# Mid-term generation forecasting at DER level (1 week ahead) – KONCAR

## Rationale & Link to BEYOND Apps

The mid-term energy generation prediction at DER level is very important for forecasting and planning the maintenance of building assets. In traditional usage tied to distributed energy resources, mid-term forecasting is commonly used for regular maintenance scheduling, so the cost of interrupted operation is minimized. Shorter term forecasts are typically used in market participation. However, at the building level mid-term forecasts may be useful in e.g. regime change for installed equipment in the building.

Effective mid-term generation forecasting is imperative for the establishment and continuous validation of energy management strategies and maintenance plans, at building level. By using mid-term forecasts of DER generation, the building manager can minimize the energy and comfort impact of scheduled maintenance.

The mid-term generation forecasts will be available in the BEYOND AI Analytics Toolkit. The dedicated AI analytic with feed with valuable insights the Self-consumption optimization features of the Digital Twin environment (BEPO application), and the personal energy analytics PEASH application.

## Overview of relevant implementations

As stochastic renewable resources, the DER production forecasts are largely impacted by the prime mover behavior – wind in case of wind power and insolation in the case of photovoltaics. Therefore, the meteorological forecasting is expected to play an important role.

For the (very) short-term forecasts (below 6 hours of forecast horizon), the methods usually focus purely on statistical analysis of the production from the DER power plants. This is due to strong autocorrelation characteristics [1] [2]. Beyond 6 hours of forecasting horizon, using numerical weather forecasts as inputs to a DER forecasting system becomes beneficial. Essentially, predicting the DER output at a becomes a task of, at the time *t,* estimating an *actual* power at the time *t+k*, based on the *forecasted* value of input features such as wind speed or insolation.

The process of predictor training relates to the configuration of the weights and other predictor inputs, so that the error measure between actual values and forecasted values is minimized. In effect, the well-trained predictor both learns the relation between the input feature (insolation or wind speed) as well as compensates for the NWP predictor errors (the difference between the forecasted values of the input features and the actual, observed values).

From the above, it follows the mid-term energy generation forecasting can be viewed as a time series problem that is dealt with several approaches coming from classical statistics [2] (i.e. linear and non-linear models) and artificial intelligence (AI). The AI methods are able to learn from past behavior of variables and recognize the complexity and non-linearity in a certain dataset, as well as capable of including additional inputs.

The statical approaches used for mid-term forecasting usually involve:

* the ARIMA (Autoregressive Integrated Moving Average) model, which originates from the autoregressive (AR) model, moving average (MA) model and
* the combination of AR and MA (ARMA). AR, MA and ARMA are applied to stationary series, while the ARIMA models are used for non-stationary series
* for seasonal time series, SARIMA models are used [3]
* stochastic models such as Markov chain [4]

The non-parametric AI based techniques used for load forecasting are usually artificial neural networks (ANN). As the tuning of predictor weights is an optimization process in nature, hybridization of ANN training process is often used so that Genetic Algorithms (GA) or Particle Swarm Optimization (PSO) is used to train the network instead of well-established algorithms such as the Back Propagation Algorithm (BPA) [5].

To perform the training and tune the predictor weights, the statistical forecasting approaches require large amounts of historical data to be present upfront as inputs to the process of supervised training [6]. When these data are not available, one can resort to modelled data such as the ones from meteorological reanalysis [7], or use an alternative forecasting method and learn online “as you go”.

The time horizon for mid-term forecasting ranges from short-term to a limit of several weeks ahead. Moreover, as the time horizon increases, so do the forecast errors [8]. In literature, medium-term forecasting covers the time spectrum from several hours to 1 week ahead, and is used for unit commitment decisions, reserve requirement decisions and generator online and/or offline decisions. In the context of BEYOND, the forecast method will rely on meteorological forecasts from a NWP provider, and it has been decided to utilize a robust neural network method with NWP data as inputs.

## Implementation in BEYOND

In BEYOND, the proposed methodology is based on artificial neural network as a predictor, that is trained on the input data in a supervised fashion, and then utilized as a trained predictor.

The required data pre-processing steps are the following:

1. Filter and nullify outliers and erroneous entries (input cleaning)
2. Fill-in eventual missing values using interpolation through padding
3. Perform resampling to 1 item daily or hourly
4. Configure the time horizon (forecasting step) ahead as a feature
5. Configure additional synthetic feature as day of the week
6. Perform normalisation of input values
7. Reshape dataset with lagging features so as to be ready for multi-variable, multi-step time series supervised learning

The training step is an iterative procedure, where the input dataset is split into training, test and validation sets. The training set is used to tune the predictor weights. The test and validation set to evaluate the performance and is not presented to the model during training. The validation set is often used as an early estimate of the model skill. Typical size ratios range in 60 to 70%, 20 to 30% and 10 to 20% for training, test and validation dataset sizes, respectively. Performing the forecast run on the test dataset provides a measure of the model performance.

The execution (usage) step of the analytics consists of running the input data through the trained predictor and obtaining the results.

### Data inputs and Analytics Pipeline (incl. assumptions /limitations)

Given the expected forecast horizon, the numerical weather prediction data is required as an input, available at appropriate time ahead of the actual forecasted production realization. This input data is used for training and validating the model. In this development phase, we rely on the meteorological reanalysis data provided by MERRA-2 system [9] and available through the Renewables.ninja website [10] coupled with publicly available data from the ECMWF [11]. The PVGIS data, also publicly available, are used for cross-validation of the results.

The analytics pipeline includes the required data pre-processing steps (see previous paragraph), training and testing of the model, as well as the evaluation of the results.

### Analytics Libraries Employed

The key Python libraries used for data manipulation and data analytics are the following:

* Pandas for time series management
* Numpy for numerical manipulation
* Sklearn and Keras for the implementation of the neural network
* Matplotlib for visualization of the training results

The first iteration of the prediction algorithm has been implemented as interactive Jupyter notebook.

# Anomaly/outlier detection of HVAC system component – KONCAR

## Rationale & Link to BEYOND Apps

The anomaly or outlier detection of HVAC system components analytics enables the detection of system failures and unusual consumption patterns derived from system malfunctions. Prompt and effective anomalies detection of HVAC systems are imperative for initiating repairs, correct maintenance plans and eliminating errors in HVAC system consumption forecasts.

The anomaly/outlier detection in the detection of HVAC system component will be available the framework of the BEYOND AI Analytics toolkit. The dedicated analytic will feed mainly the BEPO (Building Digital Twins Environment for Energy Performance Optimisation, Self-consumption Maximisation and Predictive Maintenance) application.

## Overview of relevant implementations

Generally speaking, the anomaly detection or the outlier analysis is a data mining procedure that aims to identify data points, events or observations that deviate from a normal behaviour present in the dataset.

A common feature of the applications of anomaly detection is that the input feature set is unlabeled so one must establish an a-priori criteria on which operation is considered to be normal, either by manual labelling, a priori decision on the normality criteria, or by assuming that the majority of input dataset is normal.

One of the models considers that the anomalous operation is rare in the input dataset. For this reason, another often used term for anomaly detection is novelty detection. In this setting, this is an example of unsupervised anomaly detection where the original assumption is that the majority of the input data set is normal. Still, one has to employ a distance metric to determine the distance of the observed sample from the general sample population.

Another model is to explicitly model the distance from the normal operation, which requires interpretation on how exactly the statistical distance metric reflects on the system operation. Finally, there is an approach of manually labelling the input dataset as normal versus abnormal. The latter two examples refer to supervised anomaly detection.

Typically, the anomalous operation of the system is not possible to be modelled directly as either the system modelled is unknown or too complex. Instead, there are sample observation of system operation and essentially, the task of anomaly detection is a classification task, where the current operational state of the equipment is divided into two “bins”: normal operation and anomalous or abnormal operation. For this reason the classic clustering techniques such as K-means clustering [12] are often utilized, but other techniques find their way into anomaly detection, such as neural networks, Bayesian networks, hidden Markov models, single class support vector machines and others [13].

The HVAC anomalies can be detected if the data are analysed in relation to previous or historical data. The evaluation of different distance-based, statistical based, and time-series anomaly detection algorithms shows that they are less precise to detect long-term anomalies in HVAC data set [13].

## Implementation in BEYOND

In BEYOND, the selected implementation resembles the implementation from [13]. The clustering approach is therefore used, based on the anomaly score.

1. Select the data and filter out obviously out-of-range data
2. Fill-in eventual missing values using interpolation through padding as there may be connectivity losses to the IoT sensing devices in the input dataset
3. Generate features by performing resampling
4. Perform normalisation of input values
5. Apply the pattern clustering method based on the distance metric and classify the system state as normal or anomalous.

After running the above task on the input data set, the performance score is calculated. In order to also verify the performance on a commonly used dataset, another dataset which is unrelated to HVAC performance but is commonly used for anomaly detection is also utilized in the above setting.

### Data inputs and Analytics Pipeline (incl. assumptions /limitations)

A typical approach in anomaly detection is to work with two datasets – one related to the domain in question, and to use a known dataset already well established and well utilized in the anomaly detection. For the former, IEEE DataPort HVAC Air Handling Units dataset will be used [14], and for the latter the MIT BIH arrythmia database will be used [15], available freely at PhysioNet website.

### Analytics Libraries Employed

The key Python libraries used for data manipulation and data analytics are the following:

* Pandas for time series management
* Numpy for numerical manipulation
* Sklearn for the implementation of the classifier
* Matplotlib for visualization of the training results

The first iteration of the prediction algorithm has been implemented as interactive Jupyter notebook.

[1] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time Series Analysis: Forecasting and Control*, 4. Auflage. John Wiley & Sons, 2008.

[2] F. Petropoulos *et al.*, “Forecasting: theory and practice,” *ArXiv201203854 Stat*, Oct. 2021, Accessed: Nov. 03, 2021. [Online]. Available: http://arxiv.org/abs/2012.03854

[3] V. G. Tran, V. Debusschere, and S. Bacha, “One week hourly electricity load forecasting using Neuro-Fuzzy and Seasonal ARIMA models,” *IFAC Proc. Vol.*, vol. 45, no. 21, pp. 97–102, Jan. 2012, doi: 10.3182/20120902-4-FR-2032.00019.

[4] S. X. Chen, H. B. Gooi, and M. Q. Wang, “Solar radiation forecast based on fuzzy logic and neural networks,” *Renew. Energy*, vol. 60, pp. 195–201, Dec. 2013, doi: 10.1016/j.renene.2013.05.011.

[5] A. Baliyan, K. Gaurav, and S. K. Mishra, “A Review of Short Term Load Forecasting using Artificial Neural Network Models,” *Procedia Comput. Sci.*, vol. 48, pp. 121–125, Jan. 2015, doi: 10.1016/j.procs.2015.04.160.

[6] G. Aburiyana and M. E. El-Hawary, “An overview of forecasting techniques for load, wind and solar powers,” in *2017 IEEE Electrical Power and Energy Conference (EPEC)*, Oct. 2017, pp. 1–7. doi: 10.1109/EPEC.2017.8286192.

[7] I. Staffell and S. Pfenninger, “Using bias-corrected reanalysis to simulate current and future wind power output,” *Energy*, vol. 114, pp. 1224–1239, Nov. 2016, doi: 10.1016/j.energy.2016.08.068.

[8] C. Monteiro *et al.*, “Wind Power Forecasting: State-of-the-Art 2009,” *Argonne Natl. Lab.*, Nov. 2009, doi: 10.2172/968212.

[9] “MERRA.” https://gmao.gsfc.nasa.gov/reanalysis/MERRA/docs/ (accessed May 30, 2018).

[10] “Renewables.ninja.” https://www.renewables.ninja/ (accessed Aug. 27, 2021).

[11] E. Andersson, “Medium-range forecasts,” *ECMWF*, Apr. 16, 2014. https://www.ecmwf.int/en/forecasts/documentation-and-support/medium-range-forecasts (accessed Mar. 25, 2022).

[12] Y. Cheng, “Mean Shift, Mode Seeking, and Clustering,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 790–799, 1995, doi: http://doi.ieeecomputersociety.org/10.1109/34.400568.

[13] M. Munir, S. Erkel, A. Dengel, and S. Ahmed, “Pattern-Based Contextual Anomaly Detection in HVAC Systems,” in *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, Nov. 2017, pp. 1066–1073. doi: 10.1109/ICDMW.2017.150.

[14] G. Stamatescu, “HVAC Air Handling Units: One-year Data from Medium-to-Large Size Academic Building.” IEEE, Mar. 15, 2019. Accessed: Mar. 25, 2022. [Online]. Available: https://ieee-dataport.org/documents/hvac-air-handling-units-one-year-data-medium-large-size-academic-building

[15] G. B. Moody and R. G. Mark, “MIT-BIH Arrhythmia Database.” physionet.org, 1992. doi: 10.13026/C2F305.