# Optimization Techniques for Machine Learning

AMLZC326 · #15 Time Series Forecasting

Anshid Aboobacker

# What is a Time Series?

**Definition:** A sequence of observations indexed by time:

$$y_1,\ y_2,\ \ldots,\ y_T$$

**Key property:** Observations are **not independent** — each depends on its past.

**Three goals:**

- *Description* — understand patterns
- *Explanation* — identify drivers
- **Forecasting** — predict future values

**Example: Weekly Sales**

| $t$ | $y_t$ |
|---|---|
| 1 | 12 |
| 2 | 15 |
| 3 | 11 |
| 4 | 14 |
| 5 | 13 |
| 6 | 16 |
| 7 | 12 |
| 8 | **?** |

Task: forecast $y_8$

# Learning Objectives

By the end of this lecture you should be able to:

- Identify trend, seasonality, and noise in a time series; test for stationarity using the ADF test
- Apply AR, MA, and ARIMA models; use ACF and PACF plots to select model order
- Handle seasonal patterns with SARIMA
- Understand the Kalman filter predict–update cycle and interpret the Kalman gain

# Baseline Forecasts: Historic Mean & Naive

**Given:** $y_1 = 12,\ y_2 = 15,\ y_3 = 11,\ y_4 = 14,\ y_5 = 13,\ y_6 = 16,\ y_7 = 12$

**Historic Mean:**

$$\hat{y}_{T+1} = \frac{1}{T}\sum_{t=1}^{T} y_t \quad \Rightarrow \quad \hat{y}_8 = \frac{12 + 15 + 11 + 14 + 13 + 16 + 12}{7} \approx \mathbf{13.3}$$

**Naive Forecast:**

$$\hat{y}_{T+1} = y_T \quad \Rightarrow \quad \hat{y}_8 = y_7 = \mathbf{12}$$

| Method | Formula | $\hat{y}_8$ |
|---|---|---|
| Historic Mean | $\bar{y}$ | 13.3 |
| Naive | $y_T$ | 12.0 |

**When to use:** Naive works well when the series behaves like a *random walk*. Historic Mean works when the series is stable with no trend.

# Baseline Forecast: Seasonal Naive

**Seasonal Naive:** repeat the value from the same position in the previous season.

$$\hat{y}_{T+h} = y_{T+h-s} \qquad (s = \text{season length})$$

**Example:** Daily sales over two weeks ($s = 7$):

| | Mon | Tue | Wed | Thu | Fri | Sat | Sun |
|---|---|---|---|---|---|---|---|
| Week 1 | 8 | 10 | 10 | 12 | 15 | 20 | 18 |
| Week 2 | 9 | 11 | 10 | 13 | 16 | 21 | 17 |
| Week 3 (forecast) | **9** | **11** | **10** | **13** | **16** | **21** | **17** |

| Method | Strength | Weakness |
|---|---|---|
| Historic Mean | Stable data | Ignores recent trends |
| Naive | Random walk | Ignores all structure |
| Seasonal Naive | Repeating patterns | No trend modelling |

# Table of Contents

# Simple Moving Average (SMA)

**SMA of order $k$:** average of the most recent $k$ values. $\text{SMA}_t = \frac{1}{k}\sum_{i=0}^{k-1} y_{t-i}$

**Worked example** ($k = 3$, weekly sales $y = 12, 15, 11, 14, 13, 16, 12$):

| $t$ | Calculation | $\text{SMA}_t$ |
|---|---|---|
| 3 | $(12 + 15 + 11)/3$ | 12.7 |
| 4 | $(15 + 11 + 14)/3$ | 13.3 |
| 5 | $(11 + 14 + 13)/3$ | 12.7 |
| 6 | $(14 + 13 + 16)/3$ | 14.3 |
| 7 | $(13 + 16 + 12)/3$ | 13.7 |

- Primary use: **smoothing** — reveals the underlying trend
- Limitation: all $k$ past values receive *equal* weight
- **Note:** SMA is a smoothing tool, **not** the MA model — distinction explained shortly

# Weighted MA & Exponential MA

**Weighted Moving Average (WMA):** $\text{WMA}_t = \sum_{i=0}^{k-1} w_i \, y_{t-i}, \qquad \sum w_i = 1$

**Example** (weights $[0.5,\ 0.3,\ 0.2]$, $k = 3$, at $t = 5$):

$$\text{WMA}_5 = 0.5(13) + 0.3(14) + 0.2(11) = 6.5 + 4.2 + 2.2 = \mathbf{12.9}$$

$$\text{SMA}_5 = (13 + 14 + 11)/3 = \mathbf{12.7}$$

Recent values get **higher weight** — more responsive to change.

**Exponential Moving Average (EMA):**

$$\text{EMA}_t = \alpha \, y_t + (1 - \alpha) \, \text{EMA}_{t-1}, \qquad 0 < \alpha < 1$$

- $\alpha$ close to 1: tracks data closely (high responsiveness)
- $\alpha$ close to 0: very smooth (low responsiveness)
- Weights decay exponentially: $\alpha,\ \alpha(1-\alpha),\ \alpha(1-\alpha)^2, \ldots$

# Table of Contents

# Components of a Time Series

**Additive model:**

$$y_t = T_t + S_t + R_t$$

**Multiplicative model:**

$$y_t = T_t \times S_t \times R_t$$

| **Component** | Meaning | Example |
|---|---|---|
| **Trend** $T_t$ | Long-run direction | Growing sales over years |
| **Seasonal** $S_t$ | Repeating periodic pattern | Higher sales every weekend |
| **Residual** $R_t$ | Unexplained noise | Random day-to-day variation |

## Key idea

Statistical models (AR, MA, ARIMA) aim to model $R_t$ after the trend and seasonal components have been accounted for.

# What is Stationarity?

**Weak Stationarity (required by AR/MA/ARIMA):**

- Constant mean: $E[y_t] = \mu$ for all $t$
- Constant variance: $\mathrm{Var}(y_t) = \sigma^2$ for all $t$
- Autocovariance depends only on lag $k$, not on $t$

**Why it matters:** Non-stationary series violate model assumptions $\Rightarrow$ unreliable forecasts.

**Common fixes to achieve stationarity:**

- **Differencing** — remove trend
- **Log transform** — stabilise variance
- **Seasonal differencing** — remove seasonal mean

# Testing for Stationarity

**Augmented Dickey-Fuller (ADF) Test:**

- $H_0$: unit root present (series is non-stationary)
- $p < 0.05$: reject $H_0 \Rightarrow$ series is **stationary**
- $p \geq 0.05$: fail to reject $\Rightarrow$ series is **non-stationary** — apply differencing or transformation

**Visual checks:**

- Time plot: look for trends, changing variance
- ACF plot: slowly decaying ACF suggests non-stationarity
- After differencing: re-test with ADF until $p < 0.05$

# Autocorrelation: ACF and PACF

**ACF** (Autocorrelation Function) at lag $k$:

$$\rho_k = \mathrm{Corr}(y_t,\ y_{t-k})$$

**PACF** (Partial ACF) at lag $k$: correlation between $y_t$ and $y_{t-k}$ *after removing the effect of all intermediate lags*.

**Model identification table:**

| ACF pattern | PACF pattern | Model suggested |
|---|---|---|
| Tails off gradually | Cuts off after lag $p$ | AR($p$) |
| Cuts off after lag $q$ | Tails off gradually | MA($q$) |
| Both tail off | Both tail off | ARMA($p$, $q$) |

"Cuts off" means significant spikes disappear sharply.
"Tails off" means spikes decay slowly (exponentially or in a damped wave).

# Table of Contents

# AR Models — Intuition

## Core idea

"Today's value is a weighted sum of its own past values, plus random noise."

**Analogy:**
Tomorrow's temperature $\approx$ fraction of today's temperature + random deviation

**Key insight:** AR is **ordinary linear regression** where the predictors are *lagged values of the same series*.

| | |
|---|---|
| Linear regression | $y_t = \beta_0 + \beta_1 x_t + \varepsilon_t$ |
| AR(1) | $y_t = \mu + \phi_1 y_{t-1} + \varepsilon_t$ |

The only difference: the predictor is $y_{t-1}$ (the series itself, lagged).

# AR(1) Model

**AR(1) Equation:** $y_t = \mu + \phi_1 y_{t-1} + \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, \sigma^2)$

**Stationarity condition:** $|\phi_1| < 1$

| Value of $\phi_1$ | Behaviour |
|---|---|
| $0 < \phi_1 < 1$ | Persistent; slow exponential decay toward mean |
| $-1 < \phi_1 < 0$ | Alternating zigzag pattern |
| $\phi_1 = 0$ | White noise (no autocorrelation) |
| $\lvert\phi_1\rvert = 1$ | Random walk (non-stationary!) |

**Numerical trace** ($\mu = 0$, $\phi_1 = 0.8$, starting at $y_0 = 10$):

$$y_1 = 0 + 0.8(10) + 0.5 = 8.5$$

$$y_2 = 0 + 0.8(8.5) + (-0.3) = 6.5$$

$$y_3 = 0 + 0.8(6.5) + 0.2 = 5.4 \quad \longrightarrow \text{ decays toward 0}$$

# AR($p$) Model

**General AR($p$) Equation:** $y_t = \mu + \sum_{i=1}^{p} \phi_i \, y_{t-i} + \varepsilon_t$

**Matrix form** (OLS estimation from first principles):

$$\hat{\phi} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$

where each row of $\mathbf{X}$ is $[1, \; y_{t-1}, \; y_{t-2}, \; \ldots, \; y_{t-p}]$.

**Yule-Walker equations** (alternative, uses autocorrelations):

$$\boldsymbol{\Gamma} \, \phi = \gamma$$

$\boldsymbol{\Gamma}$: Toeplitz autocorrelation matrix $\quad \gamma = [\rho_1, \ldots, \rho_p]^\top$

| | |
|---|---|
| Model selection | Use **PACF**: cuts off sharply after lag $p$ |
| Software | `AutoReg` (statsmodels), `AutoARIMA` |

# AR(1) — Worked Example

**Data:** Weekly sales $y = 10,\ 11,\ 12,\ 13,\ 14,\ 15,\ 15,\ 16$

**Fit the AR(1) model** $y_t = \mu + \phi_1 y_{t-1} + \varepsilon_t$ using linear regression of $y_t$ on $y_{t-1}$.

| $t$ | $y_{t-1}$ | $y_t$ |
|---|---|---|
| 2 | 10 | 11 |
| 3 | 11 | 12 |
| 4 | 12 | 13 |
| 5 | 13 | 14 |
| 6 | 14 | 15 |
| 7 | 15 | 15 |
| 8 | 15 | 16 |

**OLS estimates:** $\hat{\mu} \approx 1.71 \quad \hat{\phi}_1 \approx 0.93$

**Forecast for $t = 9$:**
Using the latest value $y_8 = 16$,

$$\hat{y}_9 = 1.71 + 0.93(16) \approx 16.6$$

# Table of Contents

# MA Models: Not the Same as SMA!

## Common Confusion

The **MA($q$) model** and the **Simple Moving Average** are completely different.

| | SMA | WMA/EMA | MA($q$) Model |
|---|---|---|---|
| **Input** | Past $y$ values | Past $y$ values | Past **error terms** $\varepsilon$ |
| **Purpose** | Smoothing | Smoothing | Probabilistic model |
| **Output** | Smoothed value | Smoothed value | $y_t$ with uncertainty |
| **Parameters** | Window $k$ | Weights $w_i$ | $\theta_1, \ldots, \theta_q$ |
| **Forecasting** | Limited | Limited | Yes — full model |

**Intuition for MA($q$):**

"Today's value = today's shock + echoes of *recent shocks*"

(e.g. a supply disruption affects sales for several days)

# MA(1) and MA($q$) Models

**MA(1) Equation:** $y_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1}, \qquad \varepsilon_t \sim \mathcal{N}(0, \sigma^2)$

**MA($q$) General Form** $y_t = \mu + \varepsilon_t + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j}$

MA($q$) is **always stationary** for any values of $\theta$.

**Numerical trace** (MA(1), $\mu = 0$, $\theta_1 = 0.7$, errors $\varepsilon = \{1.2,\ -0.5,\ 0.8,\ -0.3\}$):

$$y_2 = (-0.5) + 0.7(1.2) = 0.34$$

$$y_3 = (0.8) + 0.7(-0.5) = 0.45$$

$$y_4 = (-0.3) + 0.7(0.8) = 0.26$$

**Model selection:** Use **ACF** — cuts off sharply after lag $q$.

# Table of Contents

# Differencing for Stationarity

**First difference:** $\Delta y_t = y_t - y_{t-1}$

**Second difference:** $\Delta^2 y_t = \Delta y_t - \Delta y_{t-1}$

**When to apply differencing:**

- Apply $d = 1$ when the ADF test shows non-stationarity (trending series)
- Apply $d = 2$ if $\Delta y_t$ is still non-stationary (accelerating trend)
- **Rule of thumb:** most business series need $d = 1$; confirm with ADF test

# Differencing: Worked Example

**Trending series** ($y_t = 10, 13, 17, 22, 28$):

| $t$ | $y_t$ | $\Delta y_t$ | $\Delta^2 y_t$ | Stationary? |
|---|---|---|---|---|
| 1 | 10 | — | — | No |
| 2 | 13 | $+3$ | — | No |
| 3 | 17 | $+4$ | $+1$ | Approx. yes |
| 4 | 22 | $+5$ | $+1$ | Approx. yes |
| 5 | 28 | $+6$ | $+1$ | Approx. yes |

- $\Delta y_t$ still increases $\Rightarrow$ non-stationary after $d = 1$
- $\Delta^2 y_t \approx$ constant $\Rightarrow$ stationary after $d = 2$
- Interpretation: the **acceleration** (second difference) is constant — the series has a quadratic trend

# ARIMA Model

ARIMA = **A**uto**R**egressive + **I**ntegrated + **M**oving **A**verage

1. Difference $d$ times: $w_t = \Delta^d y_t$
2. Fit ARMA($p, q$) on $w_t$: $w_t = \sum_{i=1}^{p} \phi_i \, w_{t-i} + \varepsilon_t + \sum_{j=1}^{q} \theta_j \, \varepsilon_{t-j}$

| Parameter | Meaning | Choose via | Typical value |
|---|---|---|---|
| $p$ | AR order | PACF of $\Delta^d y_t$ | 0–3 |
| $d$ | Differencing | ADF test | 0, 1, or 2 |
| $q$ | MA order | ACF of $\Delta^d y_t$ | 0–3 |

# ARIMA: Special Cases

**ARIMA unifies many classic models:**

| Notation | Equivalent model | Description |
|---|---|---|
| ARIMA$(1,0,0)$ | AR(1) | Autoregressive, stationary |
| ARIMA$(0,0,1)$ | MA(1) | Moving average, stationary |
| ARIMA$(0,1,0)$ | Random Walk | $\hat{y}_{T+1} = y_T$ |
| ARIMA$(0,1,1)$ | Exp. Smoothing | EMA with optimised $\alpha$ |
| ARIMA$(p,0,q)$ | ARMA$(p,q)$ | No differencing needed |

**Model selection workflow:**

1. ADF test $\Rightarrow$ choose $d$
2. ACF of $\Delta^d y_t \Rightarrow$ choose $q$
3. PACF of $\Delta^d y_t \Rightarrow$ choose $p$
4. Compare candidate models using AIC/BIC

# Model Selection: AIC and BIC

$$\text{AIC} = 2k - 2\ln\hat{L} \qquad \text{BIC} = k\ln T - 2\ln\hat{L}$$

$k$ = number of parameters $\quad T$ = number of observations
$\hat{L}$ = maximised log-likelihood

- **Lower** AIC/BIC = better model
- BIC penalises complexity more $\Rightarrow$ tends to select simpler models
- AIC can favour slightly more complex models

**Example grid search:**

| Model | $k$ | AIC | BIC |
|---|---|---|---|
| ARIMA(0,1,1) | 2 | 214.3 | 217.1 |
| **ARIMA(1,1,1)** | **3** | **213.1** | **217.6** |
| ARIMA(2,1,2) | 5 | 216.8 | 224.3 |
| ARIMA(1,1,0) | 2 | 218.5 | 221.3 |

In practice: `AutoARIMA` searches this space automatically.

# Table of Contents

# Why ARIMA Falls Short on Seasonal Data

**Problem:** ARIMA residuals may still contain periodic patterns.

**Example:** Monthly ice cream sales fitted with ARIMA(1,1,1):

- Residual ACF still shows spikes at lags 12, 24, 36
- The **seasonal structure** (higher sales every summer) is *unmodelled*
- Forecasts will be systematically wrong at seasonal peaks/troughs

**Solution:** Extend ARIMA with a **seasonal component** that operates at lag $s$.

**Common season lengths:**

| Data frequency | Season length $s$ | Example |
|---|---|---|
| Daily | 7 | Weekly pattern |
| Monthly | 12 | Annual pattern |
| Quarterly | 4 | Annual pattern |

# SARIMA: Model and Parameters

**Full Notation:** SARIMA$(p, d, q)$ $(P, D, Q)_s$

| Symbol | Meaning | Non-seasonal | Seasonal |
|---|---|---|---|
| $p$, $P$ | AR order | Lags 1–$p$ | Lags $s$, $2s$, …, $Ps$ |
| $d$, $D$ | Differencing | $\Delta^d y_t$ | $\Delta_s^D y_t = y_t - y_{t-s}$ |
| $q$, $Q$ | MA order | Errors $\varepsilon_{t-j}$ | Errors at seasonal lags |
| $s$ | Season length | — | 7, 12, 4, … |

**Seasonal differencing:** $\Delta_s y_t = y_t - y_{t-s}$

**Example:** Monthly data ($s = 12$): $\Delta_{12} y_t = y_t - y_{t-12}$ removes the annual seasonal mean.

# SARIMA: Backshift Notation

**Compact backshift notation:**

$$\Phi_P(B^s)\,\phi_p(B)\,(1-B)^d(1-B^s)^D\,y_t = \Theta_Q(B^s)\,\theta_q(B)\,\varepsilon_t$$

where $B$ is the backshift operator: $B\,y_t = y_{t-1}, \quad B^s\,y_t = y_{t-s}$

**Identifying seasonal order from ACF/PACF:**

- Significant ACF spike at lag $s$ (and $2s$) $\Rightarrow$ seasonal MA component ($Q \geq 1$)
- Significant PACF spike at lag $s \Rightarrow$ seasonal AR component ($P \geq 1$)
- Slowly decaying ACF at seasonal lags $\Rightarrow$ apply seasonal differencing ($D = 1$)

In code: `AutoARIMA(season_length=7)` — selects all six parameters automatically.

# Table of Contents

# Kalman Filtering — A Different Perspective

## Core idea

There is a **hidden true state** $x_t$ behind noisy observations $y_t$. The filter estimates $x_t$ optimally at each step.

| | ARIMA | Kalman Filter |
|---|---|---|
| Models | $y_t$ directly | Latent state $x_t$ |
| Input | Past $y$ values | Noisy measurements $y_t$ |
| Output | Future $y$ forecast | Filtered/smoothed $x_t$ |

**GPS analogy:**

| Concept | GPS navigation | Time series |
|---|---|---|
| Hidden state $x_t$ | True position | True underlying signal |
| Observation $y_t$ | GPS reading (noisy) | Recorded value (noisy) |
| Dynamics model | Speed $\times$ time | Random walk / trend |

# State-Space Model

**State Transition Equation:** $x_t = A\,x_{t-1} + w_t, \qquad w_t \sim \mathcal{N}(0,\ Q)$
How the hidden state **evolves** over time.

**Observation Equation** $y_t = C\,x_t + v_t, \qquad v_t \sim \mathcal{N}(0,\ R)$
How we **measure** the hidden state (with noise).

| **Symbol** | Name | Role |
|---|---|---|
| $A$ | Transition matrix | How state evolves ($A = 1$: random walk) |
| $C$ | Observation matrix | How state maps to measurement |
| $Q$ | Process noise | How much state can change each step |
| $R$ | Measurement noise | Sensor / recording accuracy |

For the 1-D scalar case: $A = C = 1$; $Q$ and $R$ are scalars.

# KALMAN: PREDICT STEP

**Propagate the prior state estimate forward in time:**

## PREDICT

$$\hat{x}_{t|t-1} = A\,\hat{x}_{t-1} \qquad \text{(predicted state)}$$

$$P_{t|t-1} = A\,P_{t-1}\,A^{\top} + Q \qquad \text{(predicted error covariance)}$$

- $A$: state transition matrix; $Q$: process noise covariance
- The predicted covariance $P_{t|t-1}$ grows — uncertainty increases without new data
- Next: the Update step corrects this prediction using the measurement $y_t$

# Kalman: Update Step

**Correct the prediction using the new measurement $y_t$:**

## UPDATE

$$K_t = P_{t|t-1}\, C^\top \left(C\, P_{t|t-1}\, C^\top + R\right)^{-1} \qquad \textbf{(Kalman Gain)}$$

$$\hat{x}_t = \hat{x}_{t|t-1} + K_t\left(y_t - C\,\hat{x}_{t|t-1}\right)$$

$$P_t = \left(I - K_t\, C\right) P_{t|t-1}$$

- $\left(y_t - C\hat{x}_{t|t-1}\right)$: **innovation** (prediction error)
- $K_t$ close to 0: trust the prediction; $K_t$ close to 1: trust the measurement
- $P_t \leq P_{t|t-1}$: uncertainty decreases after incorporating new data

# Kalman Gain — Intuition & Numerical Example

$K_t$ balances trust between the **model** and the **measurement**:

| Scenario | $K_t$ value | Filter behaviour |
|---|---|---|
| $R \ll Q$ (accurate sensor) | $K_t \to 1$ | Tracks measurement closely |
| $Q \ll R$ (stable dynamics) | $K_t \to 0$ | Smooth, slow to react |
| Balanced | $0 < K_t < 1$ | Weighted combination |

**Numerical trace** ($Q = 1$, $R = 5$, $P_0 = 1$, $\hat{x}_0 = 0$, first measurement $y_1 = 4$):

$$P_{\text{pred}} = 1 \cdot 1 \cdot 1 + 1 = 2$$

$$K_1 = \frac{2 \cdot 1}{1 \cdot 2 \cdot 1 + 5} = \tfrac{2}{7} \approx 0.286$$

$$\hat{x}_1 = 0 + 0.286\,(4 - 0) = 1.14$$

$$P_1 = (1 - 0.286) \cdot 2 = 1.43$$

The filter moves 28.6% of the way toward the measurement — blending prior and data.

# Table of Contents

# When to Use Which Model

| Model | Best for | Watch out for |
|---|---|---|
| Historic Mean | Stable, no pattern | Ignores recent changes |
| Naive | Random walk data | Ignores all structure |
| Seasonal Naive | Pure repeating patterns | No trend modelling |
| SMA / WMA / EMA | Smoothing, trend reveal | Not a statistical model |
| AR($p$) | Autocorrelated, stationary | Cannot capture MA shocks |
| MA($q$) | Short-memory shock data | No long-run autocorrelation |
| ARIMA | Trend + autocorrelation | Fails on seasonal data |
| SARIMA | Seasonal + trend | More parameters to estimate |
| Kalman Filter | Noisy latent-state systems | Requires tuning $Q$, $R$ |

# Key Equations at a Glance

| Model | Core Equation |
|---|---|
| Naive | $\hat{y}_{T+1} = y_T$ |
| Seasonal Naive | $\hat{y}_{T+h} = y_{T+h-s}$ |
| SMA($k$) | $\hat{y}_{T+1} = \frac{1}{k} \sum_{i=0}^{k-1} y_{T-i}$ |
| EMA | $\text{EMA}_t = \alpha y_t + (1-\alpha)\,\text{EMA}_{t-1}$ |
| AR($p$) | $y_t = \mu + \sum_{i=1}^{p} \phi_i\, y_{t-i} + \varepsilon_t$ |
| MA($q$) | $y_t = \mu + \varepsilon_t + \sum_{j=1}^{q} \theta_j\, \varepsilon_{t-j}$ |
| ARIMA | Fit ARMA($p$, $q$) on $\Delta^d y_t$ |
| SARIMA | ARIMA + seasonal ARMA at lag $s$ |
| KF Update | $\hat{x}_t = \hat{x}_{t\|t-1} + K_t\left(y_t - C\hat{x}_{t\|t-1}\right)$ |

Thank you :)