Neural prophet is extension of prophet which incorporates deep learning. It includes trend, seasonality, and event as in prophet and adds three more components that is AR-Net, lagged regressor and future regressor.

The trend is concerned with the **overall variation of the signal** (and not the small fluctuations within the signal). NeuralProphet learns to detect the dates where a clear variation in trend occurs. These points are called change points. Between each of these, the trend is supposed to be linear

The seasonality of a time series is the **periodic movement** of the signal due to its nature.

Auto-regression (AR) refers to the process of **regressing a variable’s future value against its past values.** The number of past values included is usually referred to as the order p of the AR(p) model, order should be chosen based on the approximate length of relevant context in past observations.

A neural network model is trained with stochastic gradient descent to learn the AR coefficients. If we know the true order of the process, AR-Net effectively learns near-identical weights as classic AR implementations and is equally good at predicting the next value of the time series. If the order is unknown, AR-Net automatically learns the relevant weights.

Types:

* Linear AR:

The default AR-Net configuration **does not contain hidden layers** and is functionally identical to a classic AR model. In practice, it is a single layer NN with p inputs, h outputs, no biases, and no activation function. The single layer weights each regress a particular lag onto a particular forecast step.

* Deep AR:

AR-Net based AR module can model non-linear dynamics when **hidden layers are configured**. In this case, the module trains a fully connected Neural Network (NN) with the specified number of hidden layers and dimensions. The addition of hidden layers may lead to a better forecasting accuracy, however it a partial trade-off in interpretability. Instead of being able to directly quantify the contribution of a particular past observation to a particular prediction, we can only observe the relative importance of a given past observation on all predictions. This can be approximated by comparing the sums of the absolute weights of the first layer for each input position.

* Sparse AR:

AR-Net demonstrated that the correct order can be approximated by setting the order to a slightly larger than expected value when regularization is used to sparsify the model weights. This allows for a more convenient model configuration while retaining interpretability

Lagged Regressor:

Lagged regressors are used to correlate **other observed variables to our target** time series. They are often referred to as covariates. Unlike future regressors, the future of lagged regressors is unknown to us. At the time t of forecasting, we only have access to their observed past values.

Future Regressor:

To model future regressors, **both past and future values of these regressors have to be known**.

Events:

This last module processes **occasional events that may affect the target**. An event is a binary time series equal to 1 at time t if the event occurs. (0 otherwise). For each event, it is possible to add a lower and upper window, that is, for e.g. if we consider Christmas as the event with -2 the lower window and +1 the upper, the event will be equal to 0 every time excepting the 23rd, 24th, 25th, and 26th December with a value of 1 (-2 before Christmas to + 1 after).