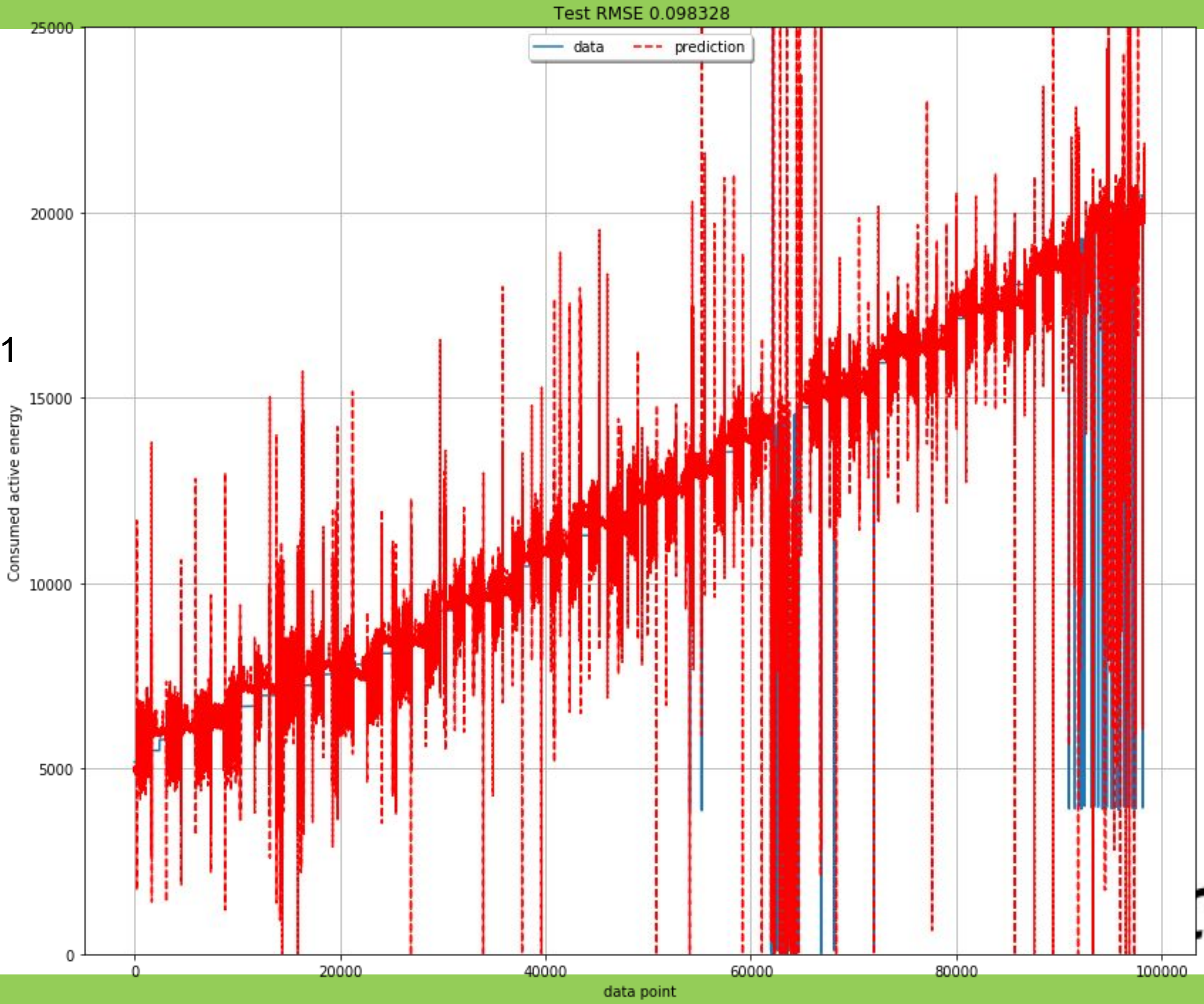
Using top 10 features to create a multivar regression model.

```
data2 = data.iloc[:, [j for j, c in enumerate(data.columns)
if c in features ]]


target = data['Consumed_active_energy_kW']
lm = linear_model.LinearRegression()
model = lm.fit(data2,target)

predictions = lm.predict(data2)
```

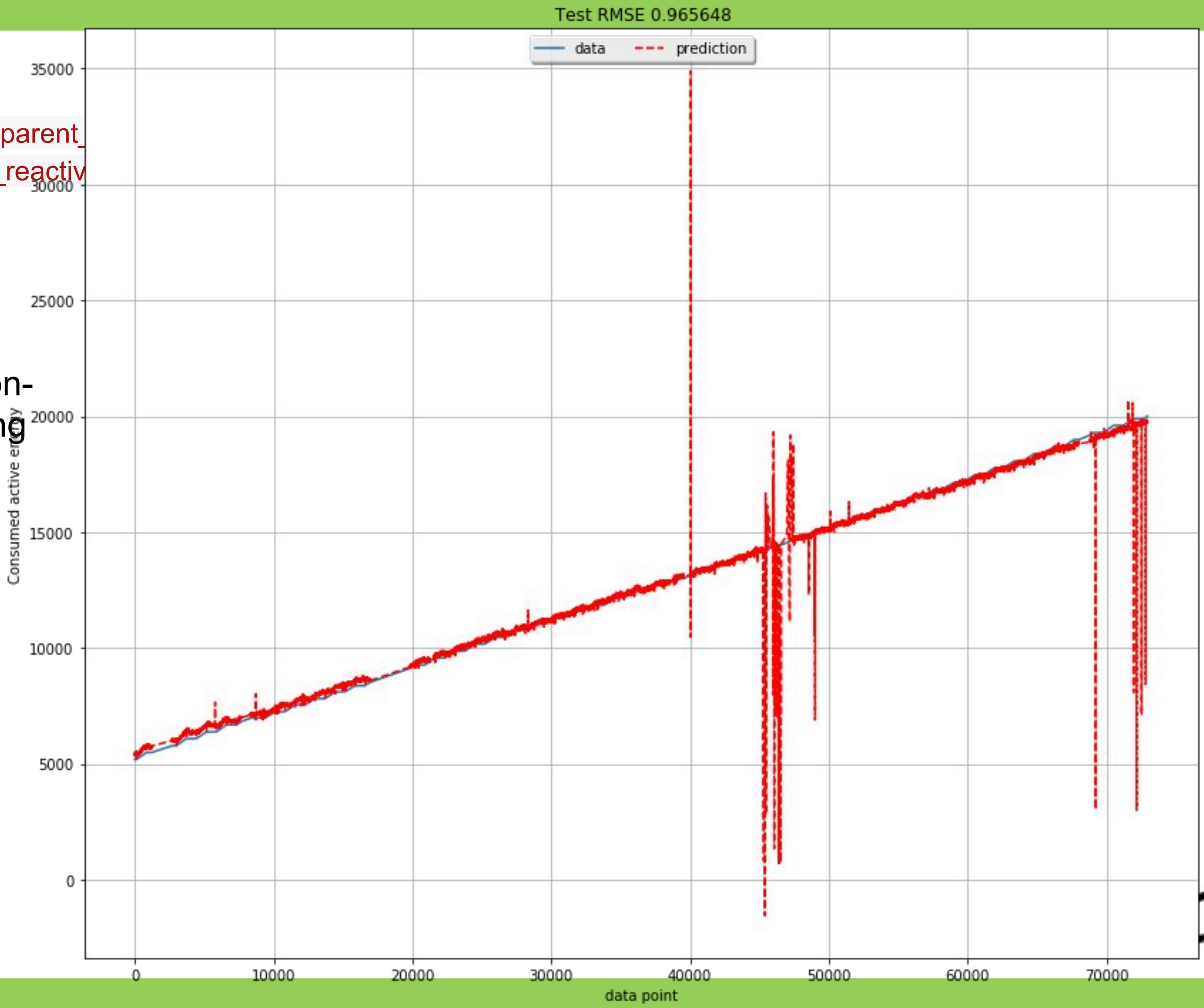
Linear regression1
No cleansing



cols = ['Consumed_apparent_
'Consumed_inductive_reactiv

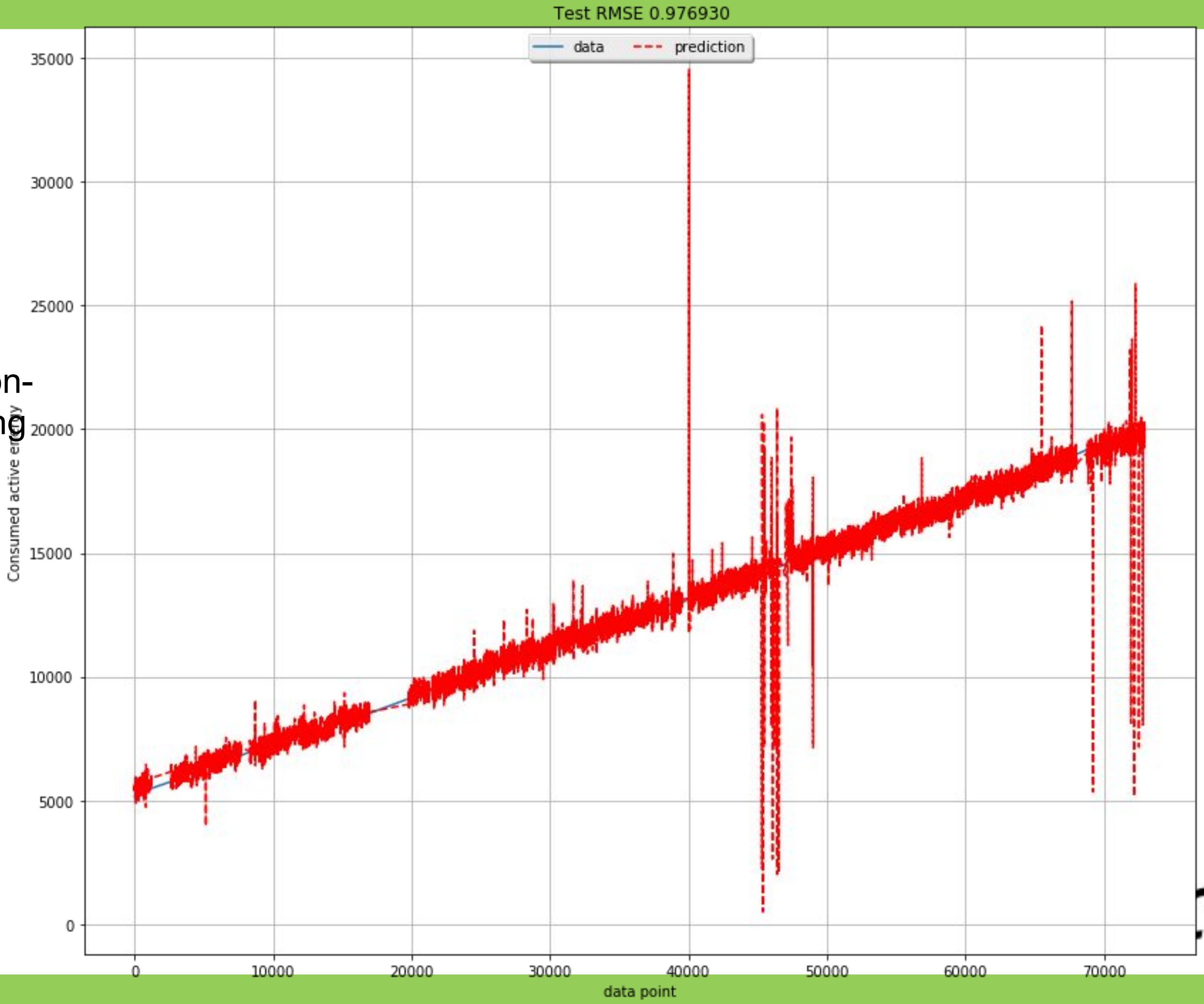
Linear regression-
clean+resampling

Use 10 columns



Linear regression-clean+resampling

Use all columns



Time series data with ARIMA

KASETSART UNIVERSITY

HPCNC

ARIMA models (for Autoregressive Integrated Moving Average)

-Models that relate the present value of a series to past values and past prediction errors

Contains: AR and MA.

AR is autoregression term and MA is the noise term.

https://medium.com/@chantrapornchai/arima-for-energy-data-i-a7b466590af4

Points to consider

-Is there a **trend,** meaning that, on average, the measurements tend to increase (or decrease) over time?
-Is there **seasonality**, meaning that there is a regularly repeating pattern of highs and lows related to calendar time such as seasons, quarters, months, days of the week, and so on?
-Are their **outliers**? In regression, outliers are far away from your line.

-Is there **a long-run cycle or period** unrelated to seasonality factors?
-Is there **constant variance over time, or is the variance non-constant**?
-Are there any abrupt changes to either the level of the series or the variance?

KU KASETSART UNIVERSITY

HPCNC

Autoregression

Time series data

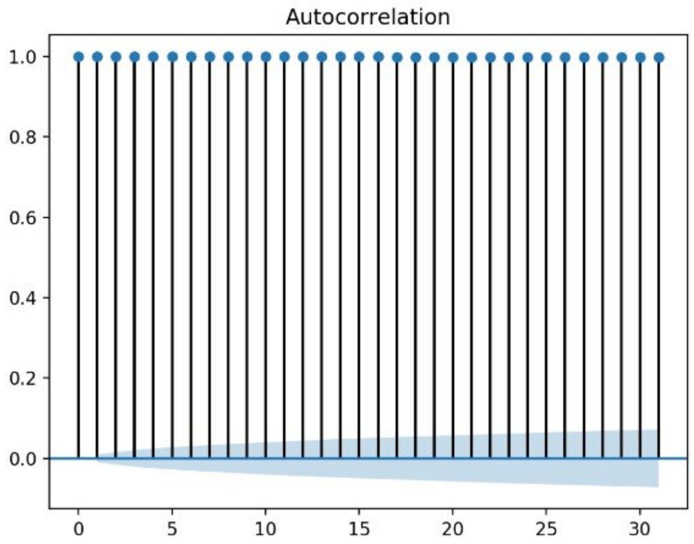
Predict x(t+1) from x(t-2), x(t-1), x(t)

--Finding autocorrelation

Finding Autocorrelation

	X(t-1)	X(t+1)
X(t-1)	1.0	1.0
X(t+1)	1.0	1.0

good!



Lag=31

Testing stationary

One requirement is to have a stationary model for ARIMA

That is it has a constant moving average.

The simplest way to find out is to calculate the moving average for a given interval. Or use Dickey-Fuller test.

Stationary analysis

Test Statistic	**0.072995**
p-value	0.964168
#Lags Used	62.000000
Number of Observations Used	72901.000000
Critical Value (5%)	-2.861580
Critical Value (1%)	-3.430440
Critical Value (10%)	-2.566791
dtype: float64	

Test statistics should be less than Critical Value (1%)

KASETSART UNIVERSITY

HPCNC

Eliminate trend and seasonal

Trend: log plot

Seasonal: diff

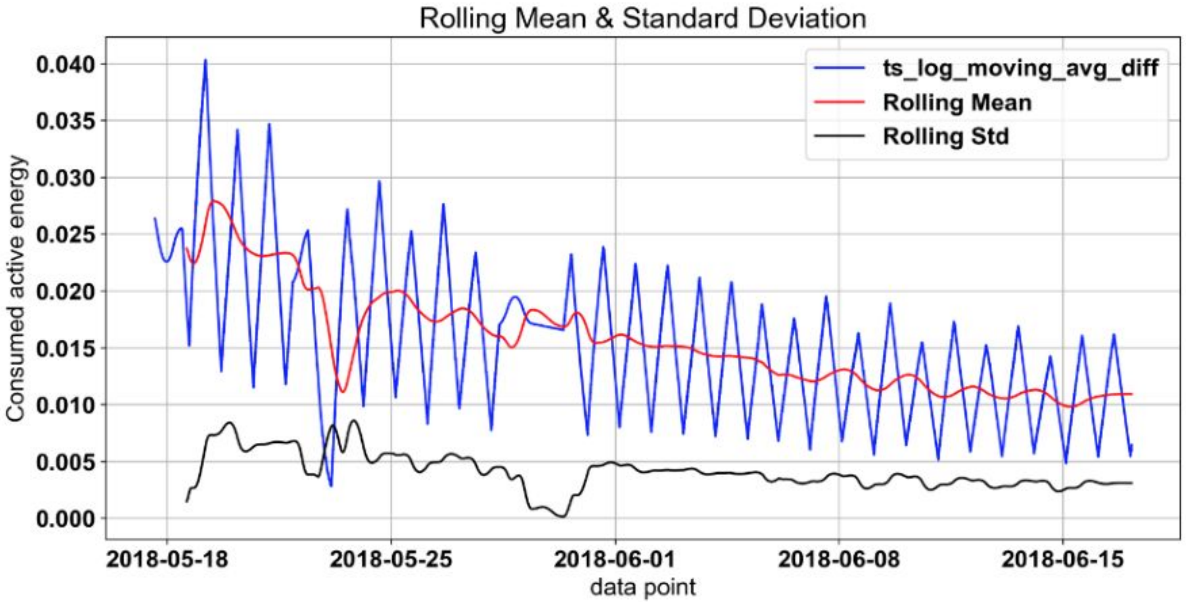
Results of Dickey-Fuller Test:

Test Statistic	**-1.041819e+01**
p-value	1.732179e-18
#Lags Used	6.300000e+01
Number of Observations Used	7.289900e+04
Critical Value (5%)	-2.861580e+00
Critical Value (1%)	-3.430440e+00
Critical Value (10%)	-2.566791e+00

dtype: float64

Solving non-stationary

-Take difference (first order diff : with prev value)



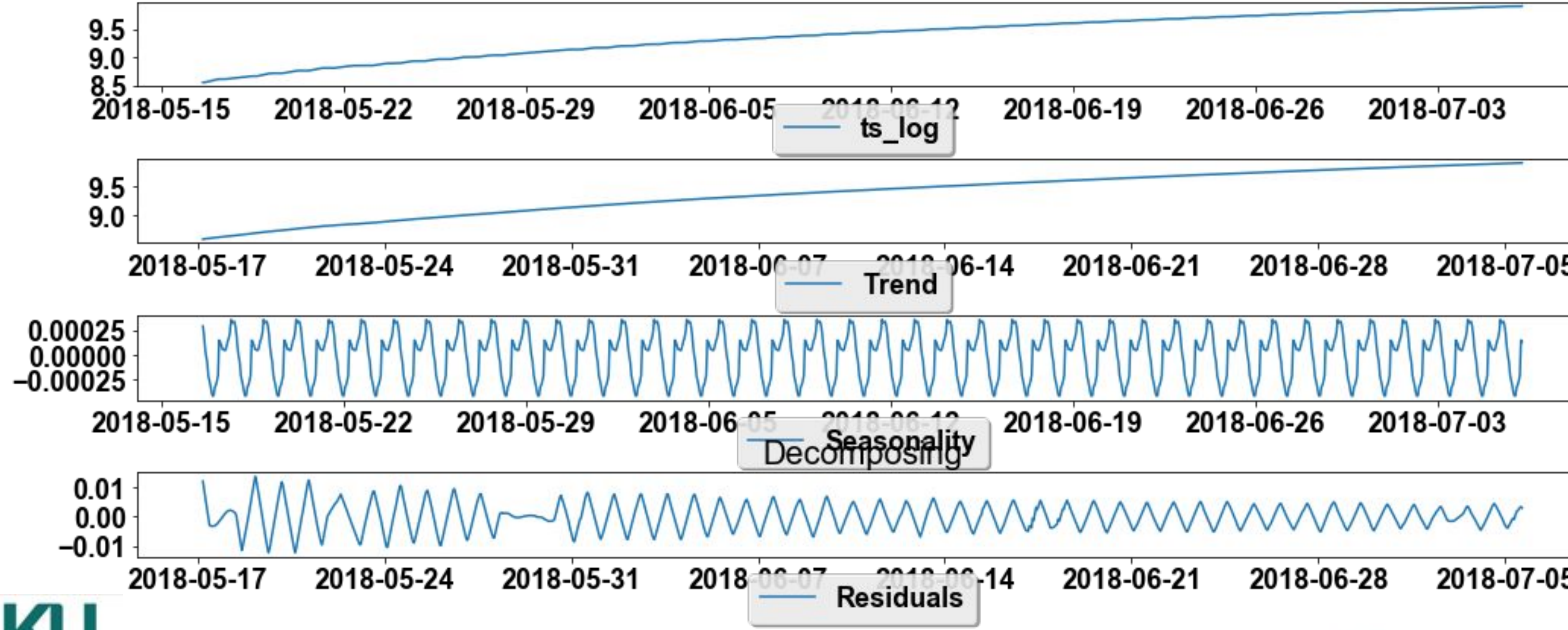
Decomposition of time series data

- **Level**: The average value in the series.
- **Trend**: The increasing or decreasing value in the series.
- **Seasonality**: The repeating short-term cycle in the series.
- **Noise**: The random variation in the series.

$$y(t) = \text{Level} + \text{Trend} + \text{Seasonality} + \text{Noise} \quad \text{or}$$

$$y(t) = \text{Level} * \text{Trend} * \text{Seasonality} * \text{Noise}$$

Decomposing



Results of Dickey-Fuller Test:

Test Statistic	**-1.456013e+01**
p-value	4.809563e-27
#Lags Used	6.200000e+01
Number of Observations Used	6.930100e+04
Critical Value (5%)	-2.861582e+00
Critical Value (1%)	-3.430444e+00
Critical Value (10%)	-2.566792e+00

dtype: float64

ACF plot (Sample Autocorrelation Function)

The ACF of the series gives correlations between $x(t)$ and $x(t-h)$ for $h = 1, 2, 3$, etc.

ACF makes sense when the series is weakly stationary.

Partial Autocorrelation Function (PACF)

A partial correlation is a conditional correlation.

-It is the correlation between two variables under the assumption that we know and take into account the values of some other set of variables.

$$\text{PACF} = \frac{\text{Covariance}(y, x_3 | x_1, x_2)}{\sqrt{\text{Variance}(y|x_1, x_2)\text{Variance}(x_3|x_1, x_2)}}$$

Correlating **the residuals from two different regressions:**

(1) Regression in which we predict y from x_1 and x_2,

(2) Regression in which we predict x_3 from x_1 and x_2.

Basically, we correlate the "parts" of y and x_3 that are not predicted by x_1 and x_2.

KASETSART UNIVERSITY

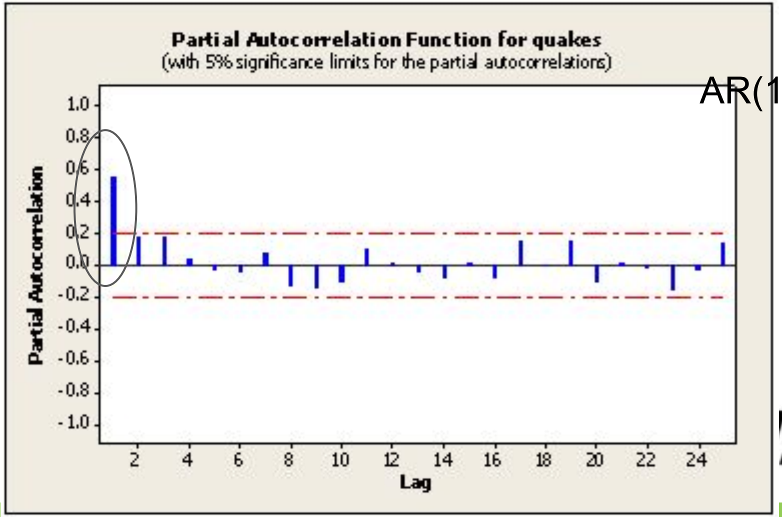
HPCNC

Notes: ACF, PACF

-Identification of an AR model is often best done with the PACF.

For an AR model, the theoretical PACF "shuts off" past the order of the model (decay to 0). The number of non-zero partial autocorrelations gives the order of the AR model.

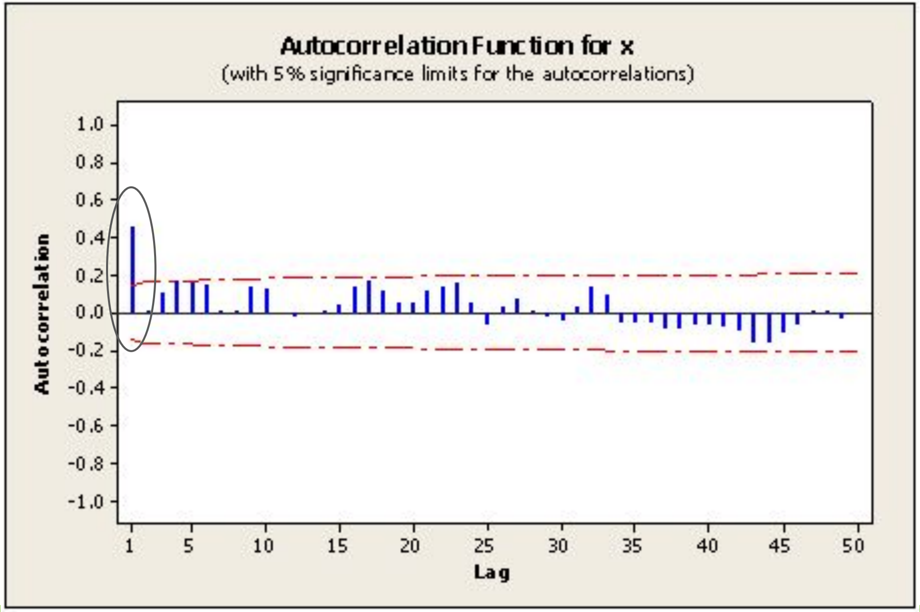
Note that the first lag value is statistically significant, whereas partial autocorrelations for all other lags are not statistically significant.



AR(1)

-Identification of an MA model is often best done with the ACF rather than the PACF.

$$x_t = 10 + w_t + 0.7w_{t-1}$$



For an MA model, the theoretical PACF does not shut off, but instead tapers toward 0 in some manner.

A clearer pattern for an MA model is in the ACF. The ACF will have non-zero autocorrelations only at lags involved in the model.

Non-seasonal ARIMA

Model parameter: (AR order, differencing, MA order).

-A model with (only) two AR terms would be specified as an ARIMA of order (2,0,0).
-A MA(2) model would be specified as an ARIMA of order (0,0,2).
-A model with one AR term, a first difference, and one MA term would have order (1,1,1).

ARIMA (1,1,1), a model with one AR term and one MA term is being applied to the variable. A first difference might be used to account for a linear trend in the data.

$$z_t = x_t - x_{t-1}$$

Difference order=2,

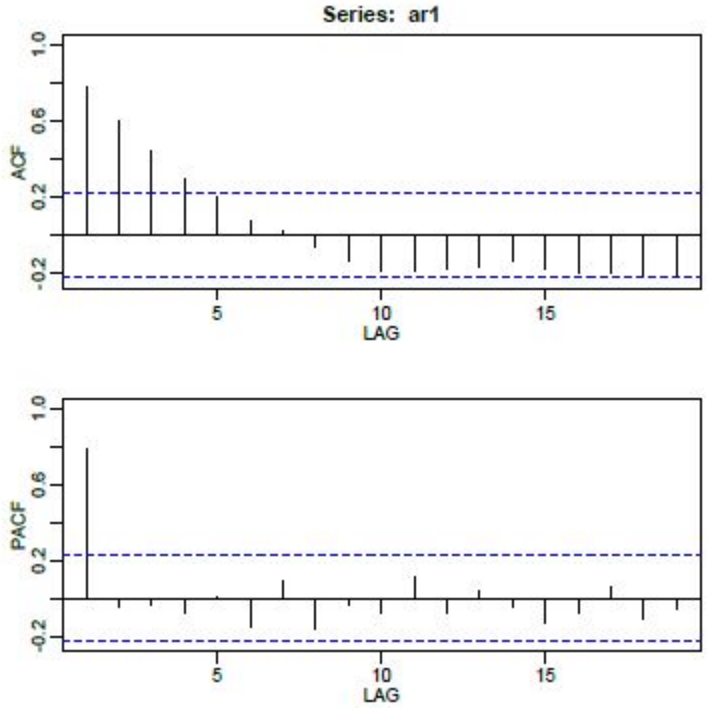
$$z_t = (x_t - x_{t-1}) - (x_{t-1} - x_{t-2})$$

Steps to identify possible model

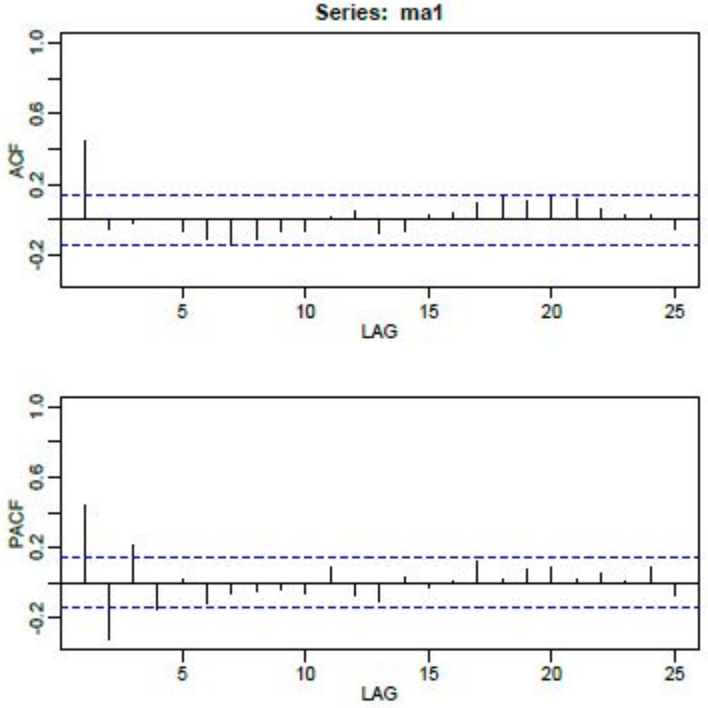
-plot data

-Calculate ACF, PACF

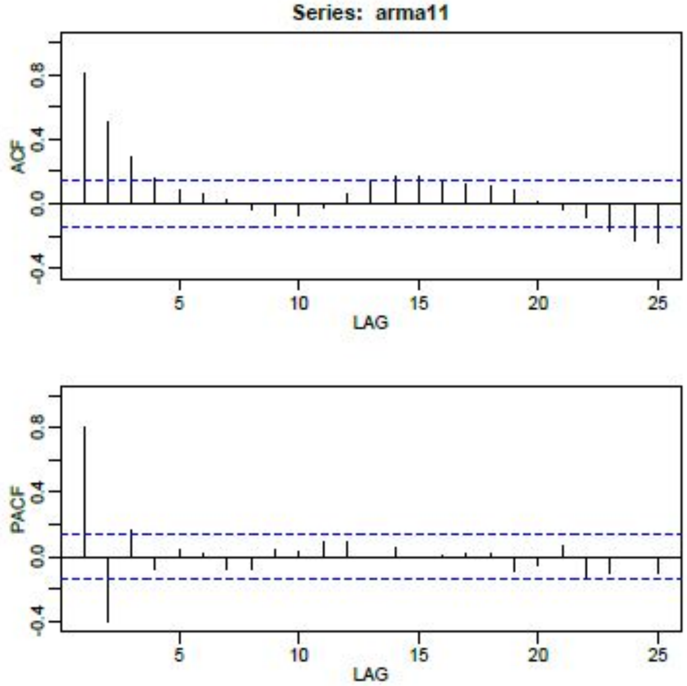
AR(1) - PACFs with non-zero values at the AR terms in the model and zero values elsewhere. The ACF will taper to zero in some fashion.

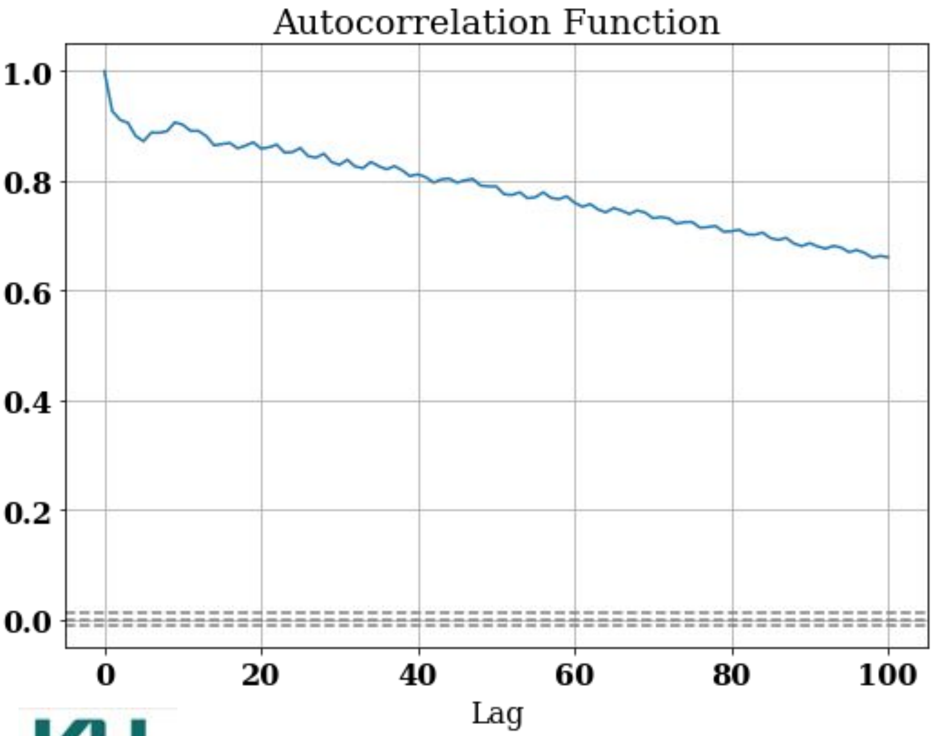


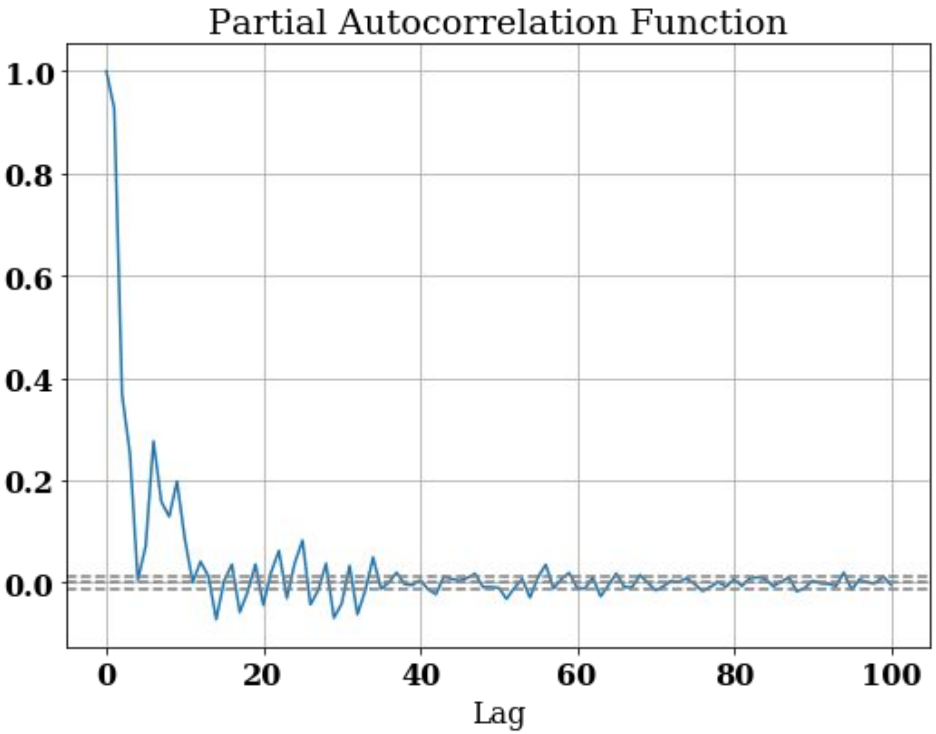
MA(1) models have theoretical ACFs with non-zero values at the MA terms in the model and zero values elsewhere.



ARIMA models (including both AR and MA terms) have ACFs and PACFs that both tail off to 0.







KASETSART UNIVERSITY

HPCNC

Seasonal ARIMA

a regular pattern of changes that repeats over S time periods, where S defines the number of time periods until the pattern repeats again.

For quarterly data, S = 4 time periods per year.

Seasonal AR and MA terms **predict x(t) using data values and errors at times with lags that are multiples of S** (the span of the seasonality).

With monthly data (and S = 12), a seasonal first order autoregressive model would **use x(t-12) to predict x(t).**

For instance, if we were selling cooling fans we might predict this August's sales using last August's sales. (This relationship of predicting using last year's data would hold for any month of the year.)

A seasonal second order autoregressive model would **use x(t-12) and x(t-24) to predict x(t).** Here we would predict this August's values from the past two Augusts.

A seasonal first order MA(1) model (with S = 12) would use w(t-12) as a predictor.
A seasonal second order MA(2) model would use w(t-12) and w(t-24).

KASETSART UNIVERSITY

HPCNC

Seasonal ARIMA Model

-incorporates both non-seasonal and seasonal factors in a multiplicative model.

$$ARIMA(p, d, q) \times (P, D, Q)S,$$

with p = non-seasonal AR order, d = non-seasonal differencing, q = non-seasonal MA order, P = seasonal AR order, D = seasonal differencing, Q = seasonal MA order, and S = time span of repeating seasonal pattern.

Differencing

Seasonality usually causes the series to be non-stationary

because the average values at some particular times within the seasonal span (months, for example) may be different than the average values at other times.

For instance, our sales of cooling fans will always be higher in the summer months.

Seasonal differencing

Seasonal differencing is defined as a difference between a value and a value with lag that is a multiple of S.

Seasonal differencing removes seasonal trend and can also get rid of a seasonal random walk type of nonstationarity.

$$(1-B^{12})x_t = x_t - x_{t-12}$$

-With S = 12, which may occur with monthly data, a seasonal difference is

The differences (from the previous year) may be about the same for each month of the year giving us a stationary series.

With S = 4, which may occur with quarterly data, a seasonal difference is

$$(1-B^4)x_t = x_t - x_{t-4}$$

Differencing for Trend and Seasonality

When both trend and seasonality are present, we may **apply both a non-seasonal first difference and a seasonal difference.**

We may need to examine the ACF and PACF

$$(1 - B^{12})(1 - B)x_t = (x_t - x_{t-1}) - (x_{t-12} - x_{t-13})$$

Removing trend doesn't mean that we have removed the dependency.

We may have removed the mean, μ_t, part of which may include a periodic component.

That is: the dependency is broken down into recent things that have happened and long-range things that have happened.

Non-seasonal Behavior Will Still Matter

-Seasonal data consists of a short run non-seasonal components in the model.

-In the monthly sales of cooling fans for instance, sales in the previous month or two, along with the sales from the same month a year ago, can help predict this month’s sales.

-the ACF and PACF behavior over the first few lags (less than S) are used to assess what non-seasonal terms might work in the model

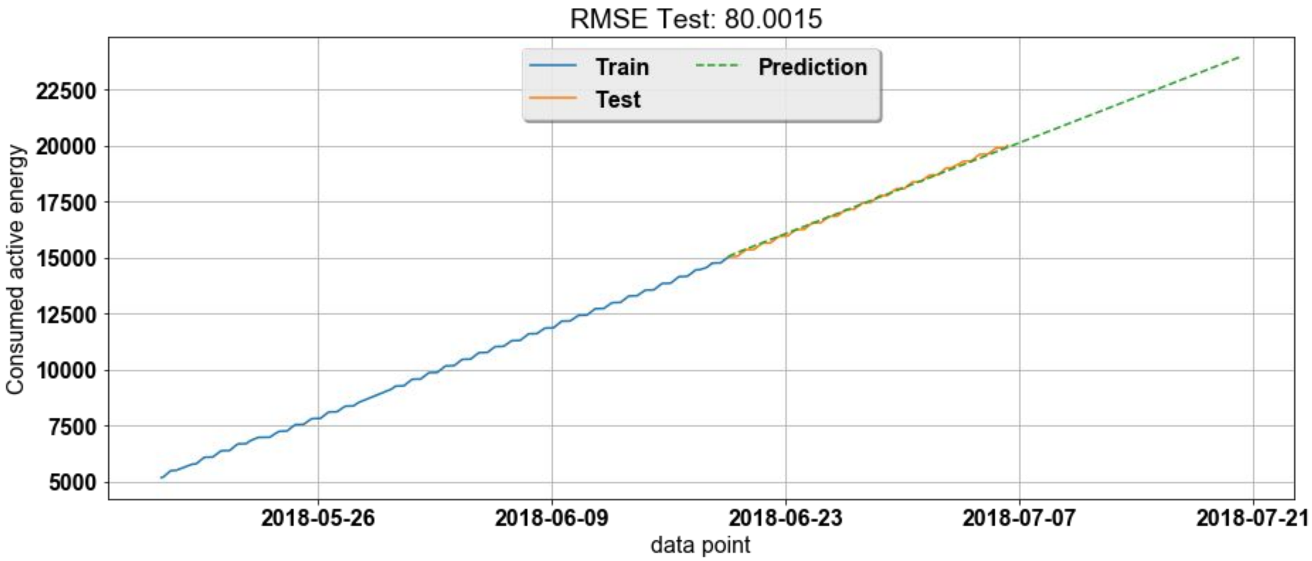
Seasonal ARIMA Model

-incorporates both non-seasonal and seasonal factors in a multiplicative model.

$$ARIMA(p, d, q) \times (P, D, Q)S,$$

with p = non-seasonal AR order, d = non-seasonal differencing, q = non-seasonal MA order, P = seasonal AR order, D = seasonal differencing, Q = seasonal MA order, and S = time span of repeating seasonal pattern.

Prediction with auto ARIMA



git clone https://github.com/cchantra/time_series.git

Or goto https://github.com/cchantra/time_series

Download each file to your desktop

And run in jupyter

https://medium.com/@chantrapornchai/energy-consumption-prediction-with-auto-arima-66e530a3f673

KU
KASETSART
UNIVERSITY

HPCNC