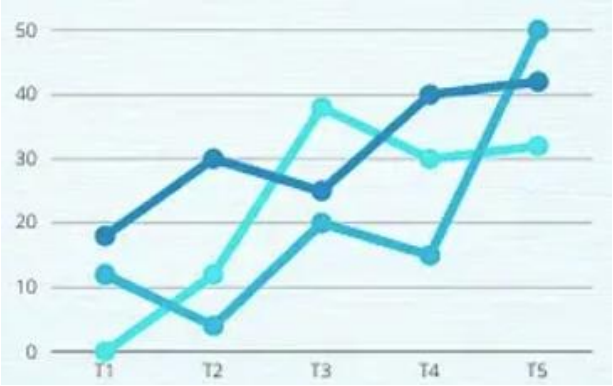
Autoregressive Time Series Modeling

Vidya Samadi, Ph.D. , M.ASCE
Clemson University

July 29, 2025

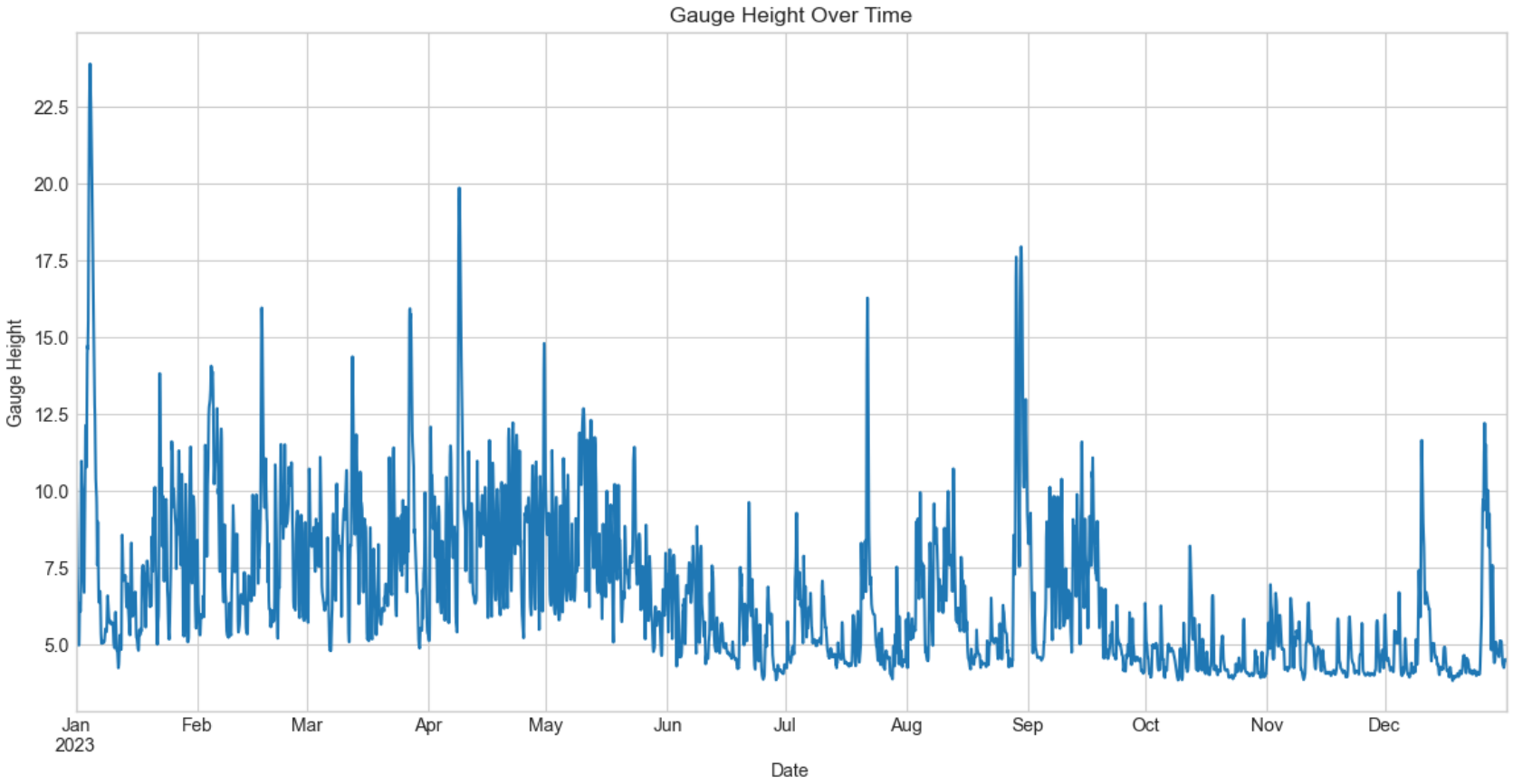


Reading

- Time Series - Preprocessing to Modelling
- Time Series For beginners with ARIMA
- ARIMA Model for Time Series Forecasting

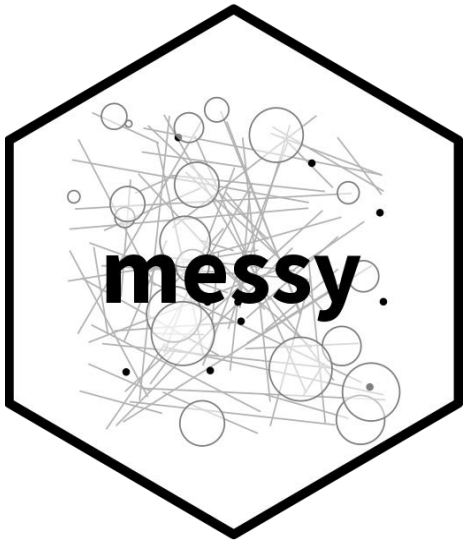
Time Series?

- A set of observations indexed by time *t*
- Discrete and continuous time series



Data preprocessing for time series prediction

- Water science time series data is messy
- Prediction models from simple rolling averages to deep learning implementation requires data to be clean
- So here are some techniques we could use before moving to deep learning implementation tomorrow!



Data preprocessing for time series prediction-cont.

Note: This data preprocessing step is general and intended to make researchers emphasize it as real-world projects involve a lot of cleaning and preparation.

Detrending/ Stationarity: Before prediction, we want our time series data to be mean-variance stationery.

- Statistical properties of a model do not vary depending on when the sample was taken.
- Models built on stationary data are generally more robust.
- Differencing is used to make the series stationary and to control the auto-correlations.

Data preprocessing for time series prediction-cont.

Anomaly detection: Any outlier present in the data might skew the prediction results, so it's often considered a good practice to identify and normalize outliers before moving on to prediction task.

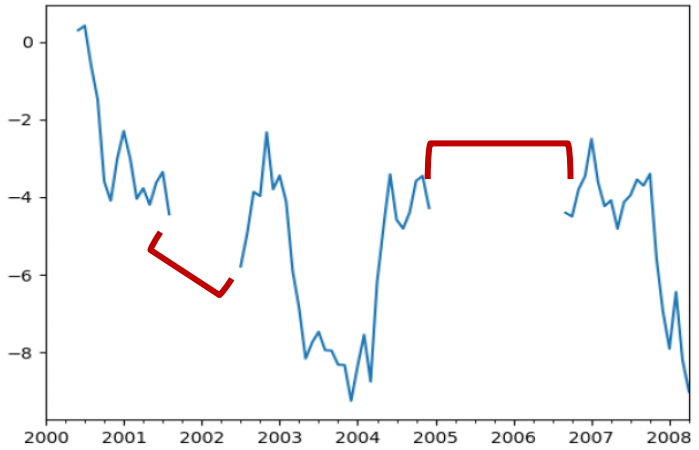
Sampling frequency: This is an important step to check the regularity of sampling. Irregular data has to be imputed or made uniform before applying any modeling techniques.

Data preprocessing for time series prediction-cont.

Missing data: missing data needs to be filled before modeling. For example, a time series data with missing values looks like this:

- Some techniques for imputing (creating) missing values include linear method, forward fill, backward fill, mean fill, and regression-based imputation (see standardization notebook in the WaterSoft package).

Note: regression models don't have methods to deal with missing values during training so passing in a dataset with "NaN" (not a number) values will cause errors in training.



Time series prediction models

Autoregressive Integrated Moving Average (ARIMA) is a class of linear models that utilizes historical values to predict future values.

Autoregressive + **Integrated** + **Moving Average**, each of which technique contributes to the final prediction. Let's understand these components one by one.

Autoregressive (AR)

Predict the variable of interest using a linear combination of past values of that variable. The term autoregression indicates that it is a regression of the variable against itself. That is, we use lagged values of the target variable as our input variables to predict values for the future. An autoregression model of order *p* will look like:

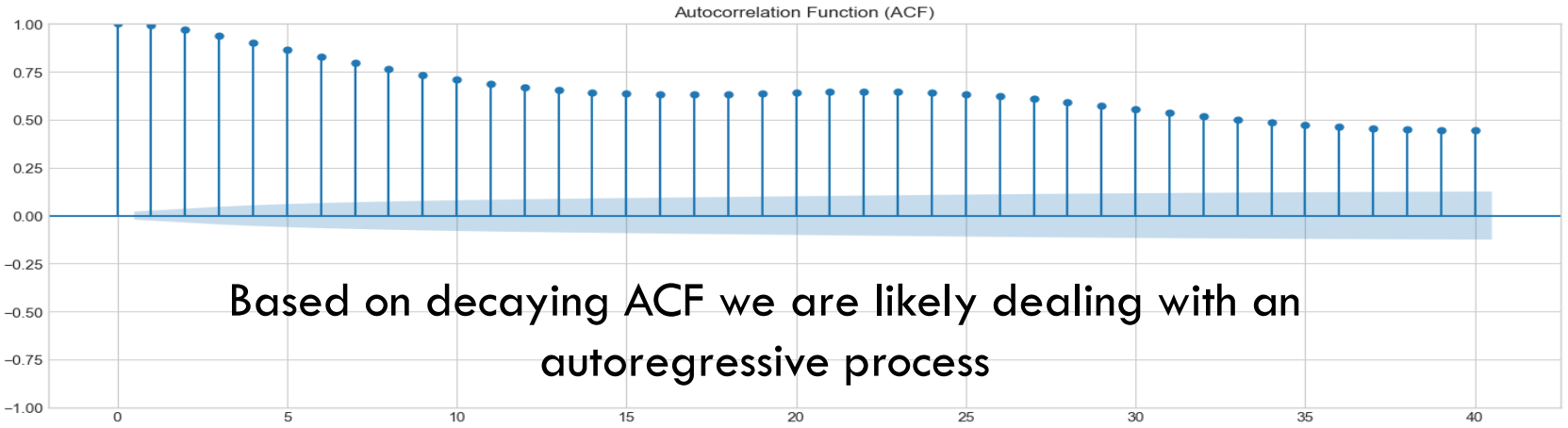
$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \cdots + \alpha_p y_{t-p} + e_t$$

Where:

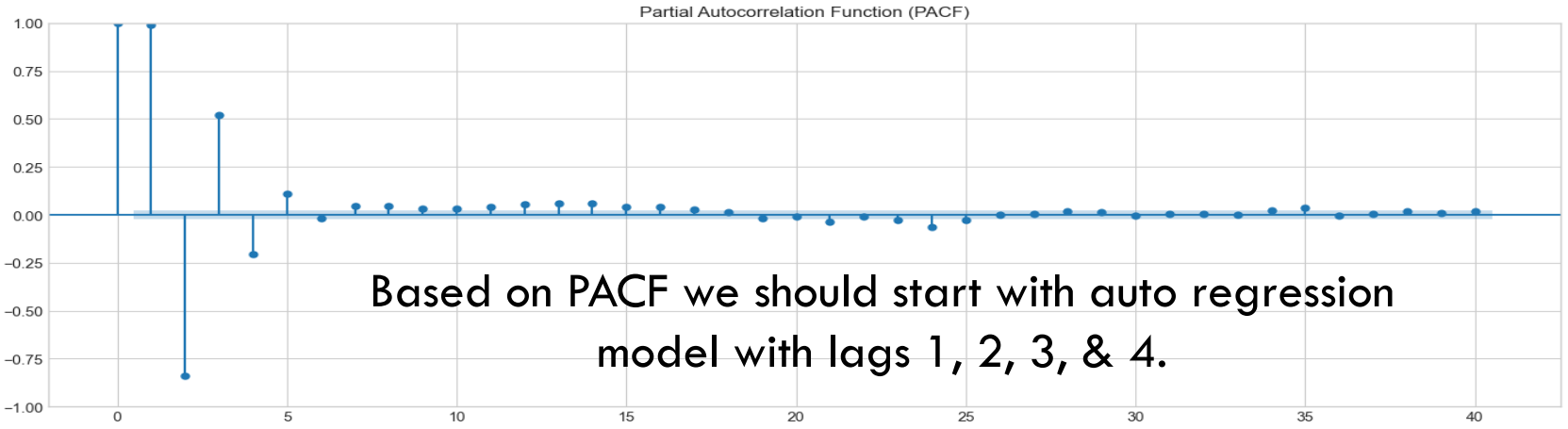
- $\alpha_1, \alpha_2, \ldots, \alpha_p$ are the AR parameters.
- e_t is the error term.
- p is the lag order, representing how many past values influence the current value.

There are some standard methods to determine optimal values of *p* one of which is, analyzing **Autocorrelation** and **Partial Autocorrelation** function plots.

Autocorrelation and Partial Autocorrelation for the Proctor Creek-Chattahoochee River watershed (USGS 02336490), GA



Based on decaying ACF we are likely dealing with an autoregressive process



Based on PACF we should start with auto regression model with lags 1, 2, 3, & 4.

Integrated (I)

Integrated represents any differencing that has to be applied in order to make the data stationary.

For first-order differencing:

$$y'_t = y_t - y_{t-1}$$

For second-order differencing:

$$y''_t = (y_t - y_{t-1}) - (y_{t-1} - y_{t-2}) = y_t - 2y_{t-1} + y_{t-2}$$

The number of times differencing is applied until the series becomes stationary.

Moving Average (Lagged Error)

Moving average models uses past prediction errors rather than past values in a regression-like model to predict future values. A moving average model can be denoted by the following equation:

Where:

$$y_t = e_t + \beta_1 e_{t-1} + \beta_2 e_{t-2} + \cdots + \beta_q e_{t-q}$$

- $\beta_1, \beta_2, \ldots, \beta_q$ are the MA parameters.
- e_t is the error at time t.
- q is the order of the moving average, indicating how many past errors influence the current value.

e (*error)* represents the random residual deviations between the model and the target variable. Since e can only be determined after fitting the model and since it's a parameter too so in this case e is an **unobservable parameter.** Hence, to solve the MA equation, iterative techniques such as Maximum Likelihood Estimation are used.

ARIMA Model

The ARIMA model combines the AR, I, and MA components into a single model.

$$d_t = \alpha_1 d_{t-1} + \alpha_2 d_{t-2} + \cdots + \alpha_p d_{t-p} + e_t + \beta_1 e_{t-1} + \cdots + \beta_q e_{t-q}$$

Where:

- d_t is the differenced series after applying differencing d times.
- $\alpha_1, \ldots, \alpha_p$ capture the relationship with past values (AR part).
- $\beta_1, \ldots, \beta_q$ capture the relationship with past errors (MA part).

```
arima = ARIMA(train, order=(6,1,6), freq='1h')
predictions = arima.fit().predict()
```

Determine p, d, **and** q**:** Use Auto-correlation Function and Partial Auto-correlation Function plots to decide on the AR and MA orders.

Estimate parameters: Use Maximum Likelihood Estimation to find the best-fitting parameters (α and β)

Model validation (residual analysis) and the use the fitted model to predict the future values.

Seasonal ARIMA

SARIMA includes seasonality contribution to the prediction. The importance of seasonality is quite evident and ARIMA fails to encapsulate that information implicitly.

In addition to the non-seasonal ARIMA parameters (p, d, q), SARIMA includes seasonal parameters $(P, D, Q)_s$, where:

- P **(Seasonal Autoregression):** Captures the relationship between the current value and its previous seasonal values.

$$y_t = \alpha_1 y_{t-s} + \alpha_2 y_{t-2s} + \cdots + \alpha_P y_{t-Ps} + \ldots$$

- D **(Seasonal Differencing):** Removes seasonal trends by differencing the series at the seasonal lag.

$$y'_t = y_t - y_{t-s}$$

- Q **(Seasonal Moving Average):** Models the dependency between the current value and past seasonal forecast errors.

$$y_t = e_t + \beta_1 e_{t-s} + \beta_2 e_{t-2s} + \cdots + \beta_Q e_{t-Qs} + \ldots$$

- s **(Seasonal Period):** The number of time steps in a seasonal cycle (e.g., $s = 12$ for monthly data with yearly seasonality).

Seasonal ARIMA-cont.

SARIMA $(p, d, q) \times (P, D, Q)_s$ Model

The SARIMA model combines both non-seasonal and seasonal components:

$$\Phi_P(B^s)\phi_p(B)(1-B)^d(1-B^s)^D y_t = \Theta_q(B)\Theta_Q(B^s)e_t$$

Where:

- $\phi_p(B)$ and $\Theta_q(B)$ are the non-seasonal AR and MA polynomials.
- $\Phi_P(B^s)$ and $\Theta_Q(B^s)$ are the seasonal AR and MA polynomials.
- B is the backshift operator, $By_t = y_{t-1}$.

Determine $p, d, q, P, D, Q,$ **and** s**:** Analyze both non-seasonal and seasonal ACF and PACF plots to identify appropriate orders.

Estimate parameters: Use Maximum Likelihood Estimation to estimate both non-seasonal and seasonal parameters.

Model validation (residual analysis) and **prediction-**utilize the fitted SARIMA model to make predictions that account for both trends and seasonality.

Summary

- ARIMA and SARIMA' simplicity and robustness make them a good candidate for modeling and forecasting- can be used as benchmark models
- Increasing *p,q* can increase the time complexity of training exponentially. So, it's advised to adjust their values priorly and then experiment.
- ARIMA and SARIMA are prone to overfitting. So, make sure you set the hyperparameters right and do validation before moving to prediction.
- Understanding the theoretical underpinnings of algorithms is indeed crucial for effective and innovative application.

THANK YOU

The support from the National Science Foundation (award # 2320979) is acknowledged.