# Temus case study

Natalino Busa

May 23 , 2022

# The Case

**Background**:

Meeting our energy and environmental needs without compromising the ability of future generations to meet their own needs, is a core focus for Temus, Temasek and the global community. While this is a grand challenge which will take many decades to achieve, it is important we focus on early and practical steps.

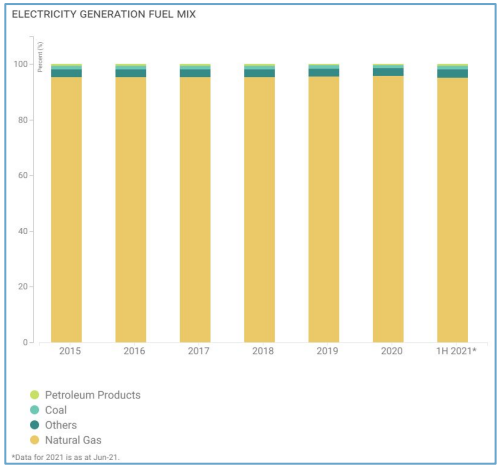
**Problem Statement:**

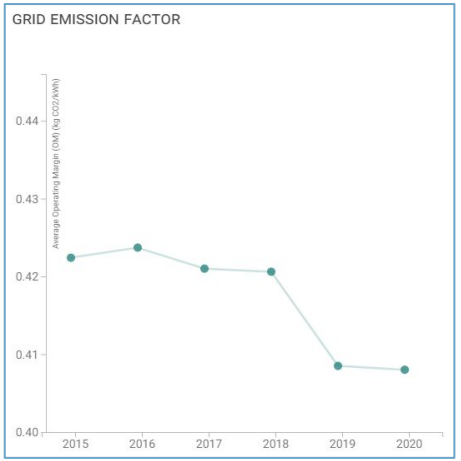
For this assignment, we would like you to identify and demonstrate an opportunity to deploy machine learning to take a small but practical step towards increasing sustainability and reducing environmental impact.

# Predict Load and Renewable Energy Sources

## Opportunity

Singapore is already using almost entirely the cleanest of non-renewable energy sources (Natural Gas: 95% of total consumption for energy production). However other sources of renewable energy such as wind and sun are becoming increasingly important. https://www.nccs.gov.sg/singapores-climate-action/power-generation/



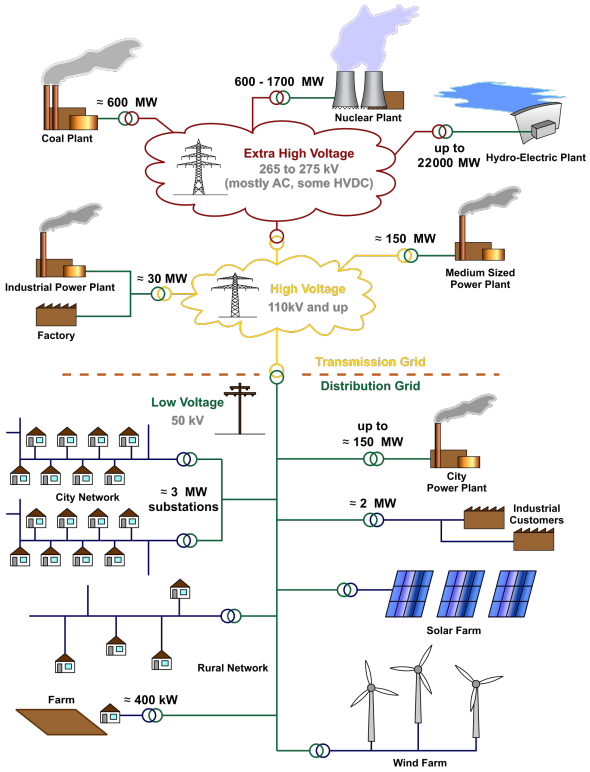



# Predict Load and Renewable Energy Sources

**Challenges**

Adding renewable sources comes with a number of challenges:

- Stability of the Grid
  - Manage overproduction of energy
  - Storage of Energy (short term: flywheel, batteries)
  - Local Grids (Malesia, Indonesia)
  - Inertia management
- Energy Production
  - Accurate forecast of production (renewable: sun, wind)
  - Throttling planning of non-renewable plans
- Energy Consumption
  - Accurate forecast of consumption
  - Smart demand (connected to the grid management)



# Data sources

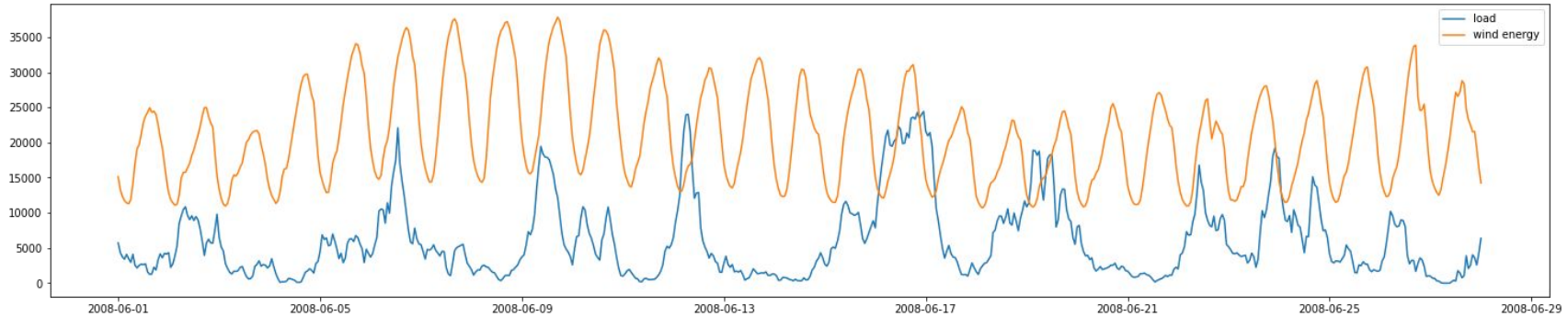
**From Kaggle**

Global Energy Forecasting Competition 2012 - Wind Forecasting
https://www.kaggle.com/competitions/GEF2012-wind-forecasting

Global Energy Forecasting Competition 2012 - Load Forecasting
https://www.kaggle.com/competitions/global-energy-forecasting-competition-2012-load-forecasting

# Variable renewable energy

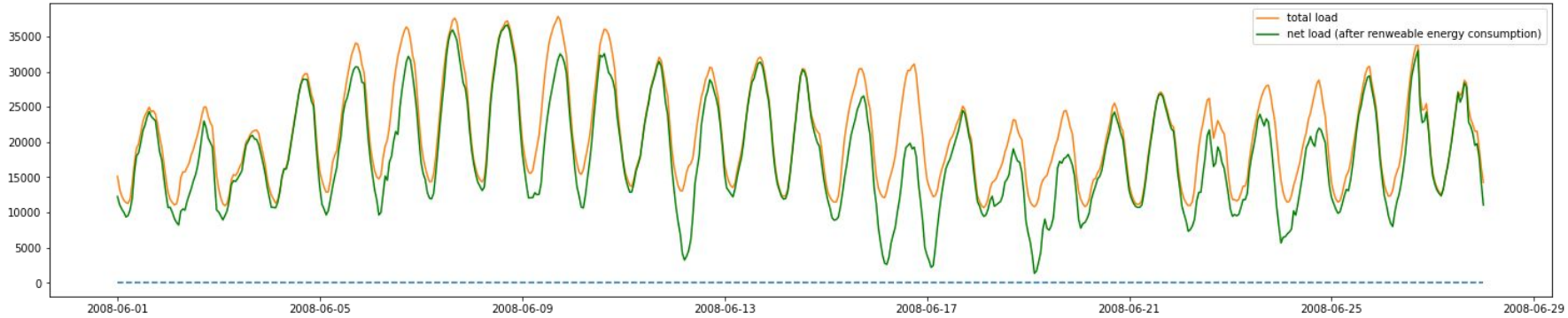
By considering the first two data sources from Kaggle, renewable energy provides energy at the cost of additional volatility and reduced inertia in the grid:



# Variable renewable energy

For a saving of 15% on traditional natural gas combustion, we must excel in the following:

- Load forecasting
- Renewable Energy forecasting
- Oversupply forecasting
- Inertia forecasting (plant ramp-up, cool-down, throttling)

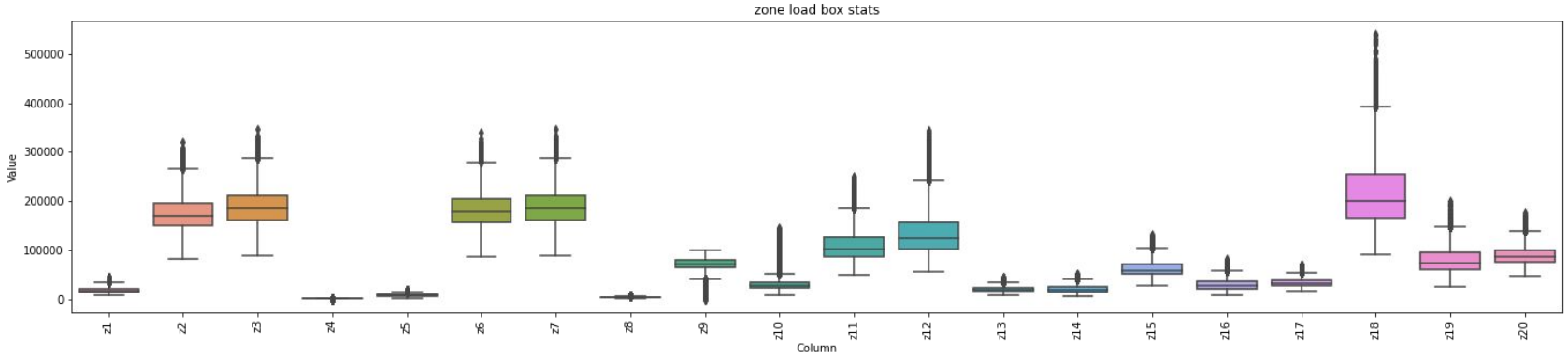


# Load Forecasting: approach

- **Exploratory Data Analysis**
- **Preprocessing**:
  - Filling Gaps
  - Anomaly Detection/Correction
- **Feature engineering**
  - Featurize time
  - Rolling and Expanding stats
- **Training**
  - 3-fold (tumbling) temporal cross validation
  - Dataset preparation for multi-step forecasting
  - Direct forecasting
  - Recursive forecasting
  - Collect Metrics
- **Model Selection**
- **Forecasting**
- Evaluation and next steps

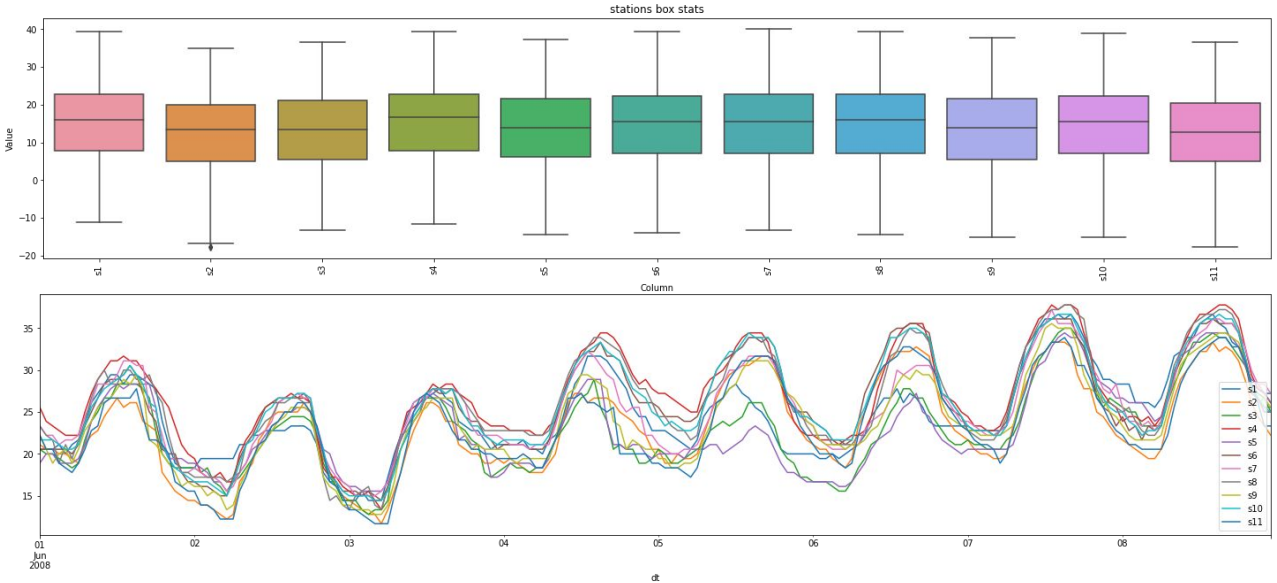
# Load Forecasting

For load forecasting, we are going to use temperature stations (11 stations ) as forecasting variables to predict the energy consumption (load) of 20 geographical zones.  The temperature stations exhibits large variation in temperature swing. The load consumed by the 20 zones varies significantly from zone to zone.



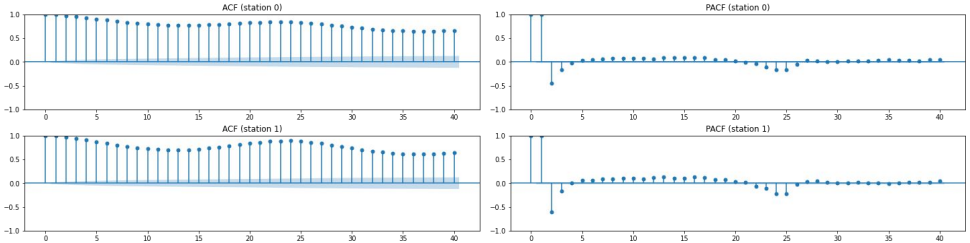
# Temperature Stations

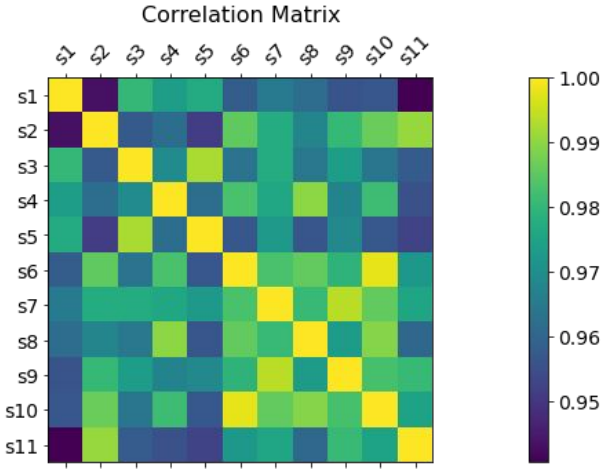
For load forecasting, we are going to use temperature stations (11 stations ) as forecasting variables to predict the energy consumption (load) of 20 geographical zones.



# Correlation: Stations

Temperatures signals are highly correlated with each other. Which is reassuring as it demonstrates that the data is reliable.

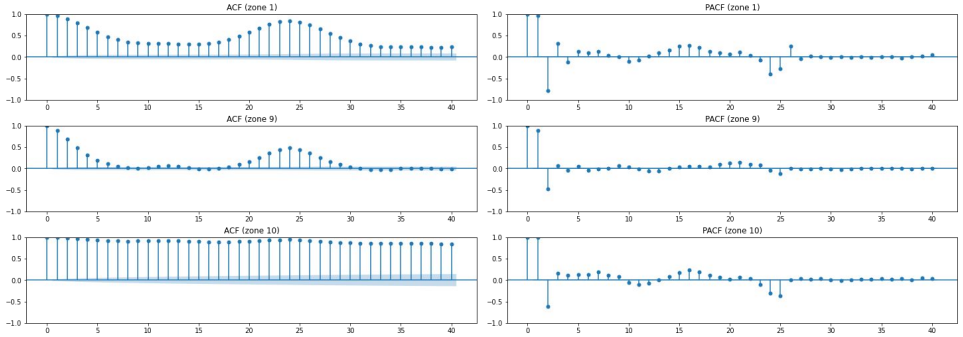


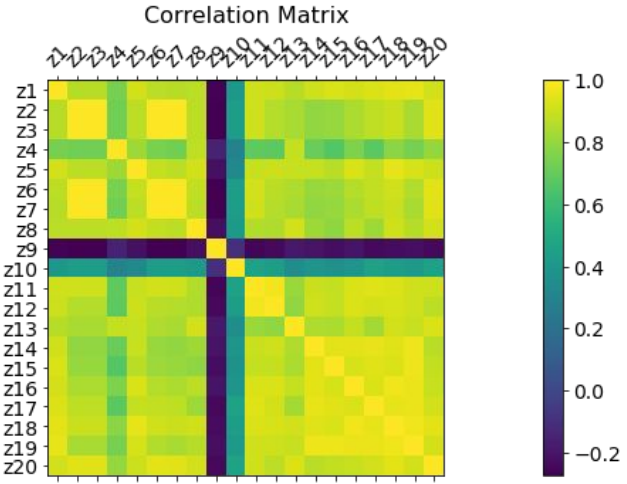




Temperatures present a typical AR pattern (autoregressive, with 24 sample seasonality)

# Correlation: Zones

Most zones highly correlate with each other in terms of load, with the exception of zone 9 and 10

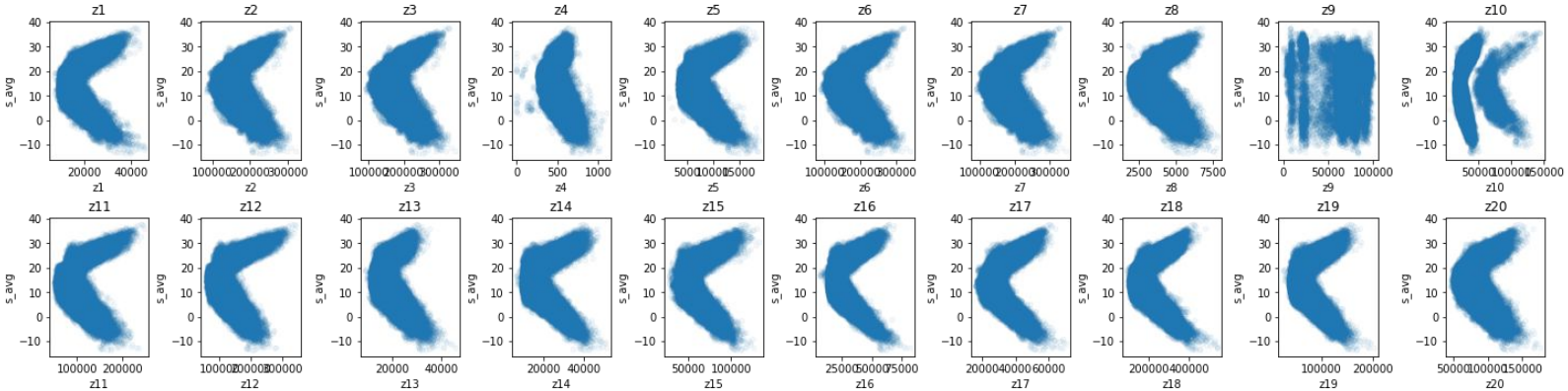



The load pattern is more complex, with both MA and AR components and seasonality 24 H

# Correlation: Zones vs Temperatures

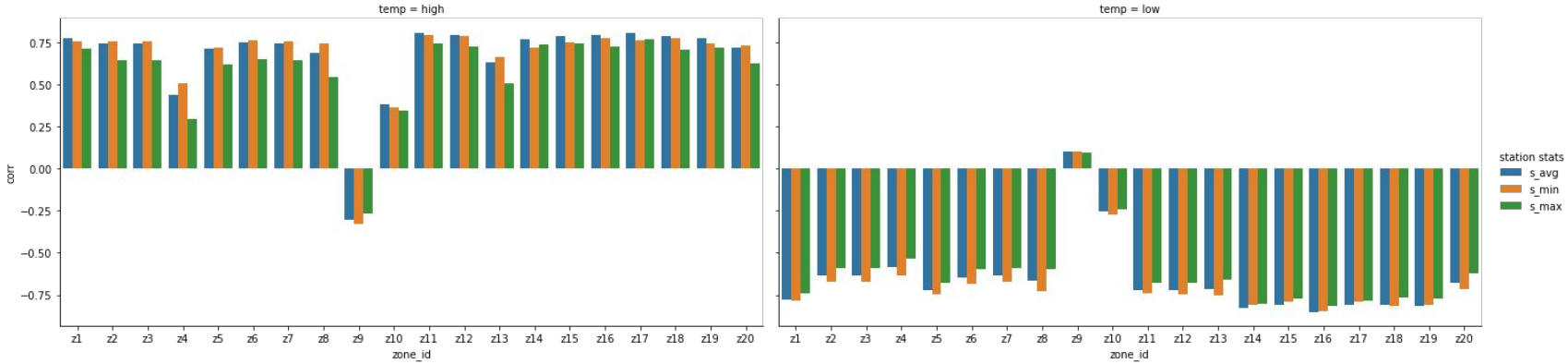
Correlation display a sort of a '<' or boomerang shape, which can be broken in two patterns for high temperature and low temperatures. As the load is minimum around 15 deg Celsius.




Intuitions: Energy consumption increases at low and high temperatures. Zone 9 seems industrial, as it does not depend on temperatures) and zone 10 exhibit a double pattern (maybe residential plus industrial). The smudge could indicate an increased energy consumption over the years to the right.

# Correlation: Zones vs Temperatures

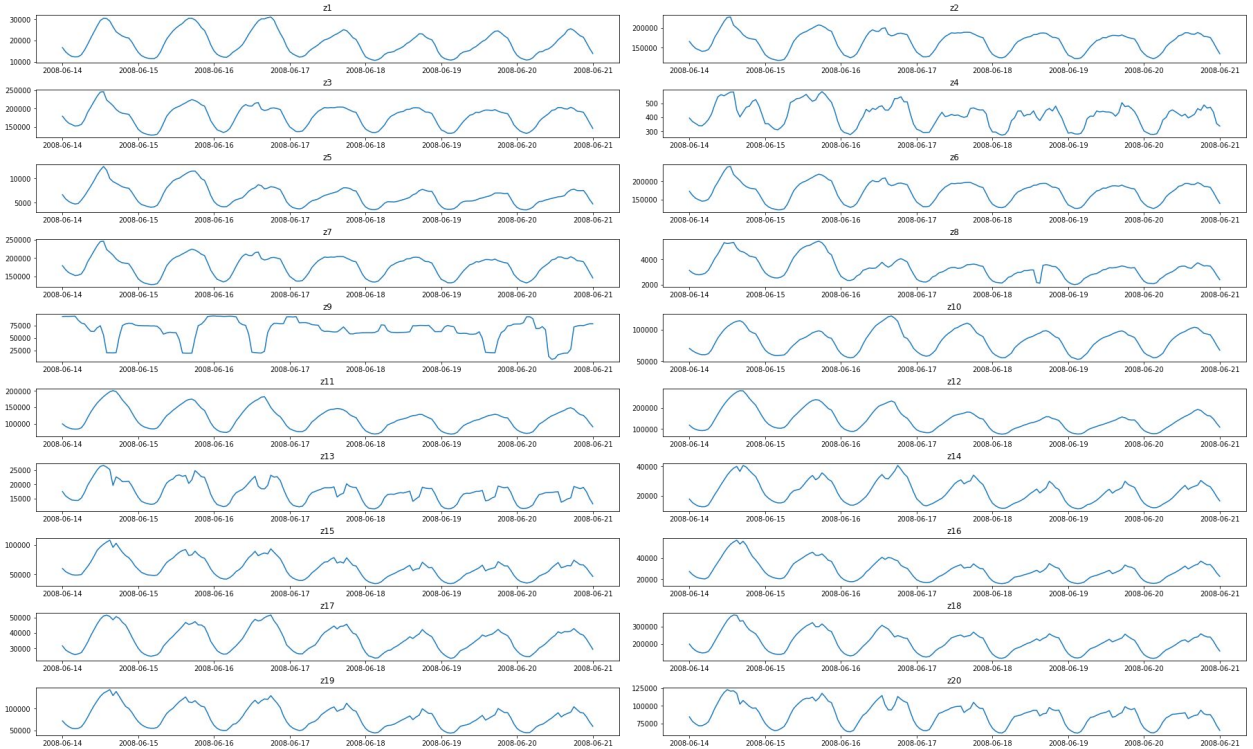
Correlation display a sort of a '<' or boomerang shape, which can be broken in two patterns for high temperature and low temperatures. As the load is minimum around 15 deg Celsius.




After breaking down low and high temperature consumptions, we detect a quite strong correlation between load and temperatures. This suggests that temperature is a good predictive variable for the forecasting

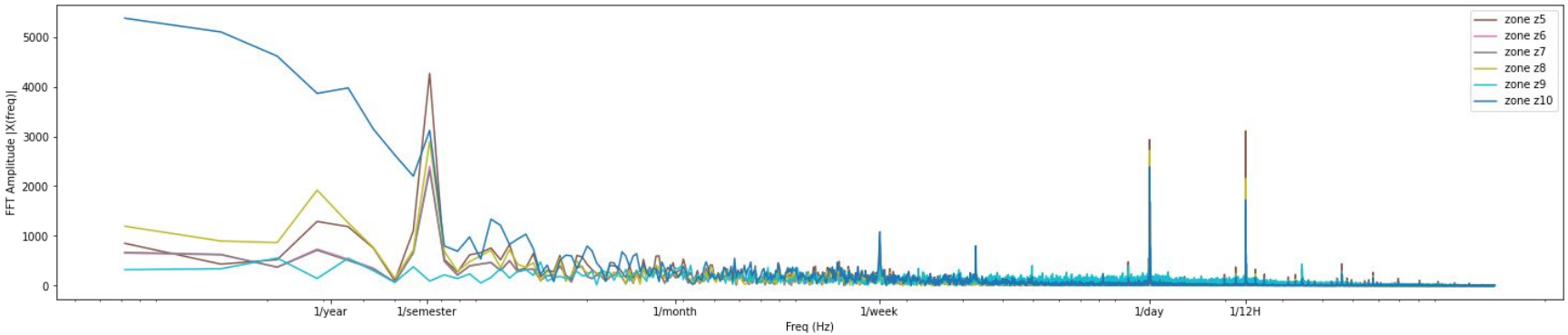
# Zones: Time series

Load is mostly regular.
In most zone it exhibit a high load 8am-10pm.

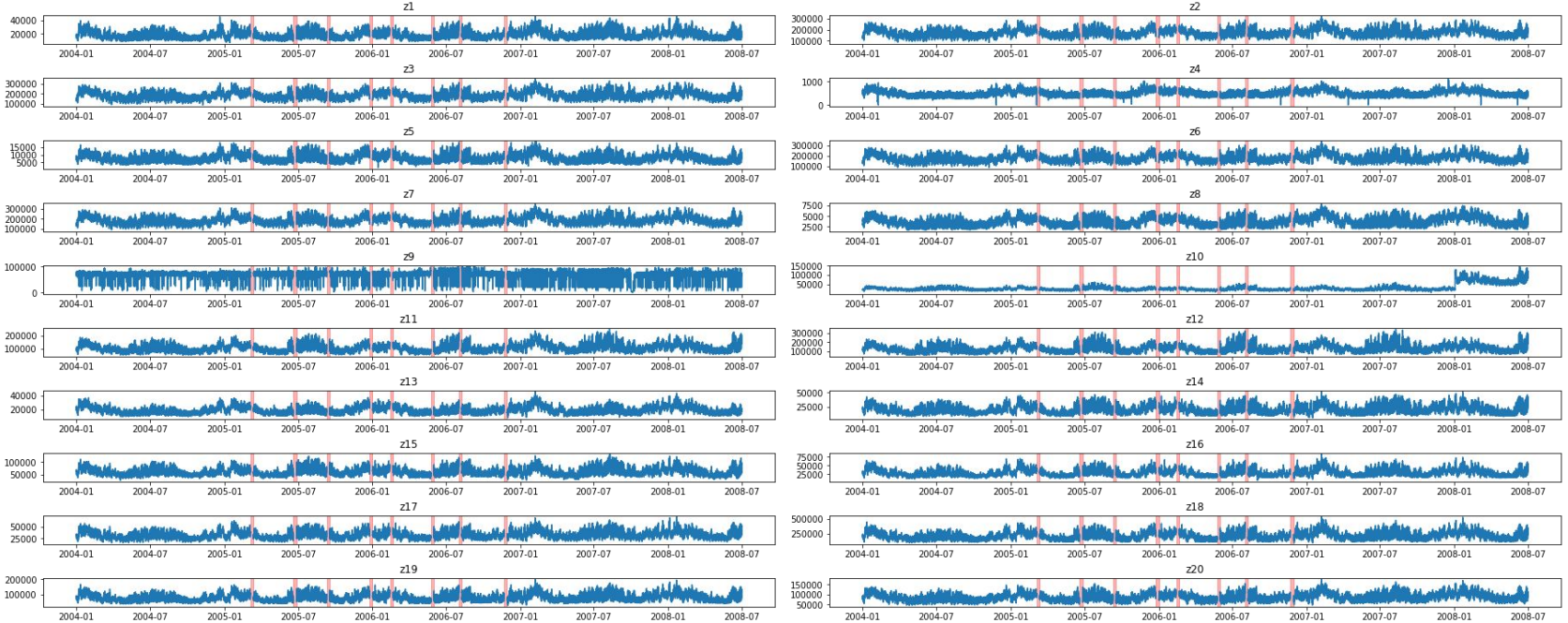


# Zones: Time series

Frequency analysis: Peak at day, week, and semester.
The 6 month peak is probably due to high energy consumption both for col and hot weather.



# Preprocessing: Time series filling




Some filling is required as not all forecasting methods can deal with NA/NaN
When missing: data is filled with a 24H lag value sNaive(s=24)

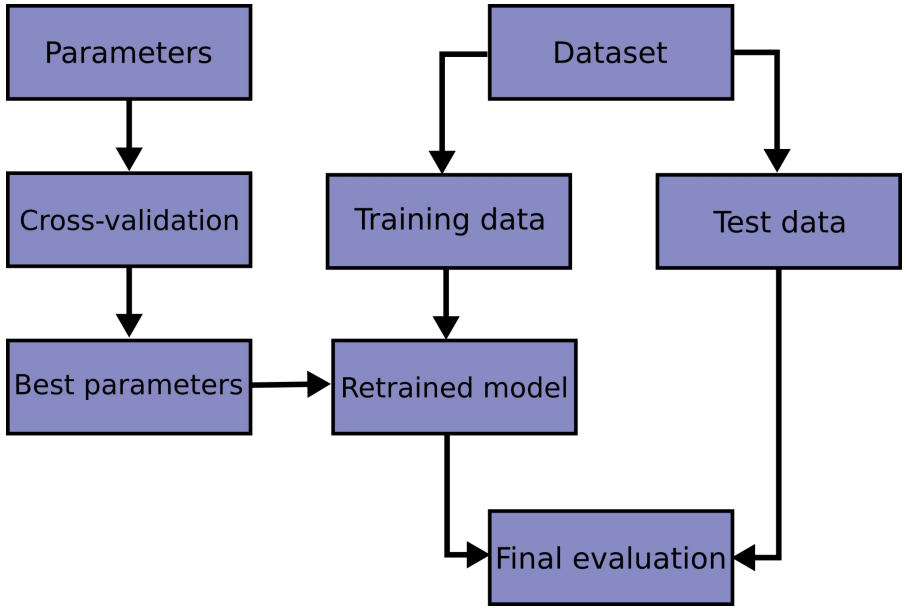
# Solution: training

The training involves three steps

- Parameter tuning in cross-validation
- Full model training on test data
- Final evaluation on test data

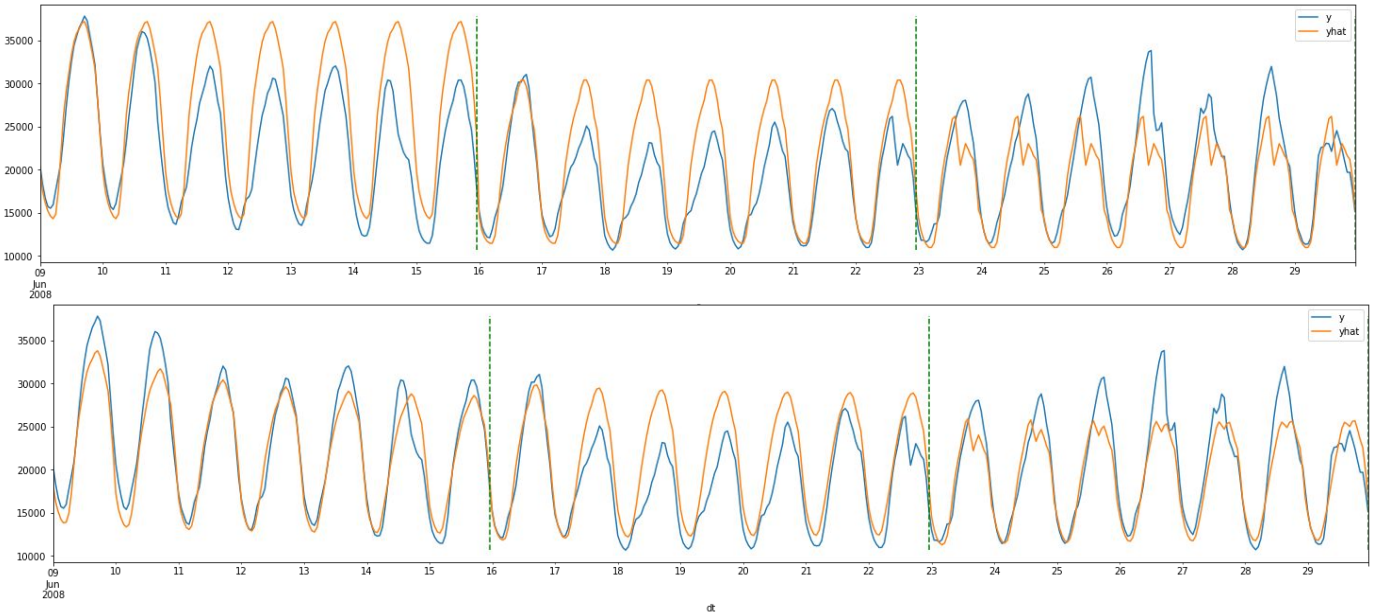
Learning the parameters of a prediction function and testing it on the same data is a methodological mistake: a model that would just repeat the labels of the samples that it has just seen would have a perfect score but would fail to predict anything useful on yet-unseen data. This situation is called **overfitting**.

During the last part, model drift is measured from previous model, when the model is operationalized.



# Forecasting: (Recursive models)

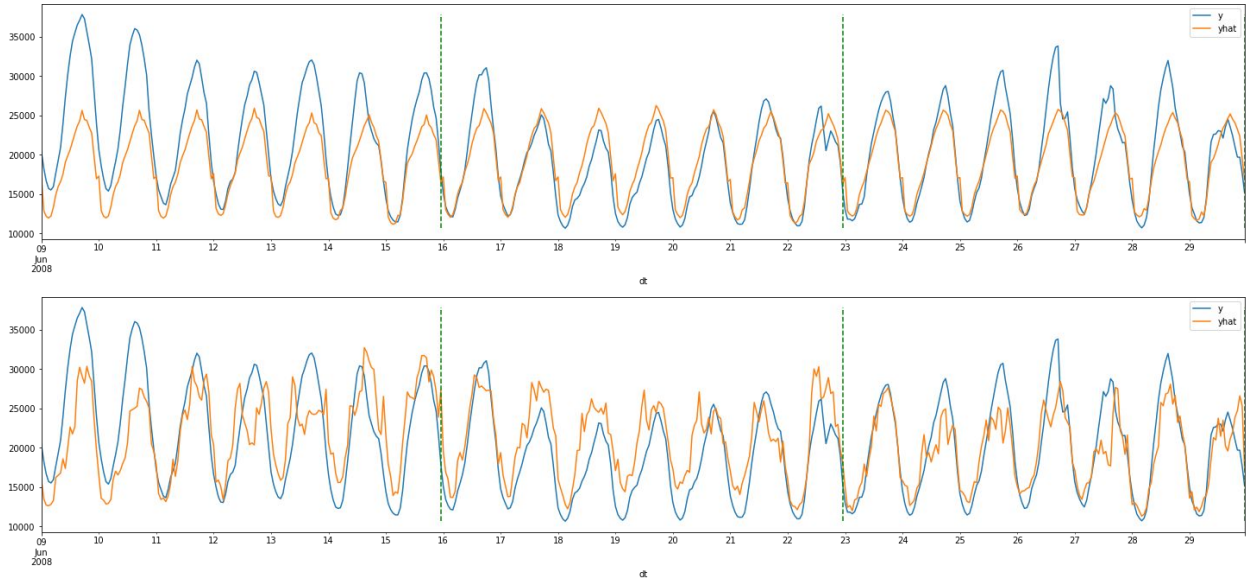
sNaive(24), sARIMA((0,0,0), (1,1,124)



# Forecasting: (Direct models: ML)

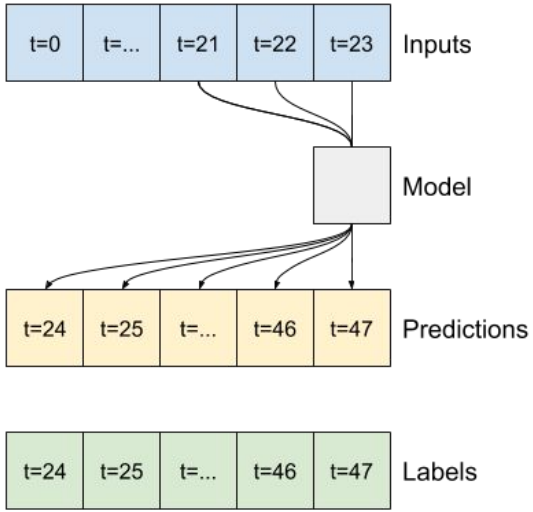
1) LGBM avgs
2) LGBM all signals with auto LGBM tuning

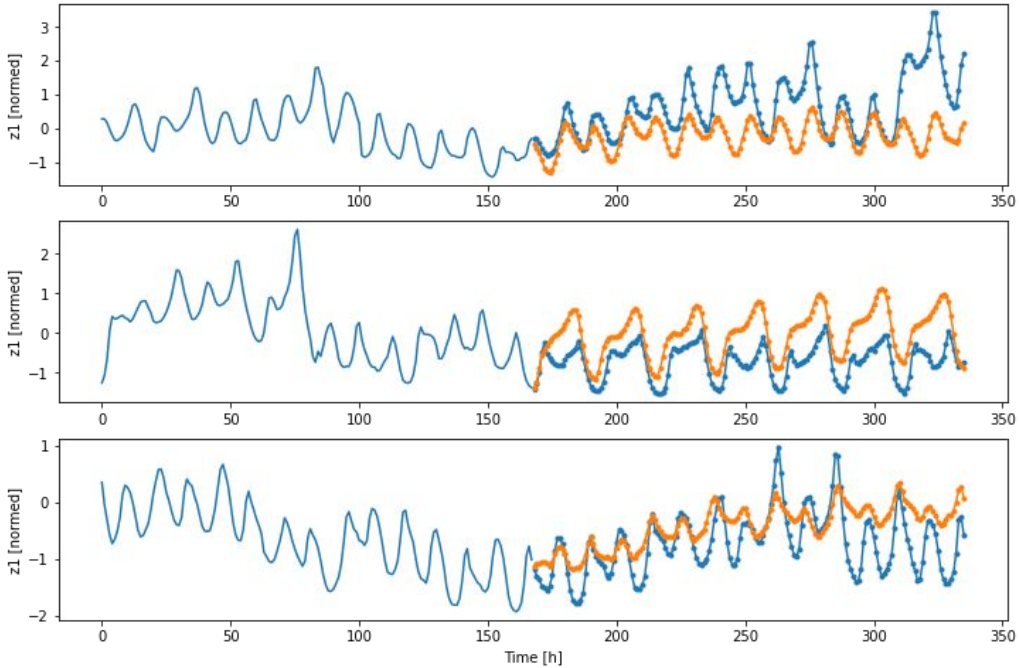
Some overfitting seems to happen on the bigger model



# Forecasting: (Direct models: Tensorflow/Keras)

CNN, 1D convolution on 44 features



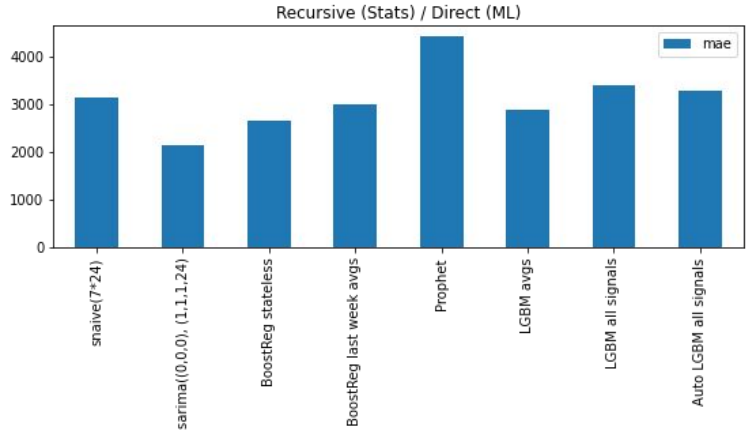



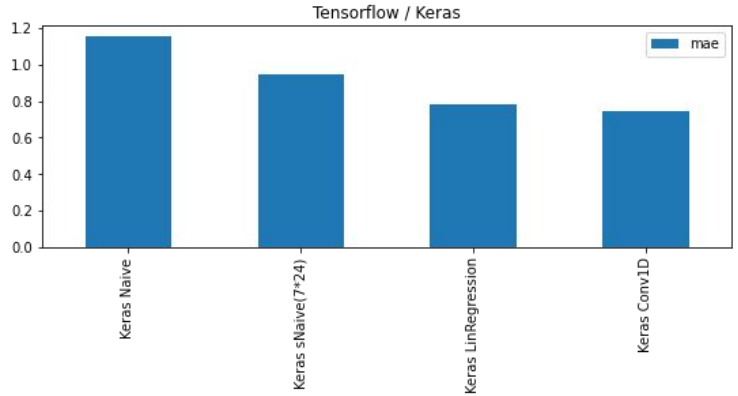

Note: the trained model uses 7*24=168 Inputs and predicts the next 168 samples

# Solution: metrics comparisons

The models are trained on a 3-fold temporal cross-validation for model selection and hyper-parameter tuning. This ensure:

- Less bias on specific data
- A fair comparison of models
- Better tuning of model parameters



# Solution: model selection

Current best: **sarima**

```
Mean Absolute Error     (MAE) : 2131.85 +/- 523.82
Root Mean Squared Error (RMSE): 2737.56 +/- 609.71
Mean Absolute % Error   (MAPE): 0.10 +/- 0.04
```

Best next candidates: '**LGBM avgs**'

```
Mean Absolute Error     (MAE) : 2879.69 +/- 733.82
Root Mean Squared Error (RMSE): 3677.08 +/- 1030.73
Mean Absolute % Error   (MAPE): 0.14 +/- 0.03
```

A possible follow-up could be to augment LGBM

... direct approach but separate models for each step
- create a model separately for each prediction step (168 models)

... with a recursive approach (autoregressive methodology)
- feed forecasted model and forecasted regressors to the next model

# Next steps

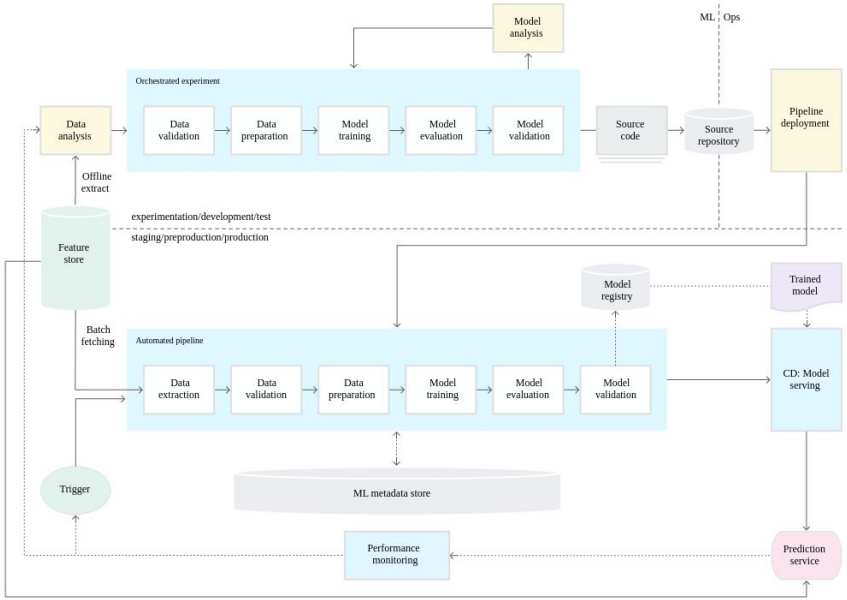
**Include more data sources:**

From Singapore Gov:
https://data.gov.sg/dataset/climate-change-and-energy-green-vehicles
https://data.gov.sg/dataset/building-energy-performance-data
https://www.kaggle.com/datasets/truecue/worldsustainabilitydataset

**Build an operational MLops Forecasting system:**
Create a pool of reliable estimators / trained models. Automate the process of promoting a research model to production.

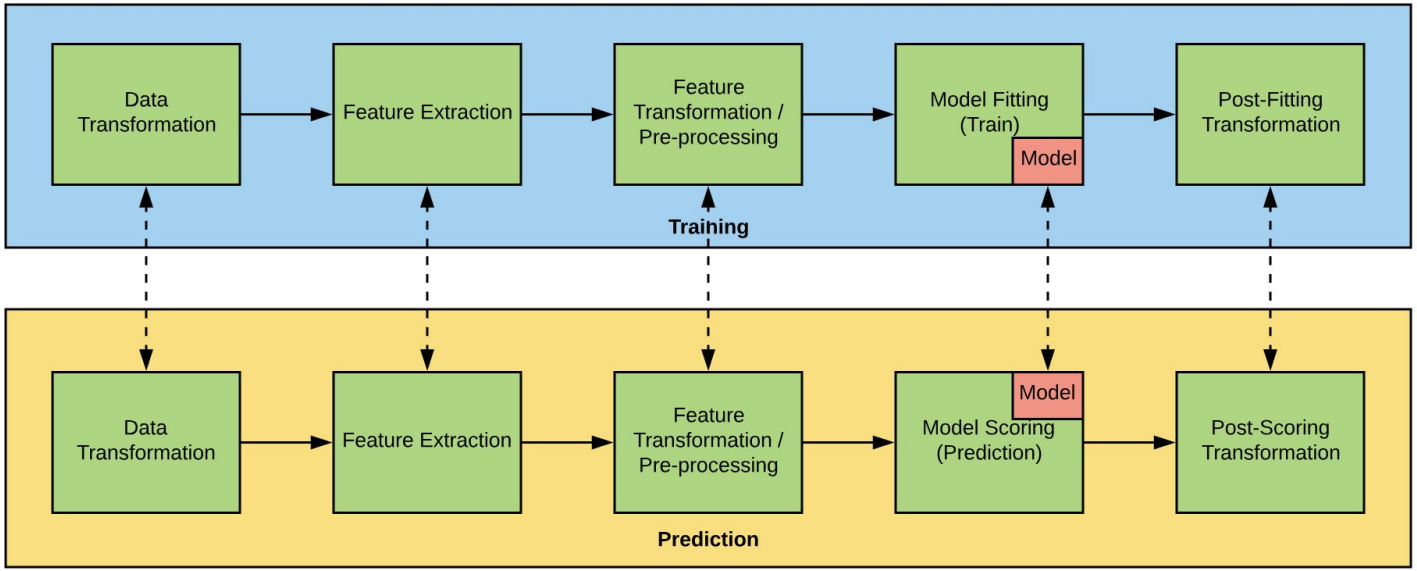
Focusing on streaming forecasting.
Measure model and data drift and bias over time.
Create a model store for MLOps recovery / serving

**Decision Science and man-in-the-loop**
Use the forecasting to drive optimization and actions.
Provide triggers for the smart grid to reduce energy volatility, to save energy storage costs and to increase production inertia.




Credit: level 2 MLOps level 2: CI/CD pipeline automation Google)
https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

# Proposed ML pipeline



# (some) References

**Forecasting Times Series:**
https://www.tensorflow.org/tutorials/structured_data/time_series
https://www.statsmodels.org/stable/tsa.html
https://facebook.github.io/prophet/

**Temporal Cross validation:**
https://scikit-learn.org/stable/modules/cross_validation.html#time-series-split
https://otexts.com/fpp3/tscv.html

**Direct vs Recursive Forecasting**
https://machinelearningmastery.com/multi-step-time-series-forecasting/
https://www.kaggle.com/tbierhance/perils-of-recursive-forecasting
https://www.r-bloggers.com/2018/01/direct-forecast-x-recursive-forecast/

**AutoML:**
https://microsoft.github.io/FLAML/