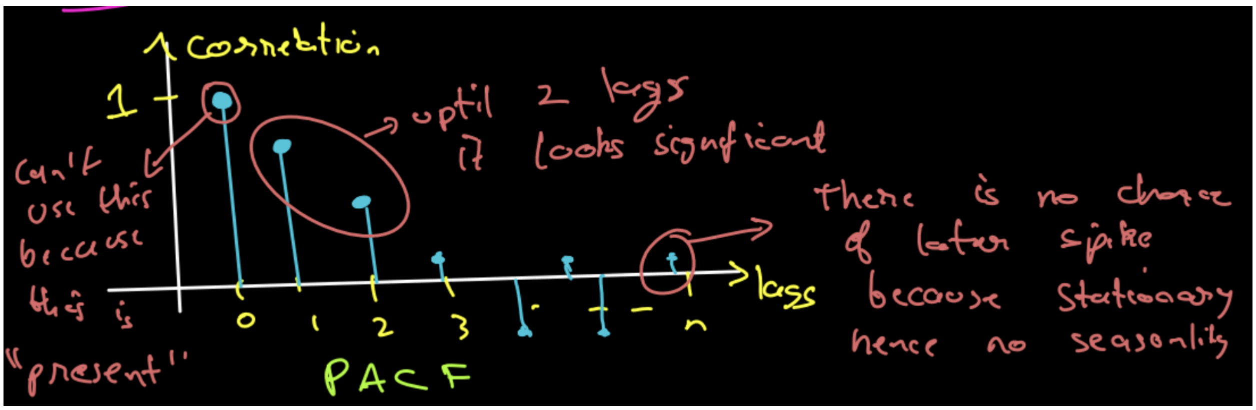
Time Series Analysis Lecture — 3

ARIMA Forecasting Techniques:



- **AutoRegression (AR(p)):**
  - Utilizes past values for forecasting in stationary time series.
  - Linear combination of past values as features for Linear Regression.
  - **Understanding p:** The hyperparameter p in AR(p) specifies the number of lagged observations included in the model. It reflects the extent to which past values influence future values, chosen based on the PACF plot to capture significant lag correlations.
- **Difference from SES:**
  - **SES**: Exponentially decaying weights, single hyperparameter (α).
  - **AR:** Weights are learned through the model, with the hyperparameter being the lag order p, which represents the number of past values considered for prediction.
- **Pre-requisites:**
  - Use PACF to avoid feature correlation; high PACF at lag k suggests correlation with future value.
  - Choose p based on PACF plot; significant lags indicate model order.
- **AR(0):** No dependence, equivalent to white noise.
- **Model Order Selection:**
  - Based on PACF spikes; ACF decays slowly in AR processes.
  - p determined by significant PACF lag values.

Moving Averages (MA(q)) Technique:

- **Concept:** Utilizes past forecast errors (residuals) for predicting current time period values.
- **Key Feature:** Creates unique feature from error of past value from mean.

- **Formulation:** Extends to order q, considering past q errors for current prediction.
- **Stationarity:** MA model inherently stationary, as observations are weighted averages of past errors.
- **Hyperparameter q:** Represents the order of the MA model, indicating the number of past errors considered. The optimal value of q is determined experimentally, often through model validation techniques.
- **Identification:**
    - ACF used to suggest order q; sharp cut-off in ACF after lag q indicates model order.
    - PACF decays more slowly in MA processes.
- **Difference from AR:**
    - AR focuses on past values, MA on past errors.
    - MA addresses correlated noise ignored by AR.

ARMA (AutoRegression-Moving Averages) Model:

- **Combination:** Integrates AR and MA techniques.
- **Formulation:** ARMA(p, q) with p as AR order, q as MA order.
- **Coefficients:** α1, α2, ..., αp for AR; m1, m2, ..., mq for MA.
- **Hyperparameters:** Orders p and q; not necessarily equal.
- **Stationarity Requirement:** Cannot handle non-stationary series.
- **Identification:**
    - AR: PACF plot for lag terms.
    - MA: ACF plot for error terms.
- **ACF and PACF in ARMA:**
    - ACF: Sharp cutoff after lag q.
    - PACF: Sharp cutoff after lag p.
- **Limitations:**
    - Unsuitable for non-stationary or seasonal data.
    - Differencing may lead to data loss and scale changes, requiring retransformation for forecasts.

ARIMA (AutoRegressive Integrated Moving Average):

- **Purpose:** Combines differencing (to achieve stationarity), AR, and MA for forecasting non-stationary time series.
- **Notation:** ARIMA(p, d, q)
    - p: Order of the AR part.
    - d: Degree of differencing.
    - q: Order of the MA part.

- **Process:**
  - 1. Differencing: To remove trend and make series stationary.
  - 2. ARMA Application: For approximating stationary series.
  - 3. Integration: Restores trend for final forecast.
- **Parameter Selection:**
  - Use grid search or AIC/BIC for optimal p, d, q.
  - AIC/BIC help compare model quality; lower values preferred.
- **AIC (Akaike Information Criteria):**
  - Balances model fit and simplicity.
  - Prefers models with higher log-likelihood and fewer parameters.
- **Limitations:**
  - May not capture seasonality effectively.

SARIMA (Seasonal AutoRegressive Integrated Moving Average):

- **Purpose:** Extends ARIMA to explicitly model seasonality.
- **Parameters:** SARIMA(P, D, Q, p, q, d, s)
  - **P, D, Q:** Seasonal AR, differencing, and MA orders.
  - **p, q, d:** Non-seasonal AR, MA, and differencing orders.
  - **s:** Seasonality period.
- **Seasonality (s):** Identifies repeating patterns over s periods; determined via ACF/PACF or tuning.
- **Hyperparameters:**
  - **P:** Seasonal AR order; impacts forecasting based on seasonal lags.
  - **Q:** Seasonal MA order; models errors from seasonal lags.
  - **D:** Seasonal differencing; removes seasonality before modeling.
- **Modeling Seasonality:**
  - P and Q enable AR and MA on data from past seasonal periods (e.g., 12, 24 months).
  - D applies seasonal differencing, enhancing stationarity with respect to seasonality.
- **Forecasting:**
  - Combines AR, MA, differencing, and seasonal components for accurate prediction.
  - Utilizes ACF and PACF for identifying appropriate seasonal lags.
- **Limitation:** Captures single seasonality type per model.
- **Identification:**
  - Seasonal spikes in ACF/PACF indicate ARIMA(P, D, Q) components.
  - SARIMA efficiently models and forecasts time series with complex seasonal patterns.
- **Implementation Example:**

```python
from statsmodels.tsa.statespace.sarimax import SARIMAX

# Sample implementation for monthly sales data
model = SARIMAX(data['Sales'], order=(p,d,q), seasonal_order=(P,D,Q,s))
results = model.fit()

# Forecasting
forecast = results.forecast(steps=12) # Forecasting the next 12 months
```

- **data['Sales']:** Time series data.
- **order=(p,d,q):** Non-seasonal parameters based on data analysis.
- **seasonal_order=(P,D,Q,s):** Seasonal parameters, with s, typically based on the data's seasonality pattern (e.g., 12 for monthly data).

This code snippet demonstrates a basic implementation of SARIMA using the statsmodels library in Python, tailored for time series with clear seasonality.