Short Term Power Burn Model

Gas Analytics

Rationale:

Python:

- Utilise more sophisticated time series forecasting techniques to try and improve forecast accuracy.
- Automate forecasts to allow for earlier model runs.

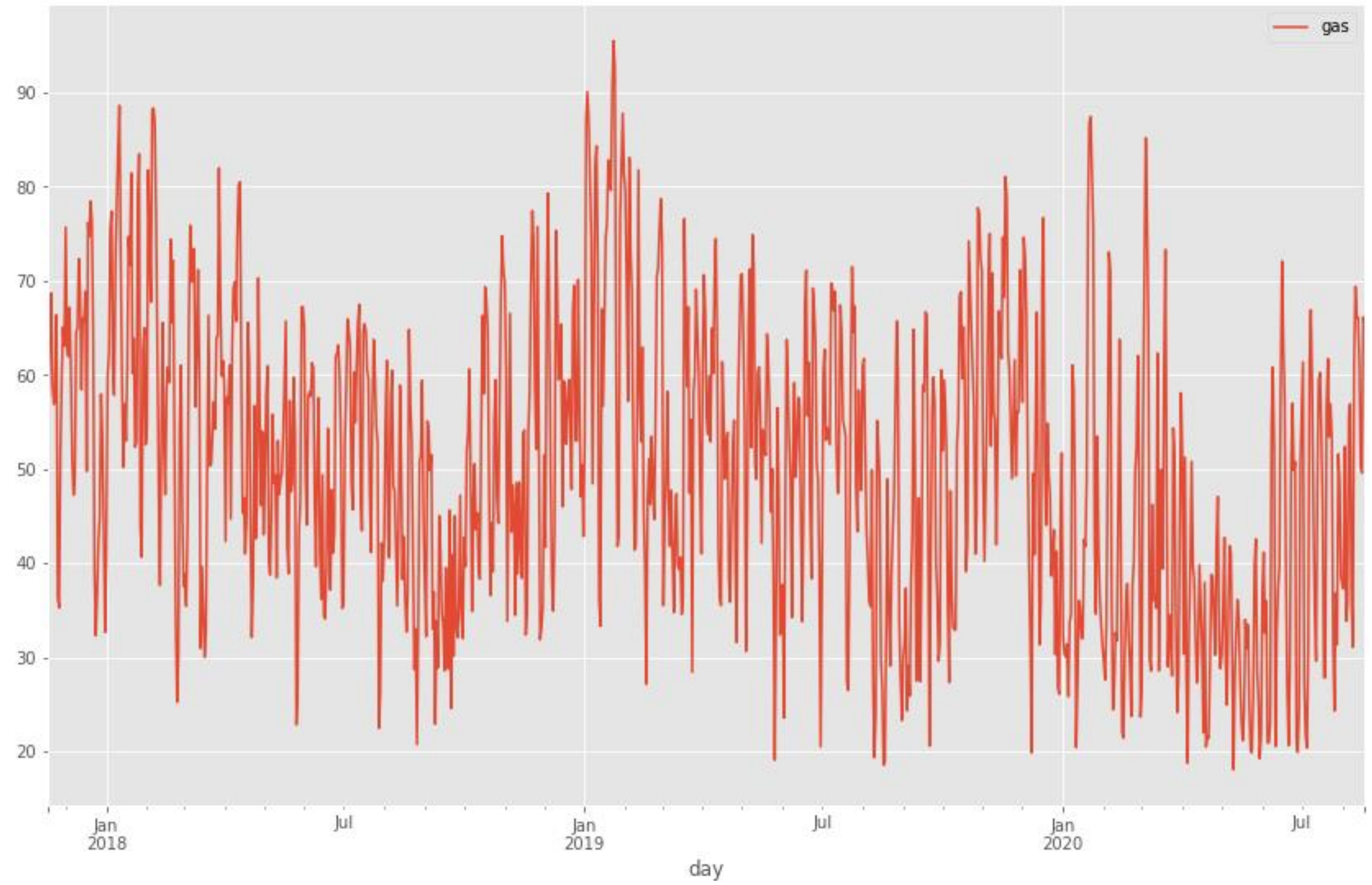
Creating Framework :

- How to approach forecasting within a framework
- Defining model accuracy

Short Term Power Burn:

- Understand what drivers short-term gas burn.
- Provide the foundations for longer term (seasonal) forecasting.

Dependent Variable

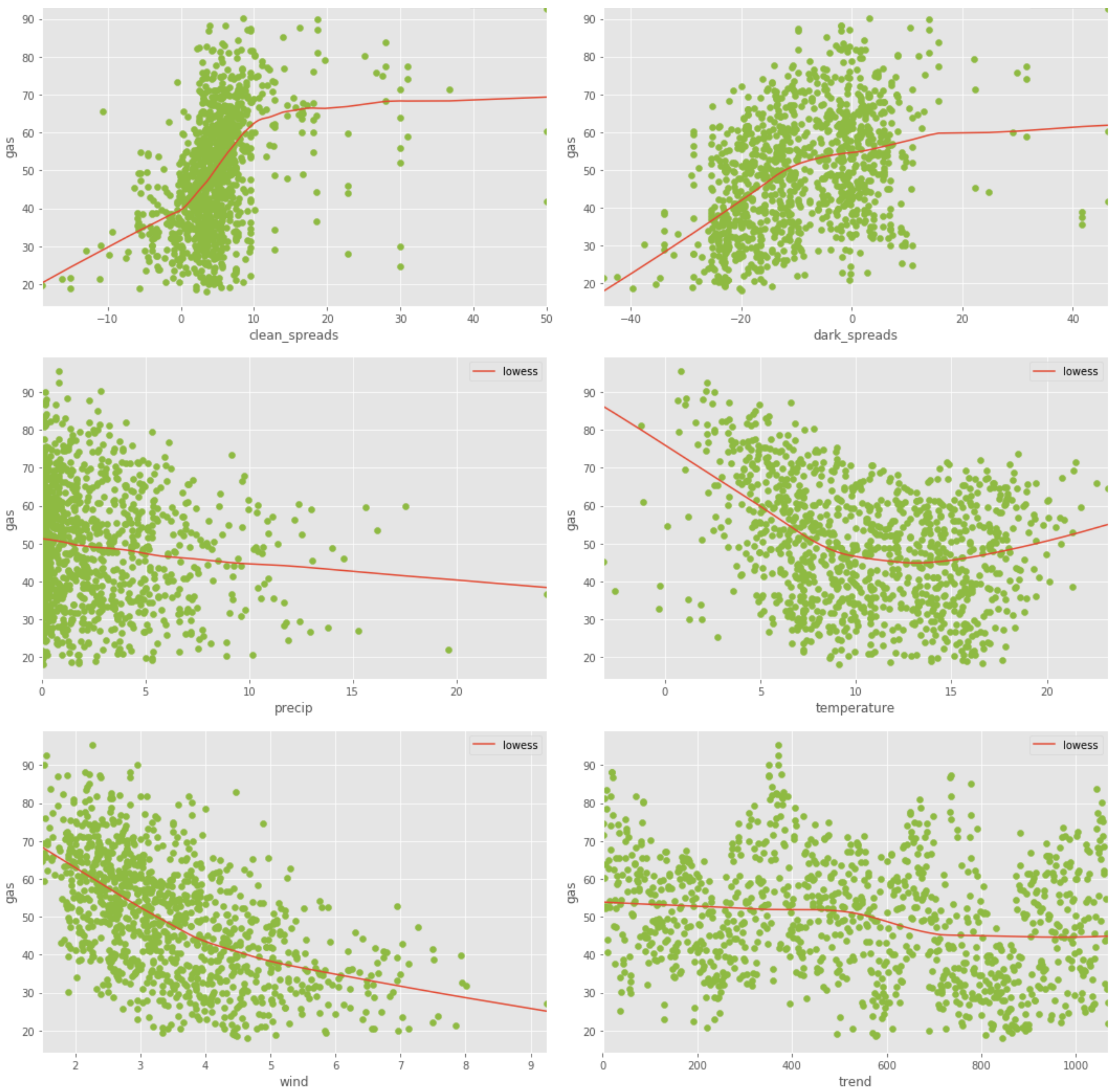


Covariates:

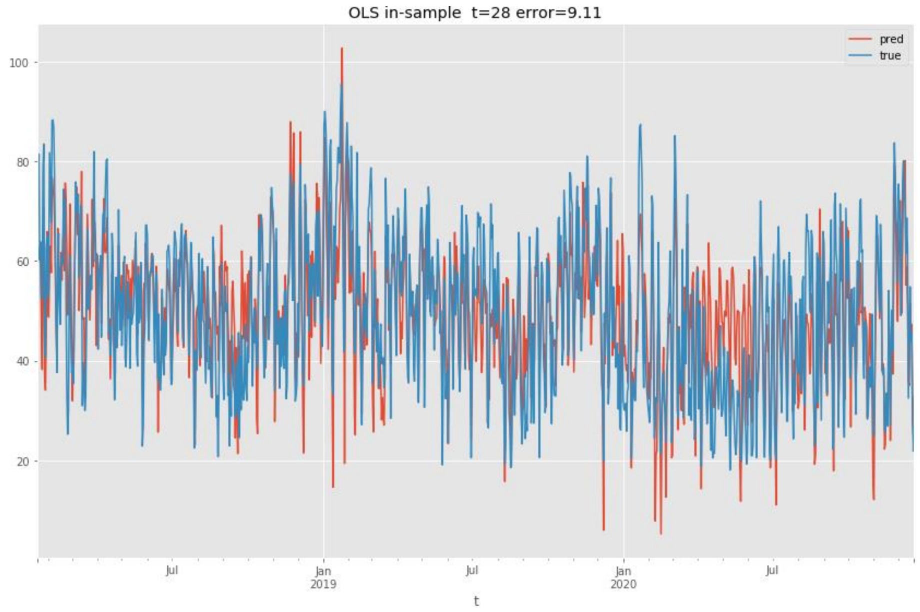
- clean spreads
- dark spreads

- temperature
- wind
- precipitation

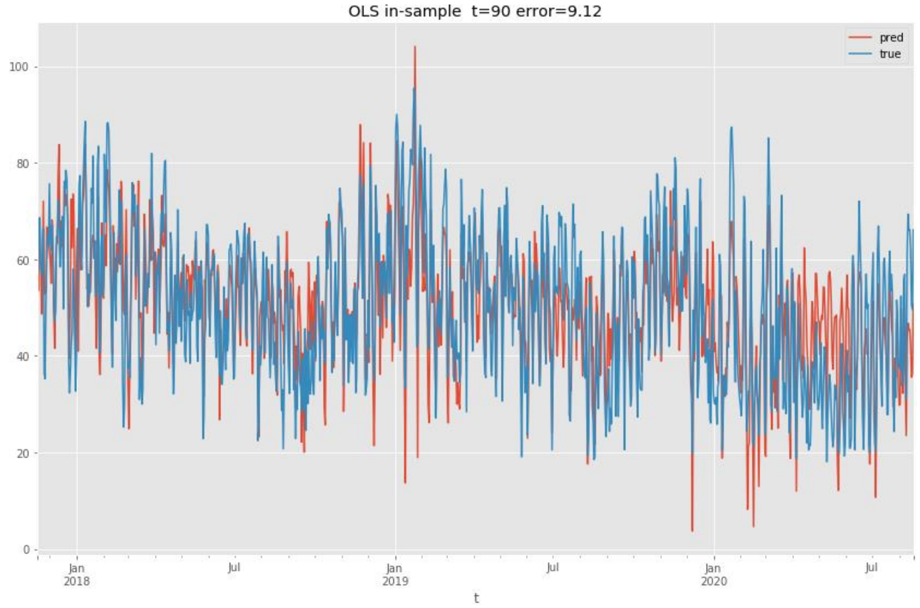
- monday_thursday flag
- Fourier series

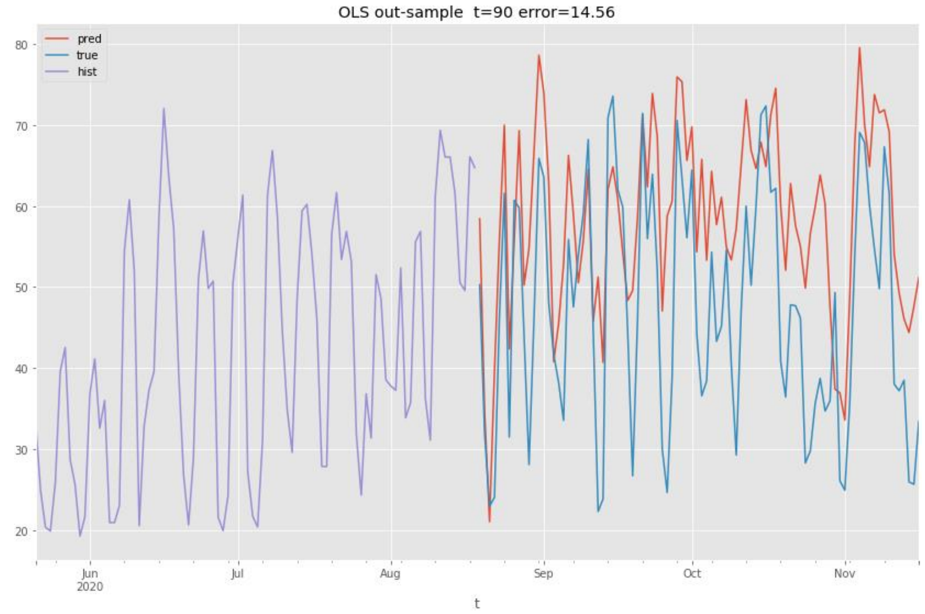


Calibrator (exog): Linear Regression



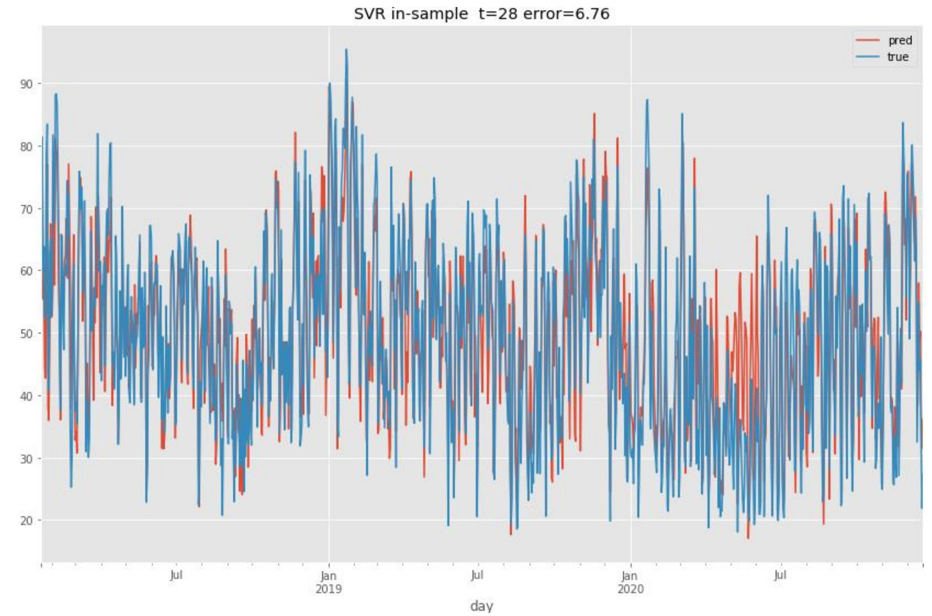
CV= 9.43 (2.08)

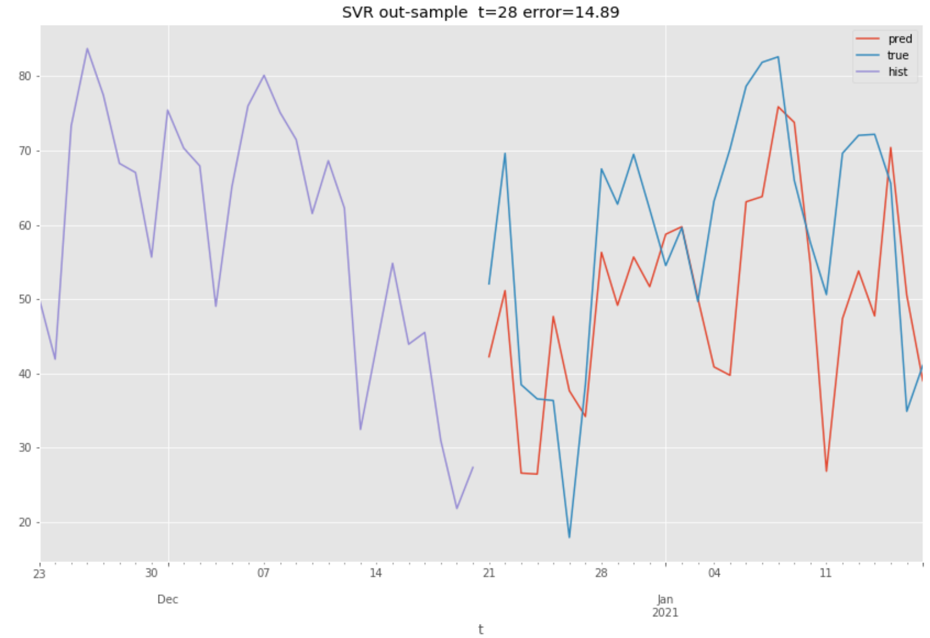


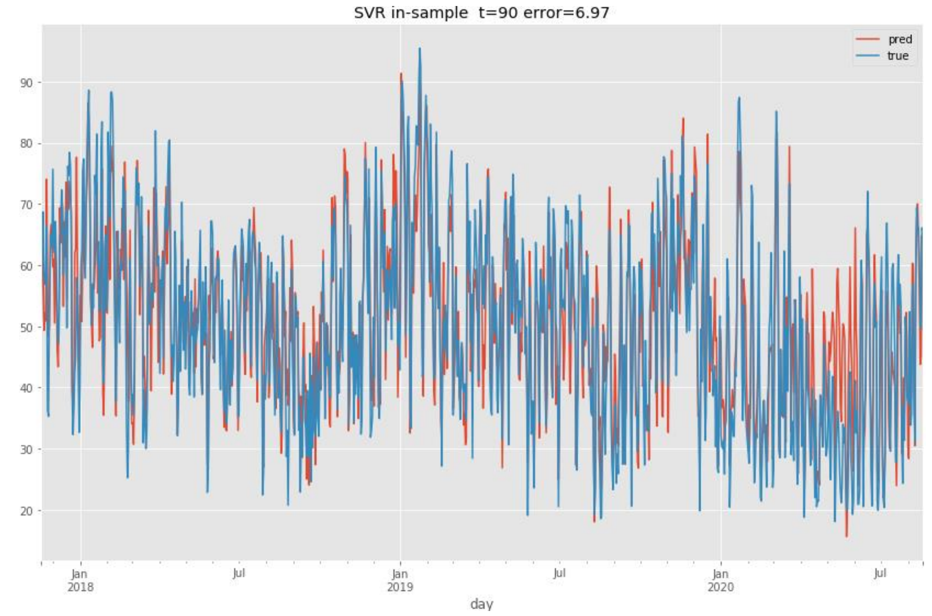


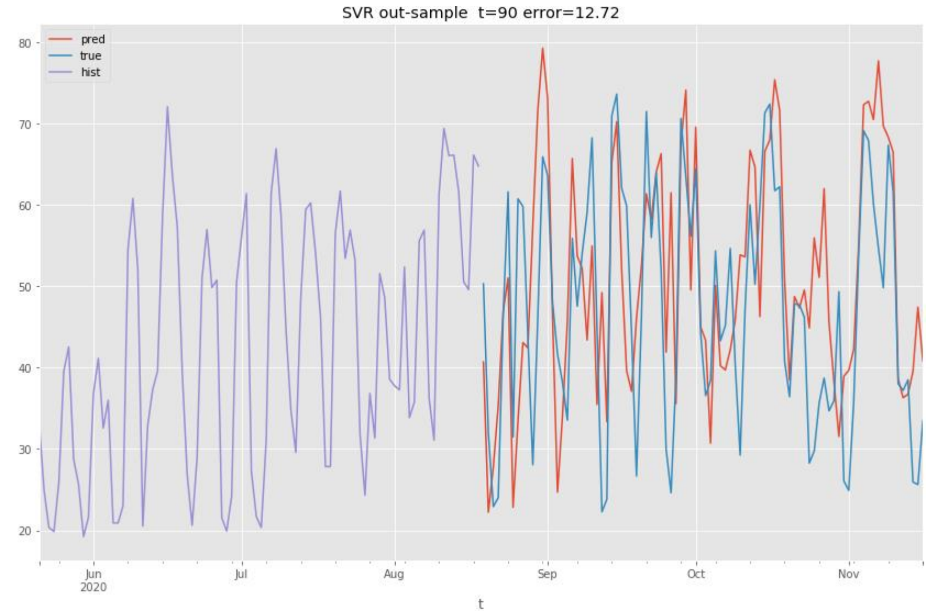
CV= 10.57 (1.79)

Calibrator (exog): Support Vector (kernel) Regression

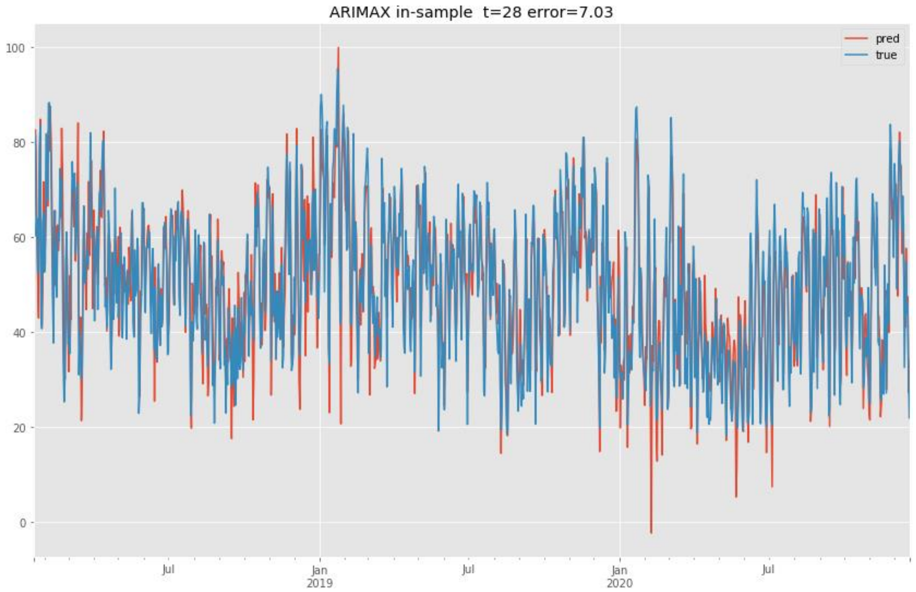




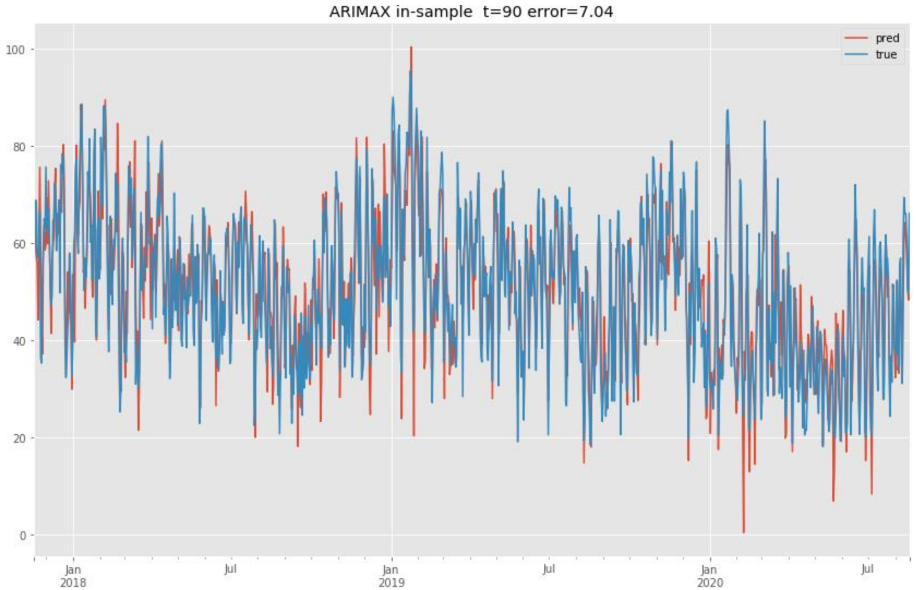


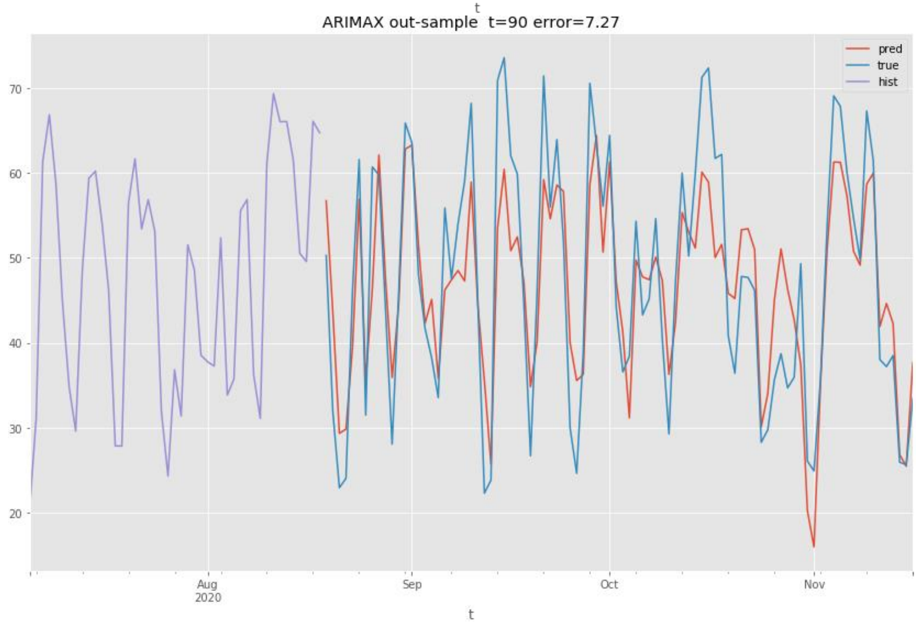


Calibrator (exog): SARIMAX



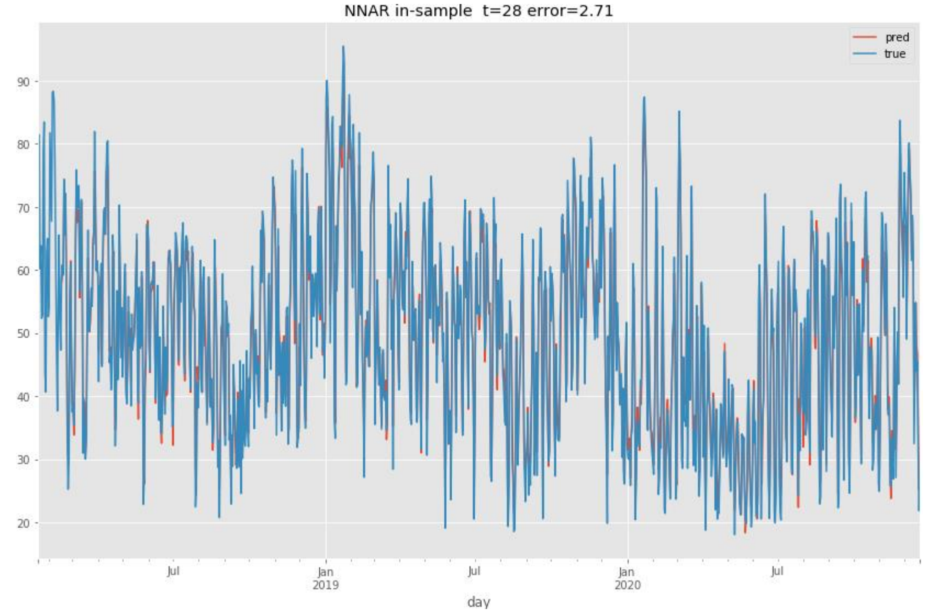
CV= 8.44 (1.71)

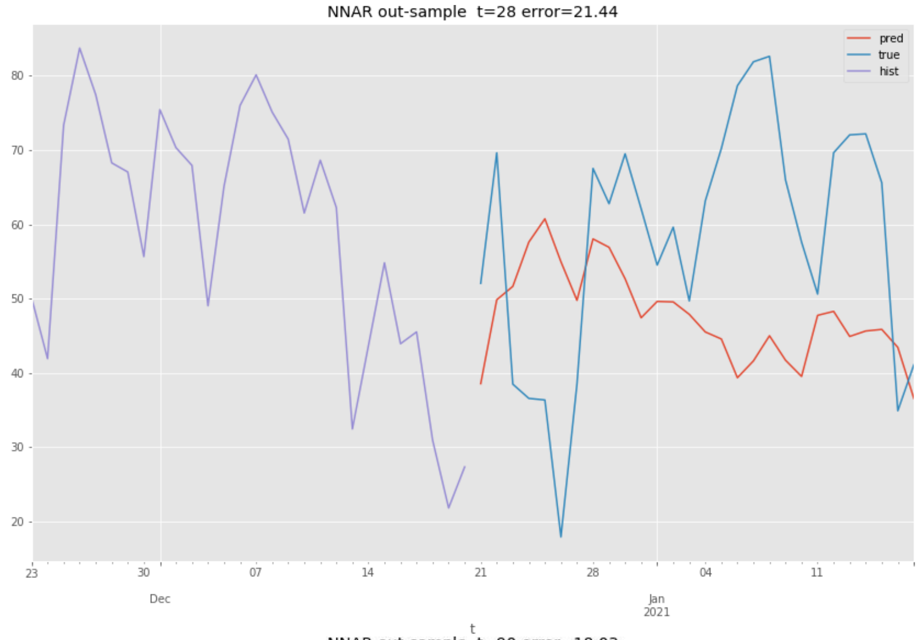


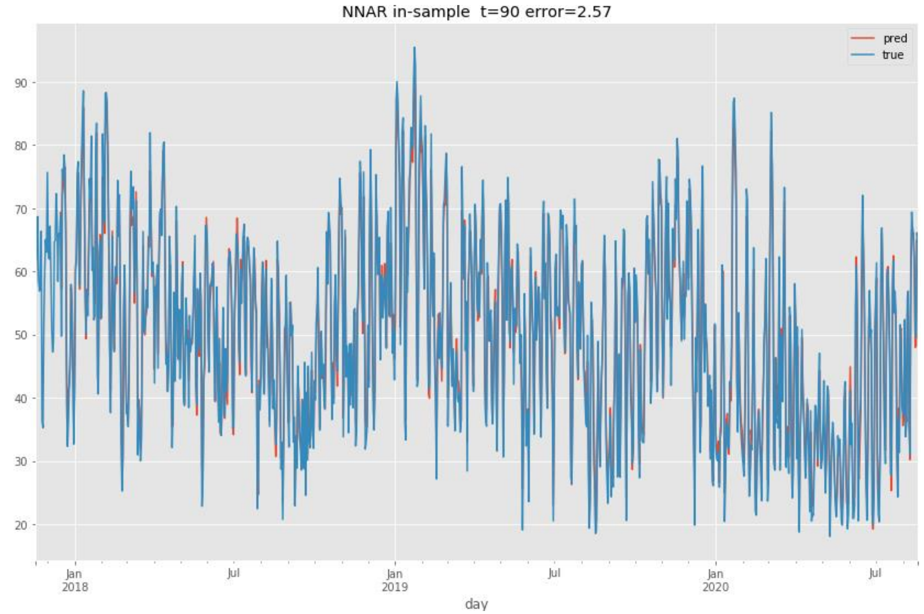


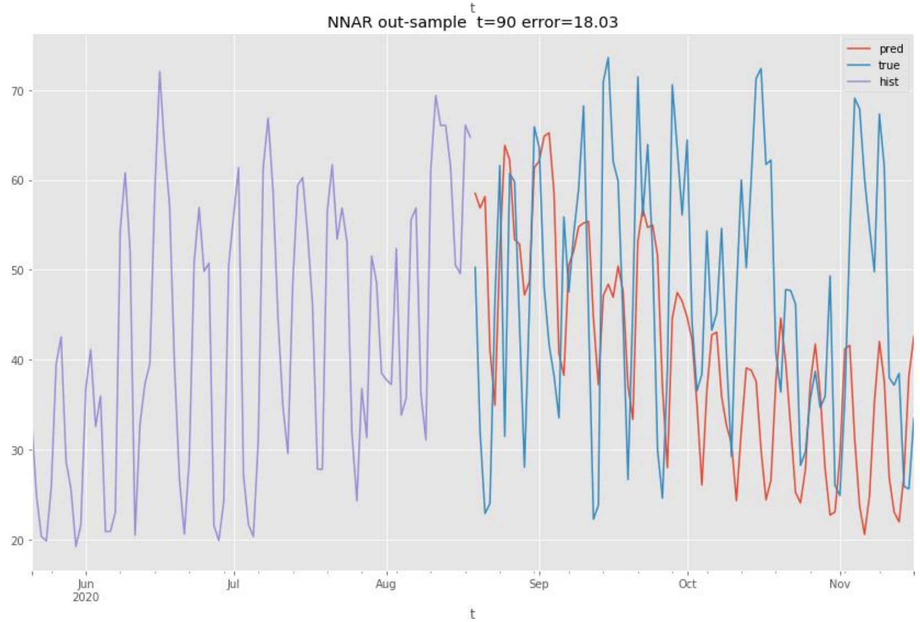
CV= 10.37 (2.41)

Calibrator (endog): Neural Network Autoregression

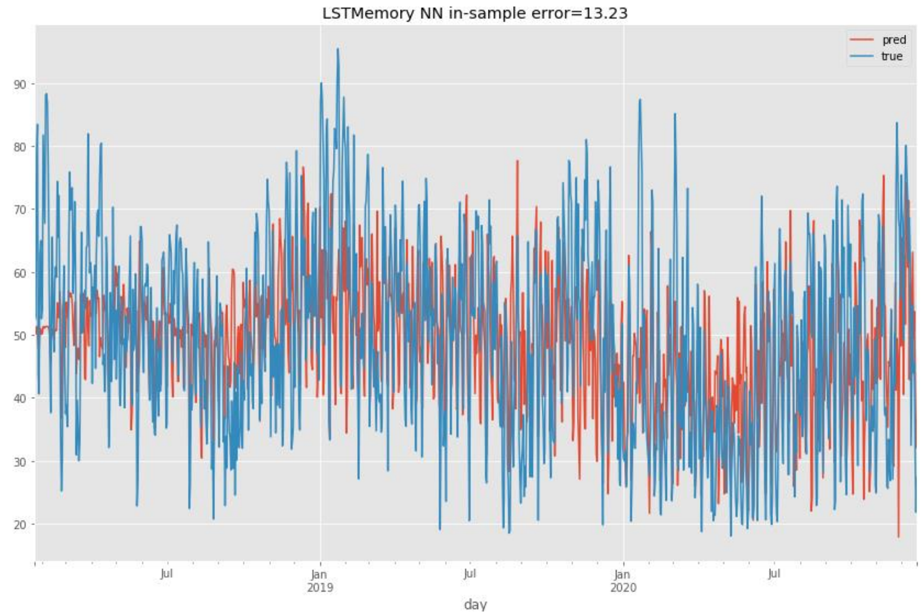


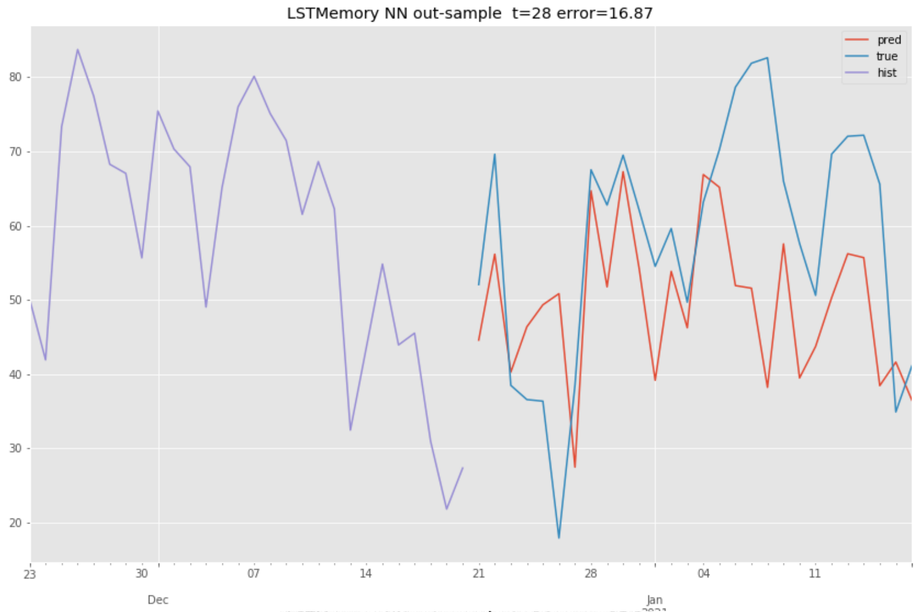


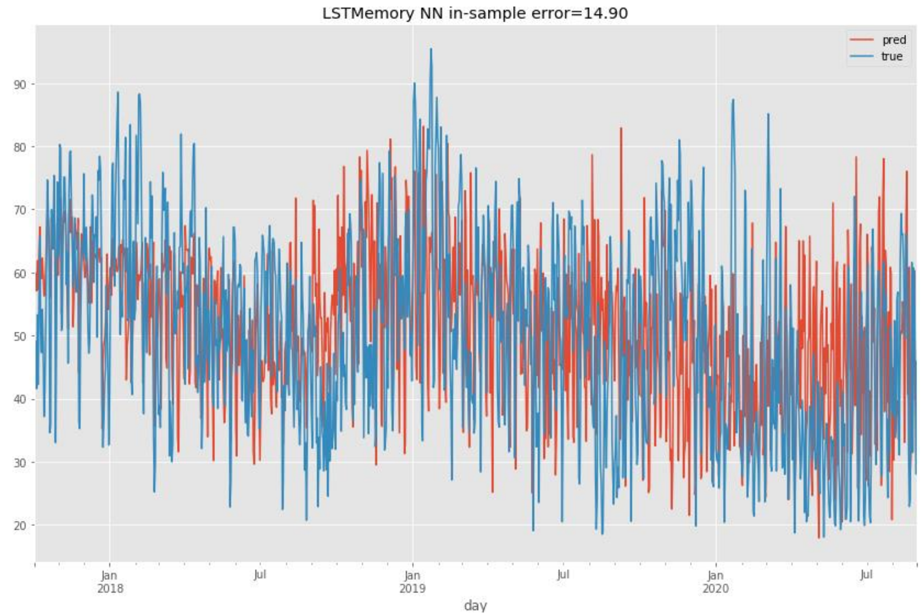


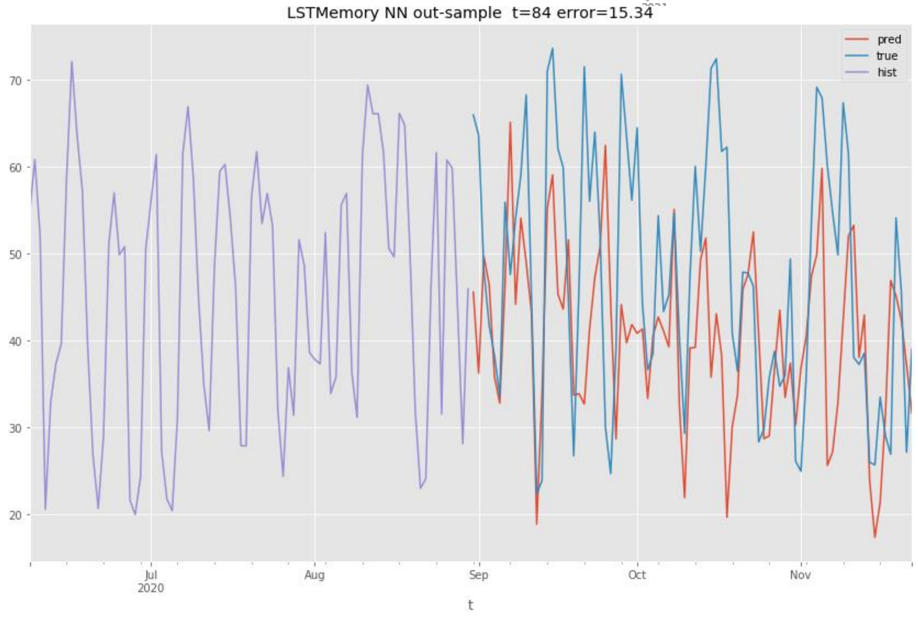


Calibrator (endog): Neural Network Long Short Term Memory









Appendix A:

Model	Parameters	Description
EXOGENEOUS RELATIONSHIP:		
Linear Regression	const, trend, clean_spreads, precipitation, temperature, wind, S1-7, C1-7, S2-7, C2-7, S3-7, C3-7, mo_th_yes	Modelling in levels Exog variables: grid search with BIC criterion 3 terms of Fourier series at weekly frequency
SARIMAX	SARIMAX(1, 0, 1)x(1, 0, 1, 7) const, dark_spreads, precipitation, temperature, wind,	Modelling in levels Specification: grid search with BIC criterion Exog variables: grid search with BIC criterion
Support Vector Regression	type="eps-regression" kernel='radial' • cost= 8 • gamma= 0.0625 • epsilon= 0.3	Dependent variable and features scaling: standardization Specification: grid search with 10-fold CV
ENDOGENEOUS RELATIONSHIP:		
Neural Network Autoregression	• Model: NNAR(29,1,15)[7] • Average of 20 networks, each of which is a 29-15-1 network with 466 weights options were - linear output units	Dependent variable scaling: standardization
Long Short Term Memory Neural Network		

Appendix B:

<table>
<tr><td colspan="9">ALL DATA: IN TIME : ACTUALS 4y</td><td>OUT OF TIME: FUTURE</td></tr>
<tr><td colspan="8">IN SAMPLE 3y</td><td>OUT SAMPLE 28days</td><td rowspan="6"></td></tr>
<tr><td colspan="8">CROSS VALIDATION 3y:</td><td rowspan="5"></td></tr>
<tr><td colspan="2">FOLD1 1y</td><td colspan="2">FOLD2 1.5y</td><td colspan="2">FOLD3 2.5y</td><td colspan="2">FOLD4 3y</td></tr>
<tr><td>train sample 0.8y</td><td>test sample 28days</td><td>train sample 1.3y</td><td>test sample 28days</td><td>train sample 2.3y</td><td>test sample 28days</td><td>train sample 2.8y</td><td>test sample28 days</td></tr>
<tr><td></td><td>error 1</td><td></td><td>error 2</td><td></td><td>error 3</td><td></td><td>error 4</td></tr>
<tr><td colspan="8">CV error = 1/4 * (error1 + error2 + error 3 + error4)

CV std = Standard Deviation (error1, error2, error3, error4)</td></tr>
<tr><td colspan="8">IN SAMPLE ERROR</td><td>OUT SAMPLE ERROR</td><td>OUT OF TIME ERROR</td></tr>
</table>

error = Root Mean Square Error

Appendix C:

Modelling Framework

Design Matrix:

- Imputations
- Design Matrix:

$$t \mid y_t \xrightarrow{f} v_t, \mid \; x_t^1,\; x_t^2,\; \dots\; x_t^k \xrightarrow{g} g_1 x_t^1,\; g_2 x_t^2, \dots, g_k x_t^k$$

$$\mathrm{DM}_t = \mathrm{DM}_t(y_t, f, f^{-1}, \{x_t^i\}_{i=1,k}, \{g_t^i\}_{i=1,k})$$

Exploratory analysis

- Autocorrelation:
  - $\mathrm{ACF}(x_t)$
  - $\mathrm{PACF}(x_t)$
- Scatter Plots:
  - y_t next to v_t and x_t^k next to $g_k x_t^k$
  - y_t vs. x_t^j for $j = 1, k$ with LOWESS for dependency shape analysis
  - x_t vs. x_{t-h} with LOWESS for autocorrelation analysis
  - y_t vs. x_{t-h}^k for given k with LOWESS for lagged-leading relationship

Calibrator:

$$\mathrm{C}(\mathcal{H}yper Params) \to \mathrm{C}$$

Model:

$$\mathrm{M} = \mathbb{M}(\mathrm{C},\ \mathrm{DM}) \to \{\hat{\theta}_l\}_{l=1,m}$$

Model Specification

$$\{\hat{\theta}_l\} \xrightarrow[\text{GridSearch}]{I(\theta):\ AIC,\ AICc,\ BIC} \{\hat{\theta}_l^*\}$$

Model Selection:

Cross Validation

$$\mathrm{CV} = \mathbb{CV}(\mathrm{M}, \text{Partitioning},\ \text{Performance Metric})$$

$$\mathrm{C} \xrightarrow{\varepsilon_{CV}} \mathrm{C}^*$$

Residuals Diagnostics:

$$\hat{\varepsilon}_t = v_t - \hat{v}_t;\ \hat{\varepsilon}_t^{std};\ \hat{\varepsilon}_t^{stu}$$

$$\mathrm{RD} = \mathcal{RD}(\hat{\varepsilon}_t, \hat{\varepsilon}_t^{std}, \hat{\varepsilon}_t^{stu})$$

Forecast:

$$\mathrm{C}^*(DM_{t+1} = g_i(x_{t+1}^i)_{i=1,k} \mid \{\hat{\theta^*}_l\}_{l=1,m}) = \hat{v}_{t+1} \xrightarrow{f^{-1}, y_t} \hat{y}_{t+1}$$