

Forecasting And Time Series Analysis

Steve Elston

11/10/2022

Why Are Time Series Useful?

- Data are often time-ordered
- Estimates 30% of data science problems include time series data
- Must use specific time series models

Why Are Time Series Useful?

“It’s tough to make predictions, especially about the future!”

Karl Kristian Steincke, Danish politician, ca 1937

- **Demand forecasting:** Electricity production, Internet bandwidth, Traffic management, Inventory management, sales forecasting
- **Medicine:** Time dependent treatment effects, EKG, EEG
- **Engineering and Science:** Signal analysis, Analysis of physical processes
- **Capital markets and economics:** Seasonal unemployment, Price/return series, Risk analysis

Why Are Time Series Data Different?

Models must account for time series behavior

- Most statistical and machine learning assume data samples are **independent identically distributed (iid)**
- But, this is not the case for time series data
- Time series values are correlated in time
- Time series data exhibit **Serial correlation**
 - *Serial correlation of values*
 - *Serial correlation of errors*
 - *Violate iid assumptions of many statistical and ML Models*

Why Are Time Series Data Different

Examples of series correlation:

- Temperature forecasts, where the future values are correlated with the current values
- The opening price of a stock is correlated with the price at the previous close
- The daily sales volume of a product is correlated with the previous sales volume
- A medical patient's blood pressure reading is correlated with the previous observations

History of Time Series Analysis

Time series analysis have a long history

- Recognized the serial dependency in time series data early on
- Joseph Fourier and Siemon Poisson worked on time series problems in the early 19th Century



Joseph Fourier

History of Time Series Analysis

Modern history of time series analysis started with George Udny Yule (1927) and Gilbert Walker (1931)

- Yule worked on sunspot time series
- Walker was attempting to forecast the tropical monsoon cycle
- Developed the **autoregressive (AR)** model to account for serial correlation
- The AR model is foundation of modern time series models

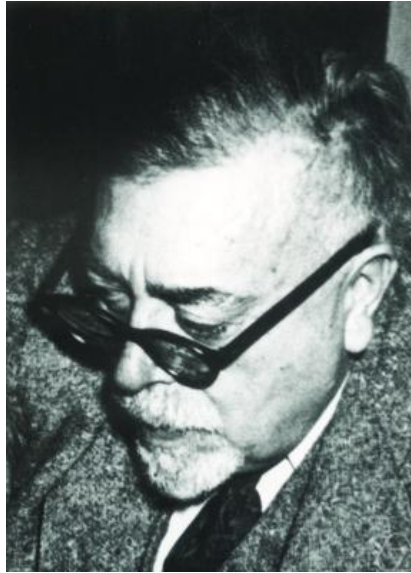


George Yule, time series pioneer

History of Time Series Analysis

Mathematical prodigy, Norbert Wiener, invented filters for stochastic time series processes, starting in the 1920s

- Wiener's filter theory is the basis of many time series filter methods
- Predictive filters for noisy signals; not discussed here
- Wiener process model for random walks is widely used



Norbert Wiener: Invented time series filters

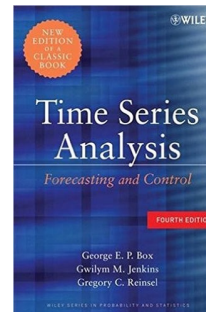
History of Time Series Analysis

George Box and Gwilym Jenkins fully developed a statistical theory of time series by extending the work of Yule and Walker in the 1950s and 1960s

- Extended the AR model to include **moving average (MA)** terms
- Included the **integrative term** to create the **ARIMA** model
- The ARIMA model is our focus



George EP Box created general time series model



Seminal time series analysis book

History of Time Series Analysis

21st Century time series analysis

- Considerable research continues to expand the frontiers
- Bayesian time series models
 - *R [bsts package](#) and Python [PyMC3](#)*
- Long short term memory (LSTM) model
- Hidden Markov Models (HMMs) widely used
 - *Python [Scikit Learn HMM](#) or R [HMM package](#)*

Software for Time Series Analysis

Most statistical packages have considerable time series modeling capability

- R time series analysis packages are wide and deep
 - *Much leading edge research appears first in R packages*
 - [CRAN Time Series Task View](#), maintained by Rob Hyndman, contains curated index to R time series packages
- Primary Python time series analysis package in [Statsmodels.tsa](#)
- Bayesian time series models supported in [PyMC](#).
- Many newer Python time series packages packages, including:
 - [Darts package](#) includes cutting edge methods like hierarchical models
 - [Facebook Kats](#) package - strong in forecasting including the [PROFIT model](#)
 - [GrayKite](#) LinkedIn's forecasting package

Fundamentals of Time Series

What are the fundamental properties of time series

- Representation and sampling
- White noise series
- Stationary time series
- Autocorrelation and partial autocorrelation
- Random walk series
- Trend
- Seasonal effects

Time Series Representation

Time series are expressed as a time ordered sequence of values $(x_1, x_2, x_3, \dots, x_n)$

- We work with **discrete samples** in time order
- In **regular time series** the sample interval Δt is fixed
- Time measured from start of series $(0, \Delta t, 2\Delta t, \dots, n\Delta t)$
- Or, time measured within an **interval**, multiples of Δt
- Even continuous time processes are sampled in practice
 - *Temperature*
 - *Pressure*
 - *Home price*

White Noise Series

White noise series are fundamental

- Values are **independent identically distributed (iid)** Normally distributed
- Can express values, $(w_1, w_2, w_3, \dots, w_n)$, of a white noise series as:

$$X(t) = (w_1, w_2, w_3, \dots, w_n)$$

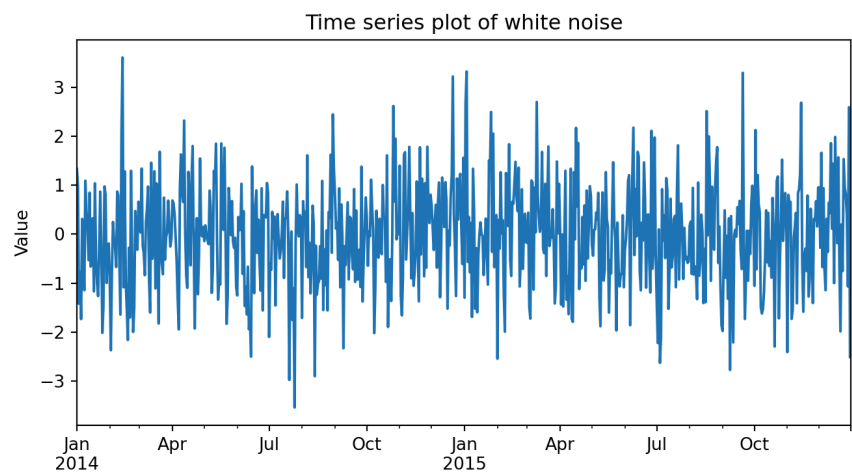
where

$$w_t \sim N(0, \sigma^2)$$

- No serial correlation between values
 - *There is no predictive information in a white noise series*
 - We want the **residuals** of time series models to be white noise series

White Noise Series

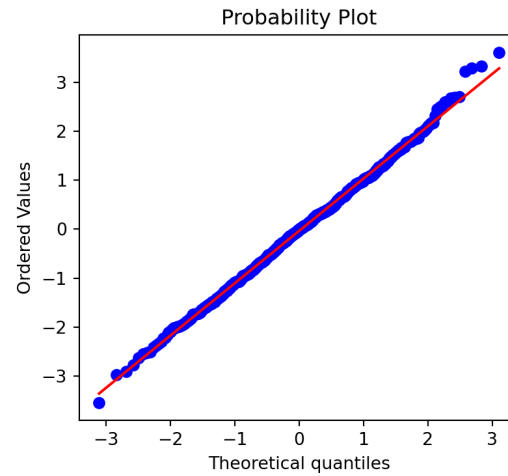
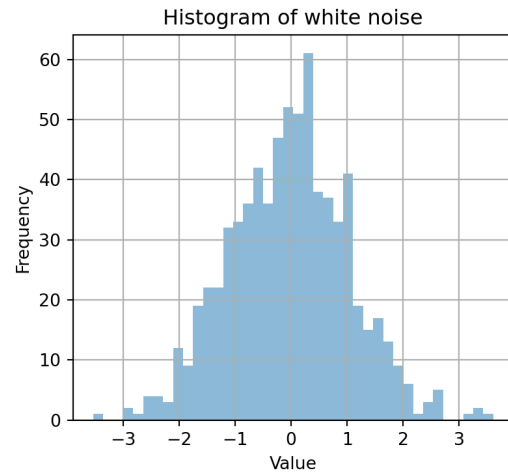
What does a white noise series look like?



White Noise Series

What does a white noise series look like?

- Each value is a sample is iid Normally distributed
- No trend



Stationary Time Series

A white noise series is **stationary**

- A stationary time series has statistical properties constant in time
- For example, a stationary time series has **constant mean and variance** over any sample interval
- Many time series models require stationarity
 - *Often transform time series to make them stationary*
 - *More on this shortly*

Stationary Time Series

What tests can be used for stationarity?

- Plots

- *Qualitative*
- *Nonstationarity from seasonality and trend are usually visible*

- The [Dickey Fuller test](#)

- *Hypothesis test for stationarity*
- *Null hypothesis is stationarity*
- *Based on roots of $AR(1)$ model; to be discussed shortly*

Autocorrelation Properties of White Noise Series

Can measure the correlation of a time series with itself

- The time series is correlated at different time offsets
- Each time step of offset is called a **lag**
- The **autocorrelation function (ACF)** is measured between the series and the series lagged in time
- Always 1 at lag 0; $ACF(0) = 1.0$

Autocorrelation Properties of White Noise Series

We compute the autocorrelation at lag k :

$$\rho_k = \frac{\gamma_k}{n\sigma^2} = \frac{1}{n\sigma^2} \sum_{t=1}^N (y_t - \mu) \cdot (y_{t-k} - \mu)$$

Where:

$k = \text{lag}$

$y_t = \text{observaton at time } t$

$\gamma_k = \text{covariance lag } k$

$\mu = \text{mean of the series}$

$$\sigma^2 = \text{variance of the series} = \frac{1}{n-1} \sum_{t=1}^N (y_t - \mu) \cdot (y_t - \mu)$$

- Notice that for any series, $\rho_0 = 1$
- Autocorrelation at each lag has values in the range $-1.0 \geq \rho \geq 1.0$

Autocorrelation Properties of White Noise Series

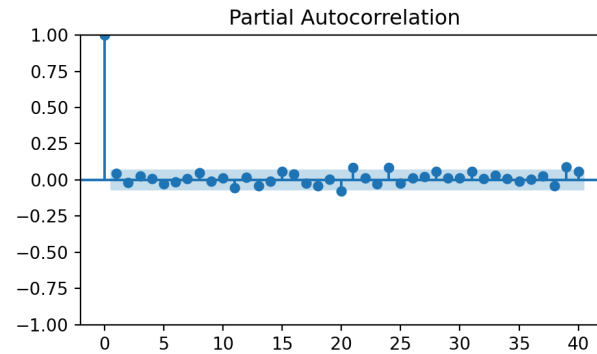
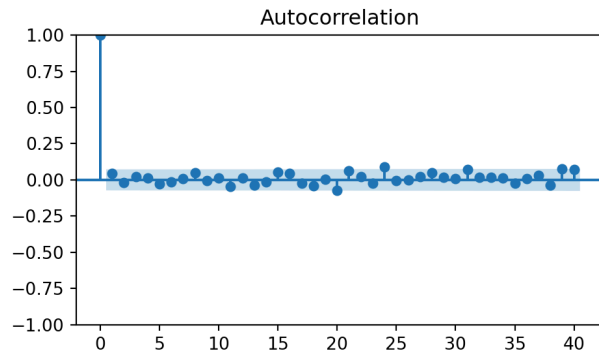
The **partial autocorrelation** is another important property of time series

- The **partial autocorrelation function (PACF)** is the residual autocorrelation once autocorrelation is accounted for
- To compute the partial autocorrelation to lag k :
 - *Compute the autocorrelation to lag k*
 - *Remove the linearly predictable autocorrelation component of the time series*
 - *Compute the (partial) autocorrelation of the residual to lag k*
- The 0 lag value of the partial autocorrelation is always 1.0

Autocorrelation Properties of White Noise Series

What are the autocorrelation and partial autocorrelation properties of a white noise series?

- The autocorrelation and partial autocorrelation are 0 for all $k > 0$
- Autocorrelation plot shows value at each lag selected



Random Walk Time Series

Random walks are a commonly encountered properties of time series

- Change in value of random walk series at one time step:

$$w_t = y_t - y_{t-1}$$

- The next value in the random walk is then:

$$y_t = y_{t-1} + w_t$$

Or, with a little bit of algebra:

$$y_t = w_t + \sum_{i=0}^{t-1} w_i$$

- w_i is the i th **innovation**
- y_t = observarition at time t
- A random walk is an **integrative process**; sum or integral of innovations

Note: innovations are referred to by other names:

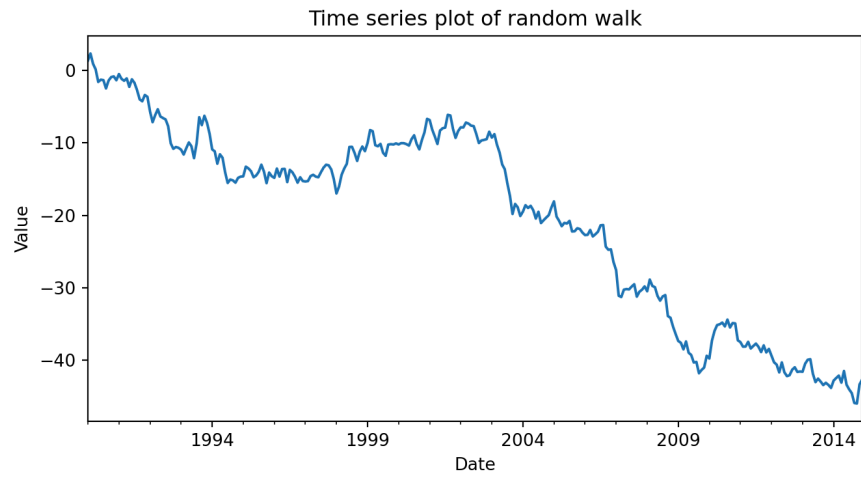
- **Shocks** in the stochastic process literature
- **Returns** in financial analytics

Random Walk Time Series

What does a random walk time series look like?

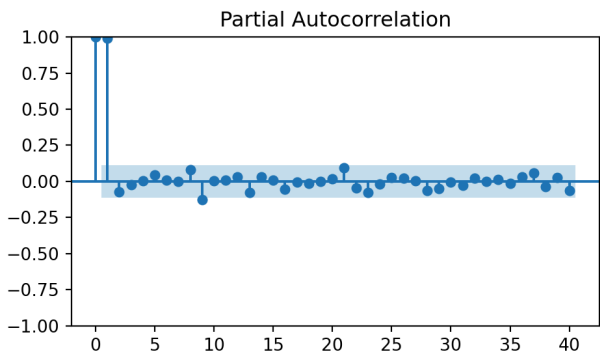
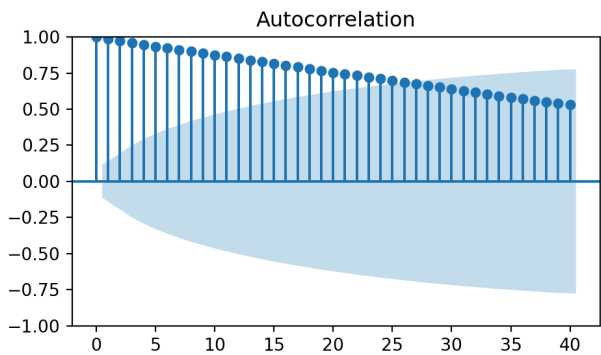
- Integrating innovations leads to a ‘drift-like’ behavior
- No actual trend; random walk will eventually change apparent slope

Example with iid Normal innovations:



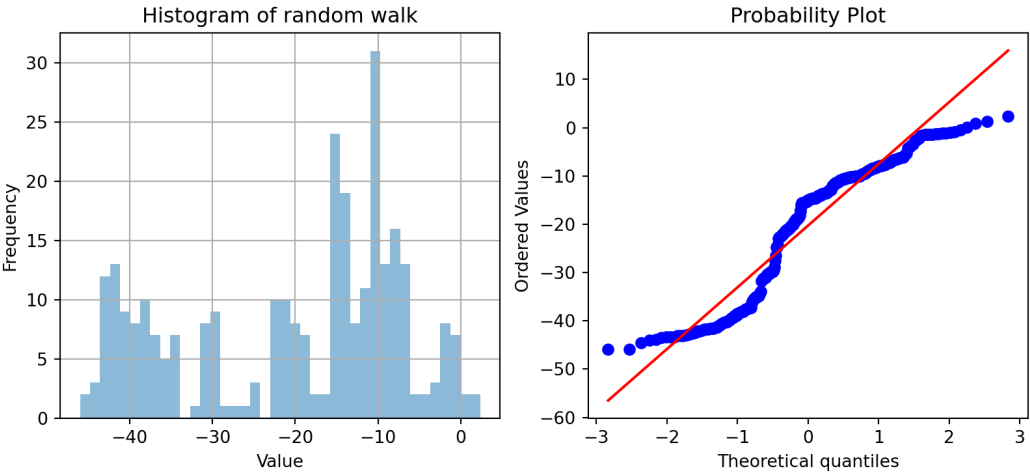
Random Walk Time Series

- Autocorrelation of white noise series dies slowly
- Partial autocorrealtion nonzero at one lag



Random Walk Time Series

Random walk series is not Normally distributed



Random Walk Time Series

Random walk time series are **non-stationary**

- Consider the **covariance** of a time series at lag k :

$$\gamma_k = Cov(y_t, y_{t+k})$$

- For a random walk, the increase in covariance is **unbounded in time**:

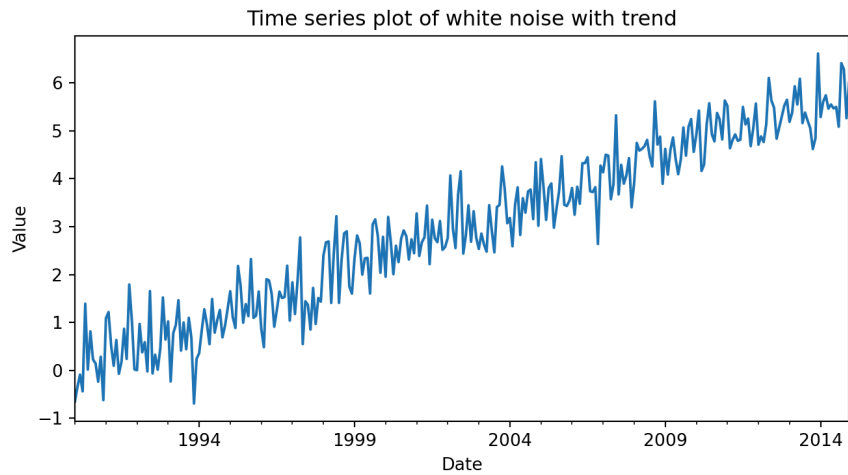
$$\gamma_k = Cov(y_t, y_{t+k}) = t\sigma^2 \rightarrow \infty \text{ as } t \rightarrow \infty$$

- Unbounded and time dependent variance make a random walk non-stationary

Time Series With Trend

Many real-world time series have a long-term **trend**

- A trend is a long term change in the mean value of the time series
- Typically model trend as linear, polynomial, non-parametric splines, etc.
 - *PROFIT* algorithm uses **generalized additive model (GAM)**
- Consider an example of a white noise series with a linear trend



Time Series With Trend

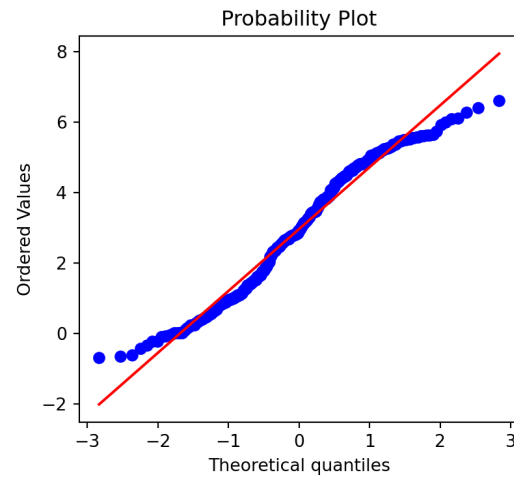
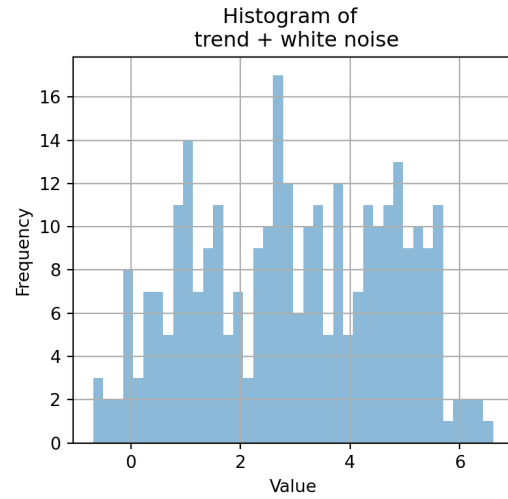
Trend models are not just strait lines

- Polynomial regression
- Piece-wise polynomial regression - e.g. splines
 - Used in *PROFIT algorithm*
 - A **generalized additive model**
- Local polynomial regression - e.g. LOESS
 - Used in *Statsmodels*

Time Series With Trend

Time series with trend are non-stationary

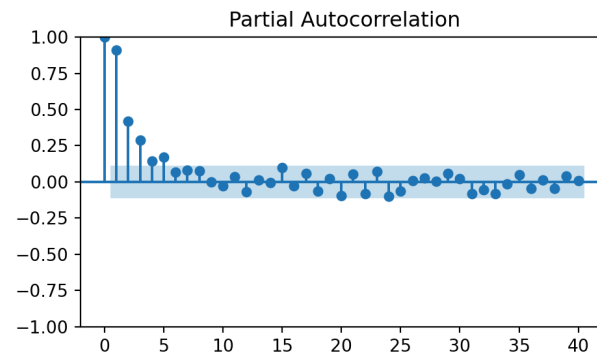
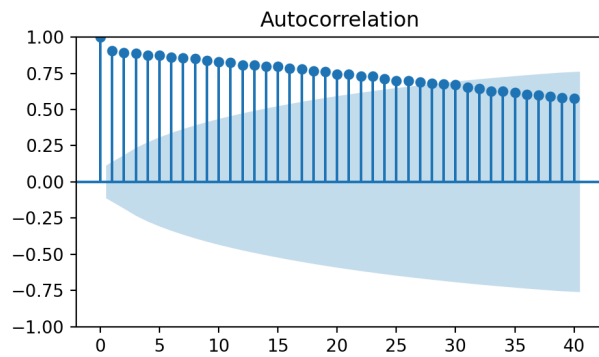
- Any time series with trend is non-stationary
 - *Mean and variance are dependent of window used to compute them*
- The distribution of even a white noise series with trend is non-Normal



Time Series With Trend

ACF and PACF are only properly defined for stationary series

- For non-stationary series, the ACF dies off slowly
 - *Integrative innovations lead to long-term dependency*
- The PACF dies off quickly with lag
- Example: ACF and PACF of the white noise series with trend



Time Series With Seasonal Effects

Many (most?) real-world time series have **seasonal effect**

- A seasonal effect has a measurable effect that occurs periodically
- Examples of seasonal events include:
 - *Day of the week*
 - *Last day of the month*
 - *Month of the year*
 - *Annual holiday*
 - *Option expiration date*
 - *Game day, e.g. Super Bowl*
 - *Electrical impulses in a heart - EKG*
 - *Orbits of planets*
 - ...
- Time series with seasonal effects are non-stationary
 - *Mean and variance depends of sample window*

Time Series With Seasonal Effects

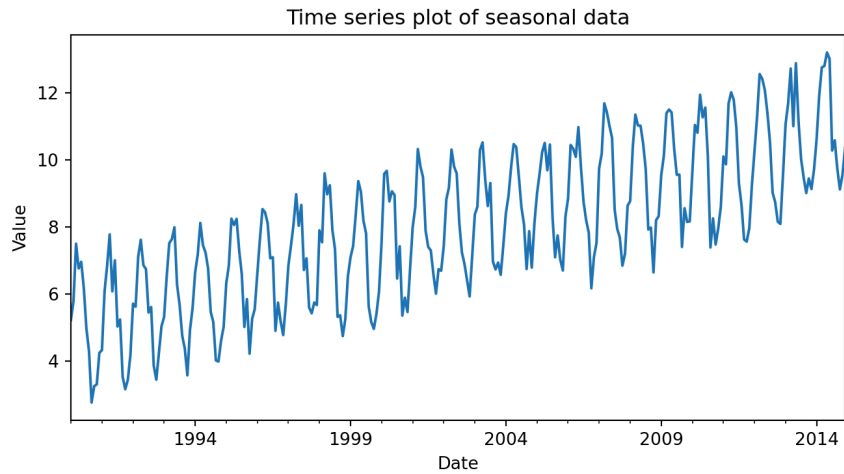
Use regression models for seasonal effects

- Simple regression model:
 - *Coefficient for each interval in period; e.g. 12 coefficients for monthly effects*
 - *Coefficient for specific effect - e.g. date of holiday*
- Basis function regression
 - *PROFIT algorithm* uses *Fourier basis functions*
 - A **generalized additive model**

Time Series With Seasonal Effects

Example of a time series with a seasonal effect

- A white noise series with trend and seasonal behavior
- The seasonal behavior is periodic



Time Series Models

Many types of time series models; we will only consider the most widely used

- Time series decomposition
- Exponential models
- ARIMA model
 - *Serially correlated components*
 - *Integrative component*

Time Series Decomposition

Two possible models for seasonal effects

- Goal, decompose the time series into its components
- The **Seasonal Trend decomposition model using Loess (STL)** model
 - *Uses a nonparametric regression model to decompose time series into components*
 - *Components are **seasonal (S)**, **trend (T)**, and the **residual (R)***
 - *Additive decomposition model*
 - *Multiplicative decomposition model*
- Differencing model

Time Series Decomposition

The **additive decomposition model** is expressed as as the sum of the components:

$$TS(t) = S(t) + T(t) + R(t)$$

- Used when seasonal effect is constant in time
- Examples: Physical process

Time Series Decomposition

The **Multiplicative decomposition model** is expressed as as the product of the components:

$$TS(t) = S(t) * T(t) * R(t)$$

The multiplicative form is can be hard to work with, so log transform to additive model

$$\log(TS(t)) = \log(S(t)) + \log(T(t)) + \log(R(t))$$

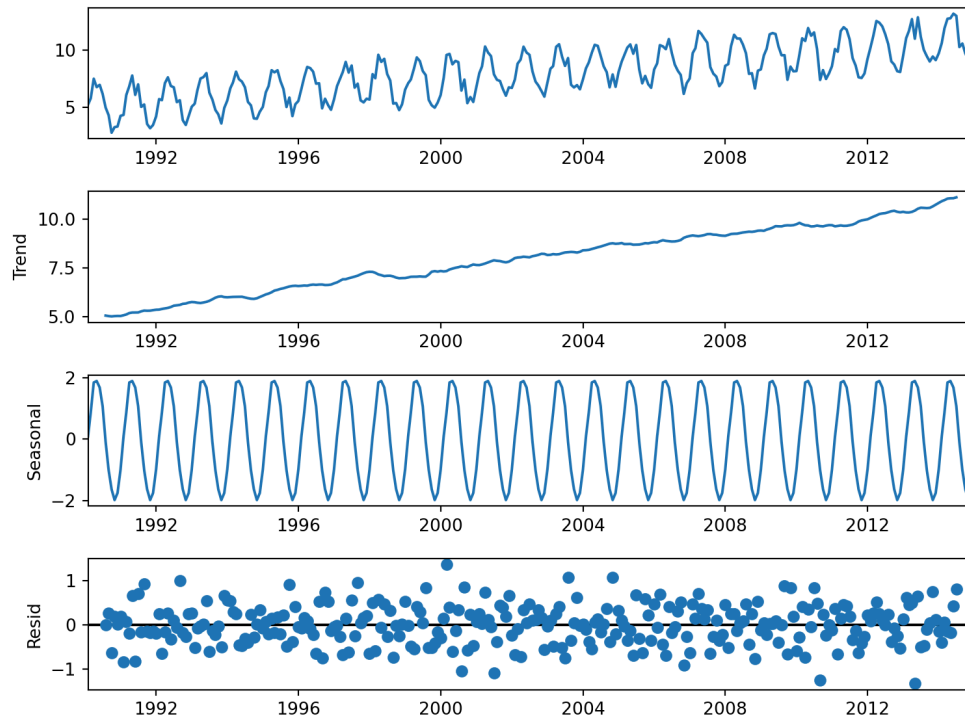
$$= S^l(t) + T^l(t) + R^l(t)$$

- Use when seasonal effect changes in time
- Example, economic time series

Time Series Decomposition

Example of additive STL decomposition of time series with linear trend and seasonal effect

- The original series plot is on top
- Notice the estimated trend is not a straight line; a result of noise
- Residuals are relatively small and **homoscedastic**



Time Series Difference Operators

Is there an alternative for dealing with trend?

How do we deal with random walks?

- **Difference operators** are useful for both cases
- Difference operators return the innovations

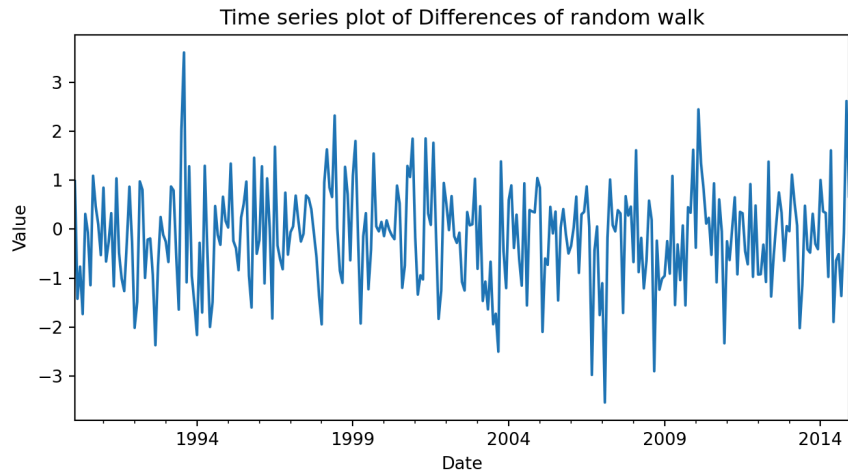
$$\nabla y_t = y_t - y_{t-1}$$

- Difference operators can be of any order in principle
 - *Typically use first order differences*
- Difference operator of order n computes a series n shorter than original

Time Series Difference Operators

Example of a first order difference operator applied to random walk

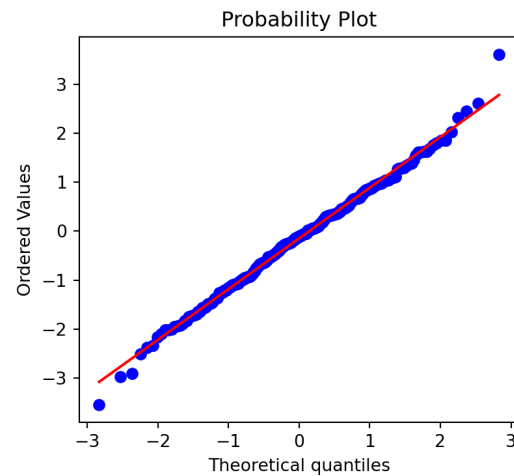
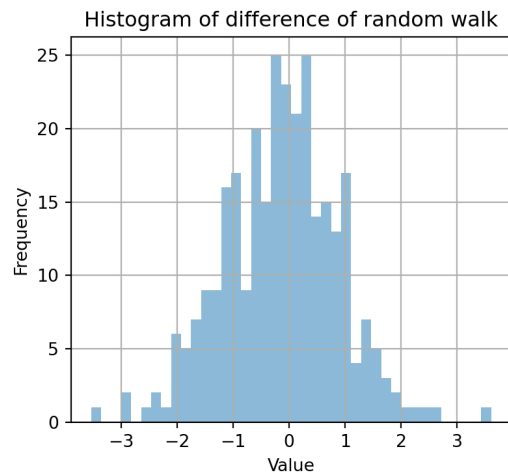
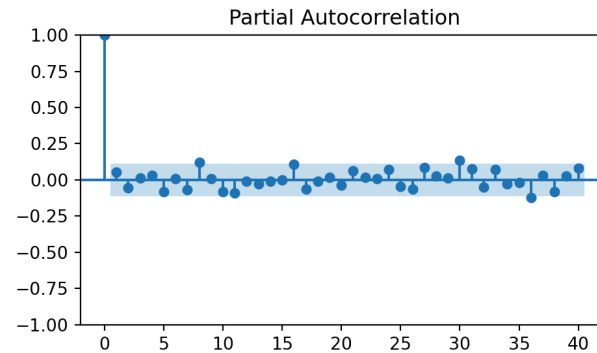
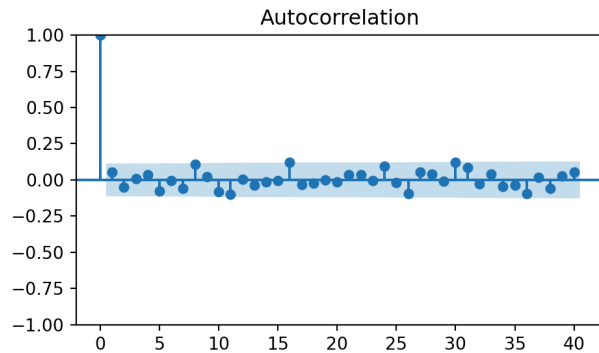
- The innovations look random
- Need to verify statistical properties



Time Series Difference Operators

Statistical properties of the difference series

- Compute the ACF and PACF
- The plots indicate the difference series is white noise



Time Series Forecasting Models

Forecasting is the goal of much of time series analysis

- Exponential models; extrapolation from simple smoothers
- ARIMA and SARIMAX models; time series linear models
- For comprehensive introduction see [Forecasting: Principles and Practice, Hyndman and Athanaosopoulos, 3rd edition, 2018](#), available as book or free online
- [Rob Hyndman's blog](#) is a source of many interesting ideas and example in time series analysis

Exponential Smoothing Models

Exponential smoothing models are simple and widely used

- Consider the simple first order model
- Set initial conditions:

$$s_0 = y_0$$

- The smoothed update is:

$$\begin{aligned} s_t &= \alpha y_t + (1 - \alpha)s_{t-1} \\ &= s_{t-1} + \alpha(y_t - s_{t-1}), \\ &\quad t > 0 \end{aligned}$$

- And, the smoothing coefficient is, $0 \leq \alpha \leq 1$
- But, model only works if no trend

Exponential Smoothing Models

Decay and exponential smoothing

- We can understand the smoothing parameter α in terms of a **decay constant**, τ

$$\alpha = 1 - e^{\left(\frac{\Delta T}{\tau}\right)}$$

- An innovation or shock has an effect for all future time
- Effect decays exponentially with time, ΔT

Exponential Smoothing Models

Can extend exponential smoothing model to accommodate trend

- Algorithm known as **double exponential smoothing** or **Holt-Winters double exponential smoothing**
- Update smoothed values and slope at each time step
- Start with initial values

$$s_1 = y_1$$
$$b_1 = y_2 - y_1$$

- Update relationships for both smoothed value and slope

$$s_t = \alpha y_t + (1 - \alpha)(s_{t-1} + b_{t-1})$$
$$b_t = \beta(s_t - s_{t-1}) + (1 - \beta)b_{t-1}$$

- Additional slope smoothing hyperparameter, $0 \leq \beta \leq 1$
- Use **third order** update includes seasonality in **Holt-Winters smoother**

Exponential Smoothing Models

Exponential smoothing models are useful for forecasting

- Forecast dependent on the choice of smoothing parameters
- Can forecast with first, second, third order models
- For second order model (with trend) the forecast m steps ahead is:

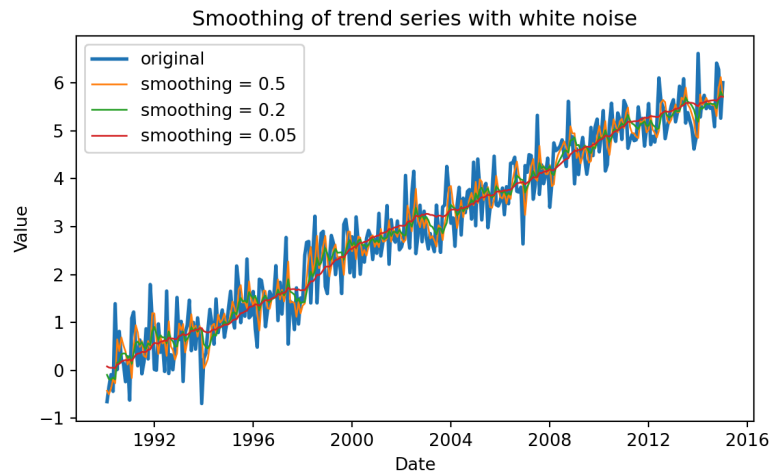
$$F_{t+m} = s_t + mb_t$$

- Third order update include seasonal terms
- Holt-Winters smoother is a **linear model!**

Exponential Smoothing Models

Example of smoothing trend plus white noise series

- Decreasing the smoothing parameter, α , increases smoothing
- Additionally, smooth trend
- Additional examples in [Statsmodels user documentation](#)



The ARIMA and SARIMAX Model

The **ARIMA model** is composed three components:

- **Autoregressive component (AR)** accounts for partial autocorrelation
 - *Serial correlation of observatons*
- **Integrative component (I)** accounts random walks and trend
- **Moving Average (MA)** accounts for autocorrelation
 - *Serial correlation of model error*
- **SARIMAX** model adds:
 - *Seasonal components (S)*
 - *Exogenous variables (X)*

The Autoregressive Model

Autoregressive model relates past observed values to the current value

- An autoregressive model of order p , $AR(p)$, uses the last p observations:

$$x_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + w_t$$

where

ϕ_k = model coefficient at lag k

w_t = white noise error at time t ; $\sim N(0, \sigma^2)$

y_t = observation at time t

- An AR process has the following properties:

- $\rho_0 = 1$ always
- $p_k = \phi_k$
- Number of nonzero PACF values = p
- A shock at any time will affect the result as $t \rightarrow \infty$

- AR model assume stationary time series

The Autoregressive Model

How can we understand the AR model?

- Consider an AR(2) model
- The value of y_t is a weighted sum of k previous values plus an error term

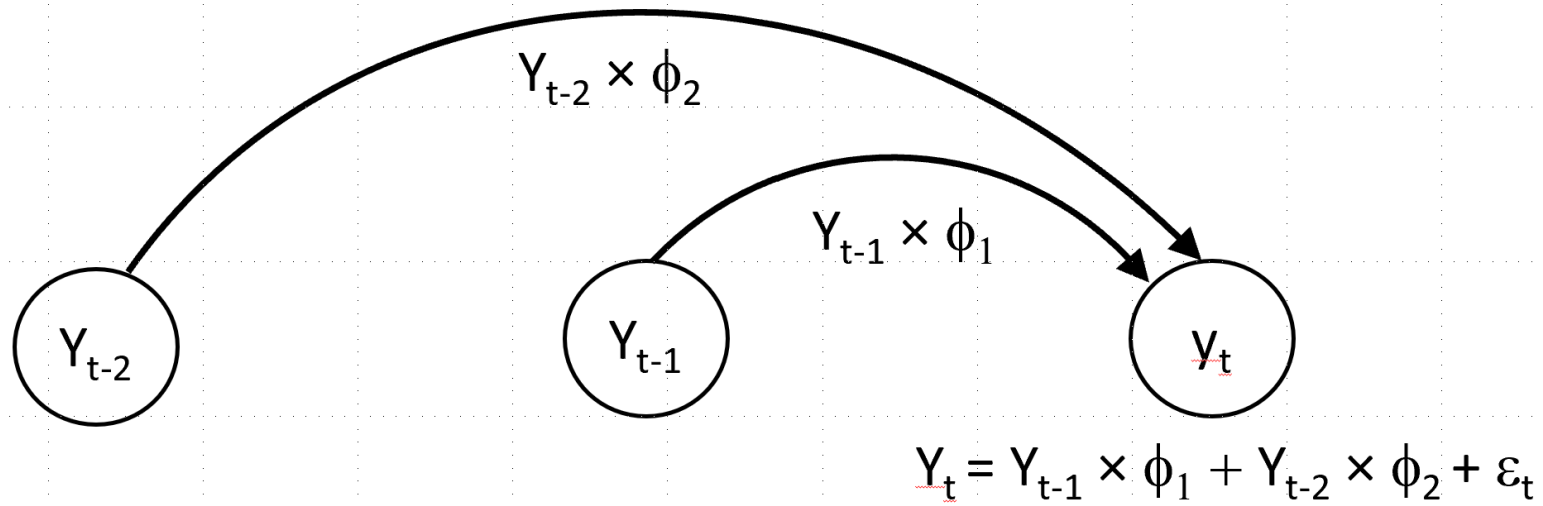


Illustration of the AR(2) model

The Autoregressive Model

How can we understand the AR model?

- Model matrix of AR(2) model

$$A = \begin{bmatrix} y_t, y_{t-1}, y_{t-2} \\ y_{t-1}, y_{t-2}, y_{t-3} \\ y_{t-2}, y_{t-3}, y_{t-4} \\ \vdots, \quad \vdots, \quad \vdots \\ y_2, \quad y_1, \quad y_0 \\ y_1, \quad y_0 \quad 0 \\ y_0, \quad 0, \quad 0 \end{bmatrix}$$

- AR model is a **linear model!**
- For coefficient vector, $\Phi = [\phi_1, \phi_2, \dots, \phi_p]$, solve linear system:

$$Y = A\Phi$$

The Autoregressive Model

We can rewrite the AR(1) model in terms of expectations:

$$\mathbb{E}(y_t) = \phi_t \mathbb{E}(y_{t-1}) + \mathbb{E}(\epsilon_t)$$

or

$$\mu = c + \phi_t \mu + 0$$

therefore

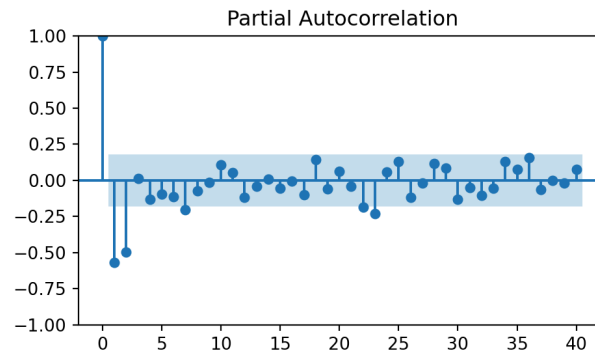
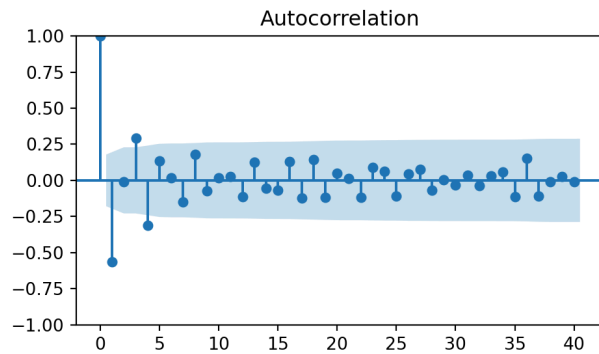
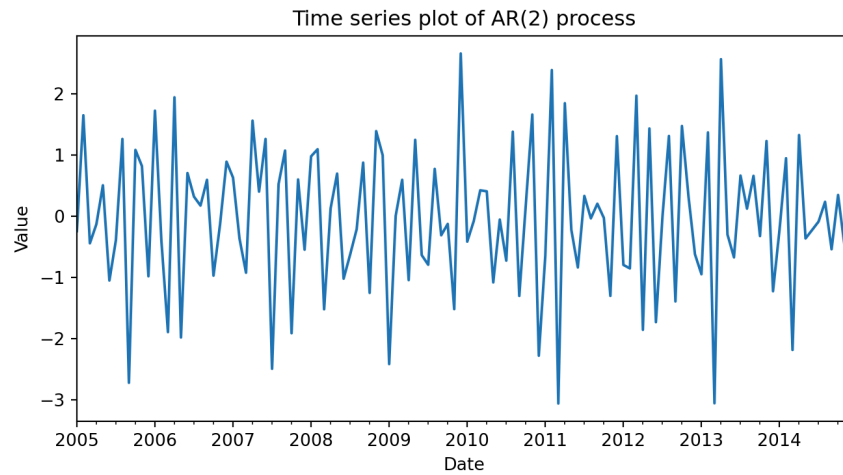
$$\mu = \frac{c}{1 - \phi_t^2}$$

- The AR model is unstable for the roots of the polynomial $1 - \phi_t^2$
- To be a stable AR process, $\phi_t^2 \leq 1$
- Violation of this condition leads to unstable model!

The Autoregressive Model

Example of AR(2) time series with coefficients = $(1.0, 0.75, 0.25)$:

- Time series looks a bit random
- But, notice the statistical properties; ACF, PACF
- PACF has 2 non-zero lag values, so $p = 2$



The Autoregressive Model

Example model summary for AR(2) model:

- Both AR coefficients are statistically significant
- Variance term is statistically significant

The Moving Average Model

- A **moving average** model of order q , $MA(q)$, uses the last q error terms or shocks:

$$y_t = \mu + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q}$$

where

θ_k = model coefficient at lag k

y_t = observation at time t

ϵ_t = innovation or error at time t ; $\sim N(0, \sigma^2)$

- An MA process has the following properties:
 - For autocorrelation, $\rho_0 = 1$ always
 - Number of $\rho_k \neq 0, = q, k > 0$
 - Shocks die off quickly in MA processes
- MA model assumes stationary time series

The Moving Average Model

How can we understand the MA model?

- Model matrix of MA(2) model

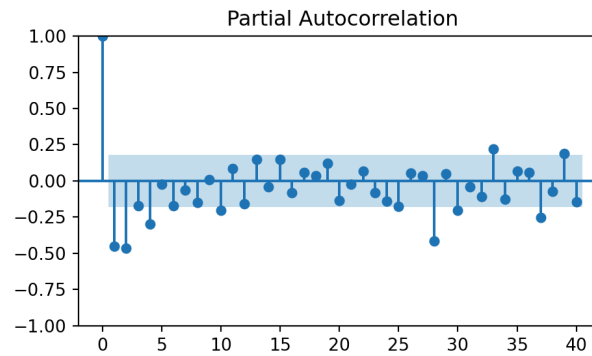
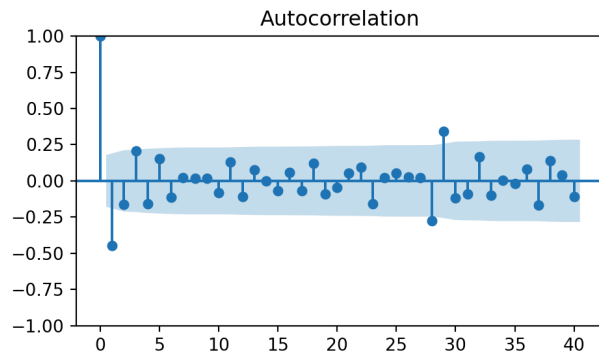
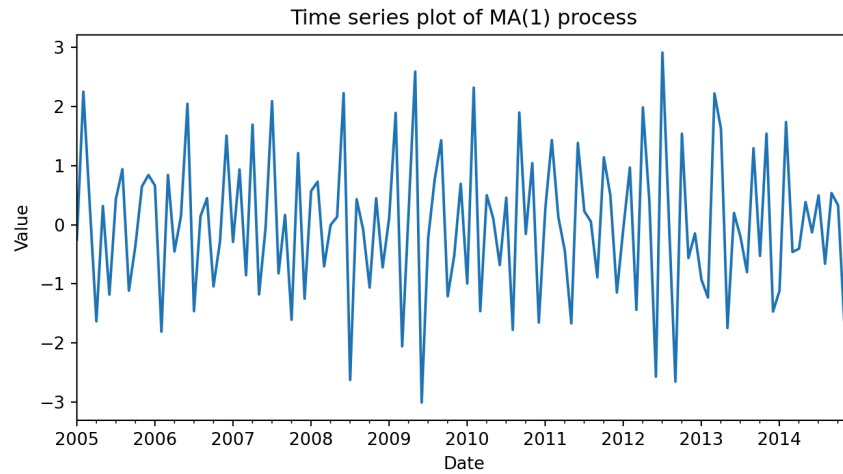
$$A = \begin{bmatrix} y_t, \epsilon_{t-1}, \epsilon_{t-2} \\ y_{t-1}, \epsilon_{t-2}, \epsilon_{t-3} \\ y_{t-2}, \epsilon_{t-3}, \epsilon_{t-4} \\ \vdots, \quad \vdots, \quad \vdots \\ y_2, \quad \epsilon_1, \quad \epsilon_0 \\ y_1, \quad \epsilon_0 \quad 0 \\ y_0, \quad 0, \quad 0 \end{bmatrix}$$

- MA model is a **linear model!**
- But, the value of ϵ_t depends on $[\epsilon_{t-1}, \epsilon_{t-2}, \dots, \epsilon_{t-q}]$
- The ϵ_k s are **unobservable!**
- So, fitting requires **nonlinear iteratively reweighted least squares**

The Moving Average Model

Example of an MA(1) model with coefficients $(1, -0.75)$

- The time series looks fairly random
- The ACF has 1 statistically significant nonzero lag value



The Moving Average Model

Example model summary for MA(1) model:

- The MA coefficient is statistically significant
- Notice that true value is within the confidence interval
- Confidence interval is wide

Autoregressive Moving Average Model

We can combine AR and MA terms to create the **autoregressive moving average (ARMA)** model of order (p, q) :

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \dots + \theta_q \epsilon_{t-q}$$

- Fit ARMA model by solving a nonlinear equation:

$$y_t - \phi_1 y_{t-1} - \phi_2 y_{t-2} - \dots - \phi_p y_{t-p} = \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \dots + \theta_q \epsilon_{t-q}$$

- Can write as polynomial equation in terms of coefficient vectors $\Phi = [1, \phi_1, \phi_2, \dots, \phi_p]$, $\Theta = [1, \theta_1, \theta_2, \dots, \theta_q]$:

$$(1 - \Phi)Y = \Theta\epsilon$$

- Model is linear in coefficients
- ARMA model assumes stationary time series

The ARIMA Model

The integrative model addresses certain non-stationary components of a time series

- Random walks
- Trends
- Based on difference operator
 - *Typically first order difference*
 - *Can be higher order*
 - *Is deterministic, no model coefficient to estimate*

The ARIMA Model

The **autoregressive integrative moving average (ARIMA)** model includes AR, integrative and MA terms

- The order of an ARIMA is specified as (p,d,q)
 - *p is the AR order*
 - *d is the order of differencing*
 - *q is the MA order*
- The integrative term helps transforms trend and random walks to stationary process
- Does not account for seasonal effect
- For difference values, ∇y_t , formulate as:

$$(1 - \Phi)\nabla Y = \Theta\epsilon$$

Seasonal Models

Several possible seasonal models

- Seasonal effects can be periodic or single event (e.g. holiday, game day, etc.)
- Linear regression model to find effect for each time step in period
- STL decomposition
- SARIMAX, the S term
- Each model requires:
 - *Known period of the cycle or time of seasonal event*
 - *Additive or logarithmic transformation*

SARIMAX Model

The SARIMAX model adds seasonal and exogenous terms

- ARIMA terms are same, (p,d,q)
- Seasonal terms:
 - *ARIMA model, order (P,D,Q,S)*
 - *Must specify period, S*
- Order of SARIMAX model is specified as $(p,d,q)(P,D,Q,S)$
- See [Statsmodels State Space User Guide](#) for more details and examples

SARIMAX Model

The SARIMAX model (with no exogenous variables) is formulated

$$\phi_p(Y)\phi_P(Y^*)\nabla^d\nabla^D y_t = A(t) + \theta_q(Y)\theta_Q(Y^*)$$

- $\phi_p(Y)$ and $\phi_P(Y^*)$ are the AR polynomials non-seasonal and seasonal terms
- $\theta_q(Y)$ and $\theta_Q(Y^*)$ are the MA polynomials non-seasonal and seasonal terms
- ∇^d and ∇^D are the non-seasonal and seasonal differencing operators
- $A(t)$ is the trend term

SARIMAX Model

SARIMAX model can include exogenous variables, x , leading to a new system of equations”:

$$y_t = \beta_t x_t + \mu_t$$
$$\phi_p(Y)\tilde{\phi}_P(Y^*)\nabla^d\nabla^D\mu_t = A(t) + \theta_p(Y)\tilde{\theta}_P(Y^*)$$

- Time series model for latent variable, μ_t
- μ_t acts as the intercept term for the regression model for x_t
- The coefficient vector, β , contains the **effect sizes** for the exogenous variables

Evaluating and Comparing Time Series Models

How can we evaluate time series models?

- RMSE; compare forecast to actual values
- Could use log-likelihood; $\log(p(X|\theta))$
 - $\theta = \text{model parameters}$
 - Use **score function** $= -2 \log(\text{likelihood}) = -2 \log(p(X|\theta))$
- But, score will decrease with model complexity
- Need to adjust for number of model parameters
 - We always prefer simpler models; fewer parameters to learn
 - **Akai Information Criteria (AIC)**
 - **Bayes Information Criteria (BIC)**

Evaluating and Comparing Time Series Models

Akai Information criteria, AIC

$$AIC = 2k - 2 \ln(\hat{L})$$

where

\hat{L} = the likelihood given the fitted model parameters $\hat{\theta} = p(x|\hat{\theta})$

x = observed data

k = number of model parameters

- AIC penalizes the score function for the complexity of the model by $2k$
- Model with lowest AIC is best

Evaluating and Comparing Time Series Models

Bayes Information criteria, BIC

$$BIC = \ln(n) k - 2 \ln(\hat{L})$$

where

$\hat{L} =$ the likelihood given the fitted model parameters $\hat{\theta} = p(x|\hat{\theta})$

$x =$ observed data

$k =$ number of model parameters

$n =$ number of observations

- BIC penalizes the score function for the complexity of the model, k
- BIC adjusts for number of samples used to learn the k model parameters
- Model with lowest BIC is best
- BIC is often preferred to AIC for time series models

Evaluating and Comparing Time Series Models

Can compare and select models using BIC or AIC

- Backwards step-wise model selection

1. Start with initial order of the model; e.g. $(p, d, q)(P, D, Q, S)$

2. Fit (learn) the model parameters

3. compute the BIC, and if reduced consider this a better model

4. Reduce the order of one of the model components

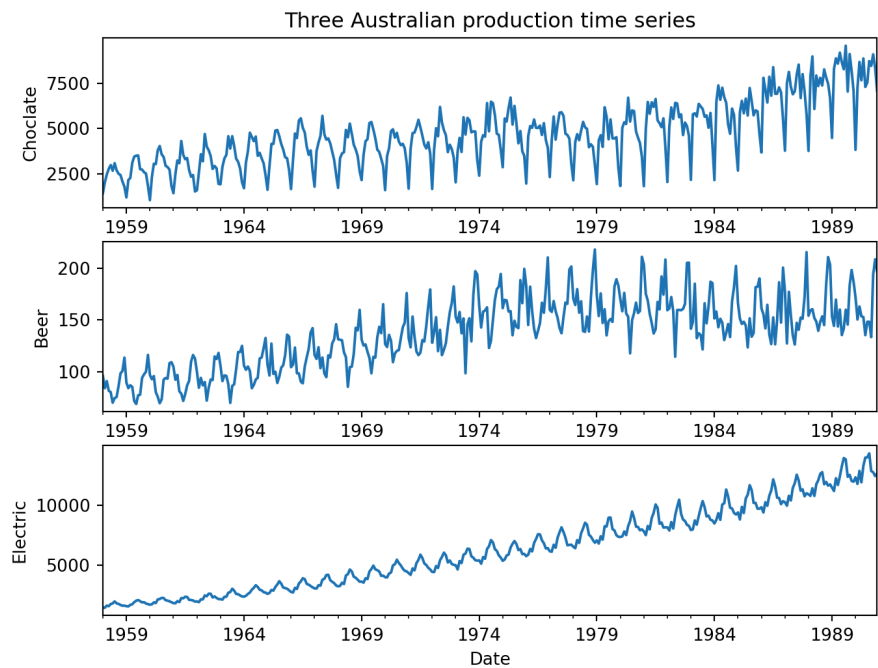
5. Repeat steps 2, 3 and 4 until no further improvement

- Tips for comparing models:

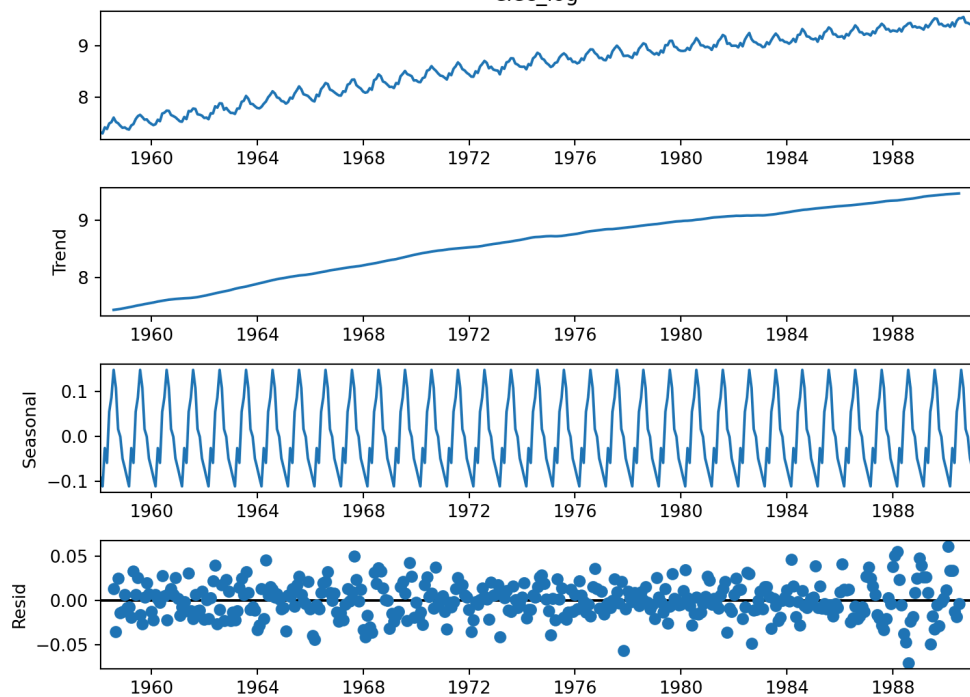
- BIC and AIC are approximations; small changes (3rd or 4th decimal) are not important
- If close tie for best model pick the simpler (lower order) case
- Often best to consider integrative terms, d and D , separately

SARIMAX Example

Example: 3 time series of Australian production



elec_log



SARIMAX Example

Example of SARIMAX model of order $(0,1,1)(0,1,2,12)$ for monthly electric production series

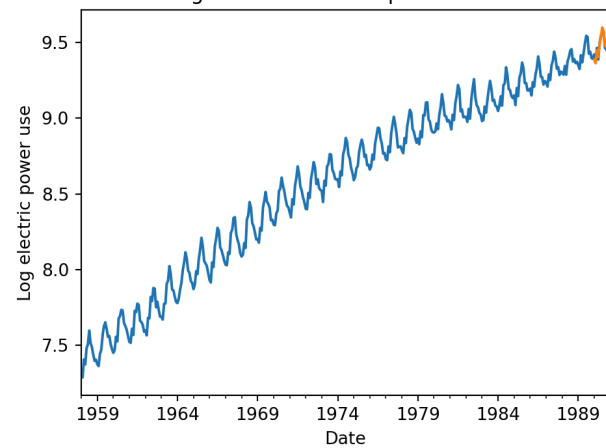
- Model selected by backwards step-wise method
- First order model integrative term and MA(1)
- First order model integrative term and MA(1) for period 12 seasonality

SARIMAX Example

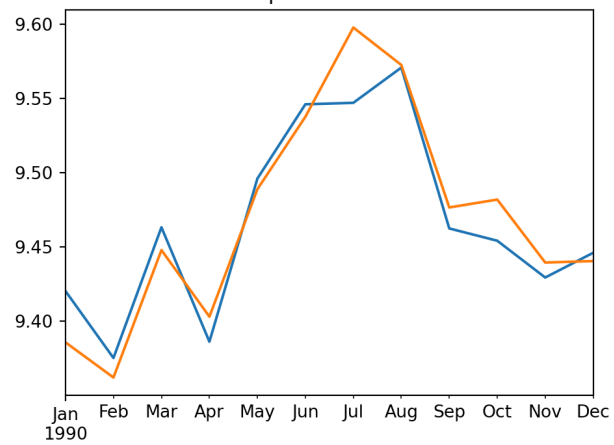
Predictions for the last 12 months of the time series

```
prediction = pd.Series(best_model.predict(n_periods=12),
                        index = pd.date_range(start = '1990-01-31', end = '1990-12-31', freq = 'M'))
```

Full log electric use with predicted values



Log electric use for 12 months
with predicted values

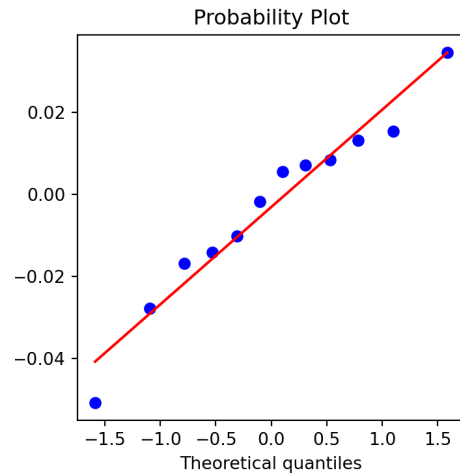


SARIMAX Example

Residuals of the predictions

```
residuals = CBE.elec_log['1990-01-31':] - prediction

fig, ax = plt.subplots(nrows=1, ncols=1, figsize=(4, 4))
_ = ss.probplot(residuals, plot = ax);
plt.show()
```



Summary

Fundamental elements of time series

- Fundamental components which cannot be predicted
 - *White noise*
 - *Random walks*
- Autocorrelation and partial autocorrelation
- Trend
- Seasonal components
- Stationarity properties; Dicky Fuller test

Summary

Time series models must account for serial correlation

- Exponential models; e.g. Holt-Winters
- Second order accounts for trend
- Third order accounts for trend and seasonal

Summary

Time series models must account for serial correlation

- e.g. ARIMA and SARIMAX
- AR components for serial correlation of values
- MA components for serial correlation of errors
- Integrative components for random walk and trend, I
- Seasonal, (P,D,Q,S)
- Exogenous variables, X

Summary

Evaluation and model comparison

- RMSE
- AIC and BIC, penalize score function for model complexity
- Use BIC (or AIC) to perform backwards step-wise model selection