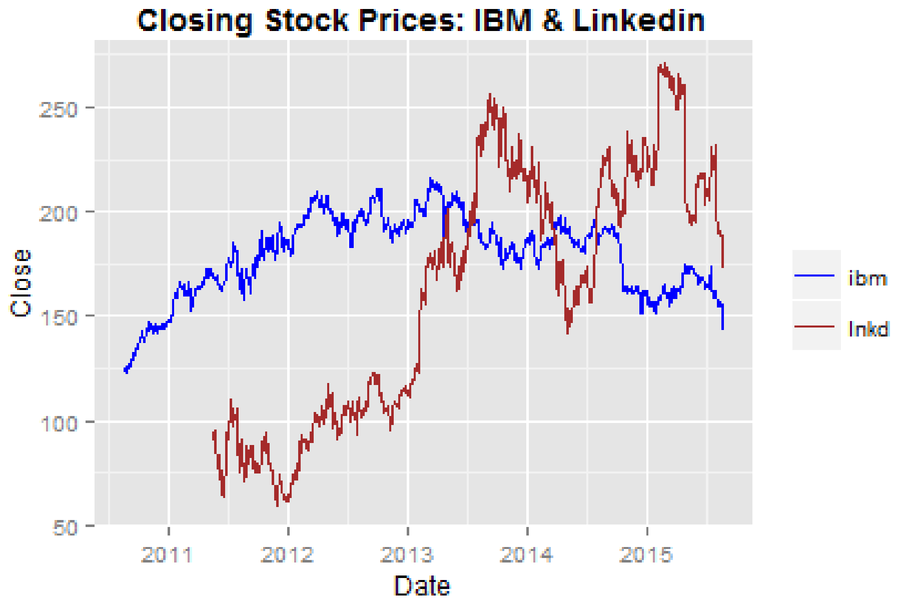
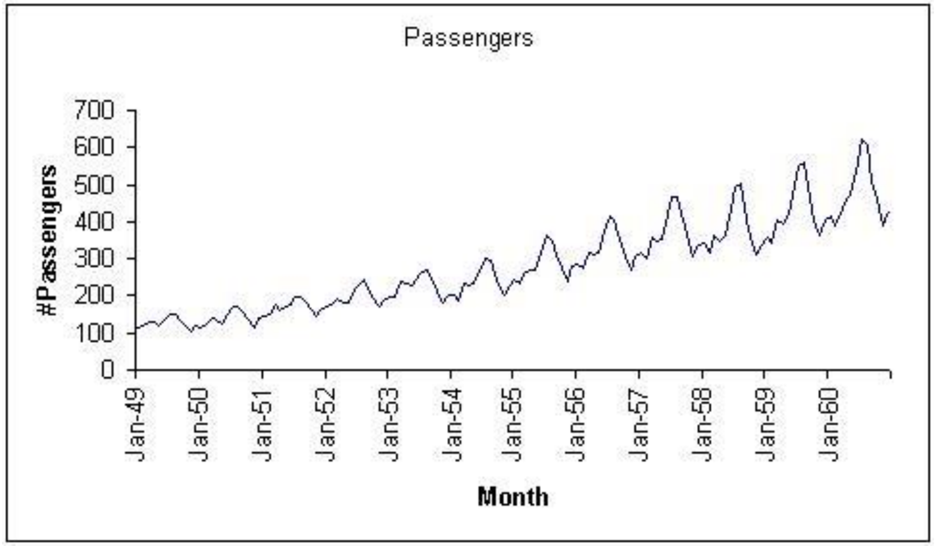
DigitalHouse >

DATA SCIENCE

Séries Temporais I

Definição Time Series



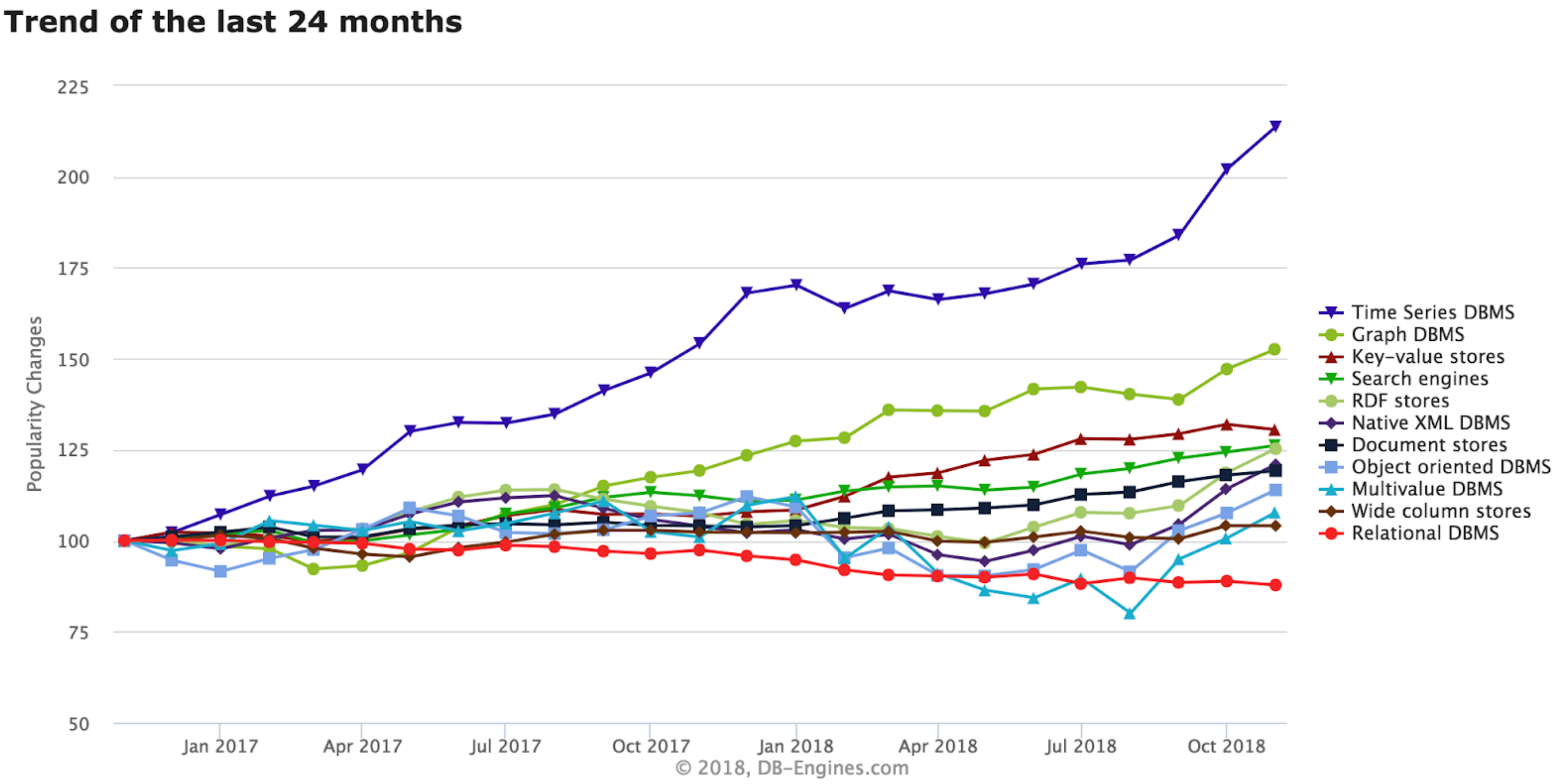


Podemos entender Time Series como qualquer tipo de dados/observações que contém informações de **sequência ordenadas pelo tempo entre suas observações**. Ou seja, uma série de pontos de dados indexados (listados ou representados graficamente) na ordem do tempo **de maneira uniforme**.

IoT + *Wearables -> BIG DATA*

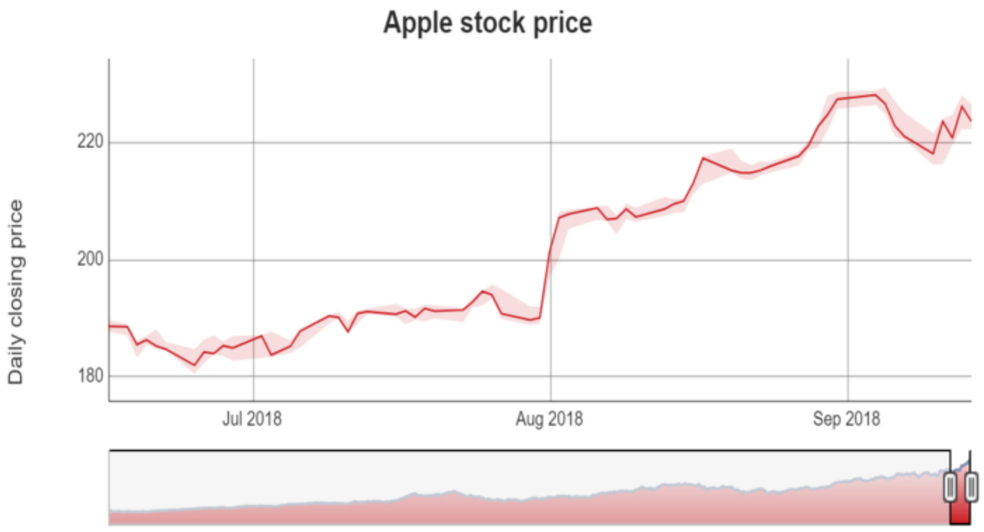
DigitalHouse>
Coding School

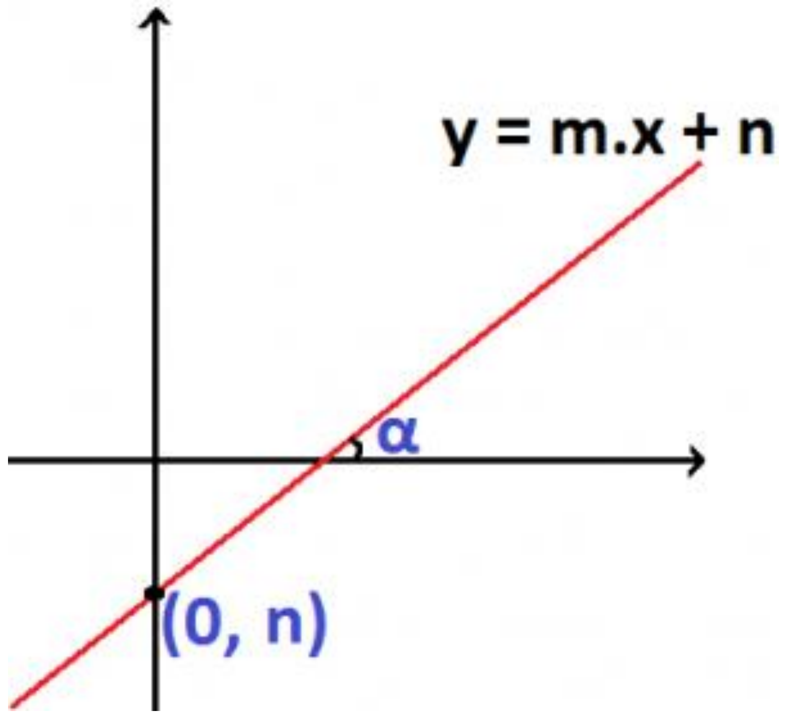
Time Series DBMS



https://db-engines.com/en/ranking/time+series+dbms

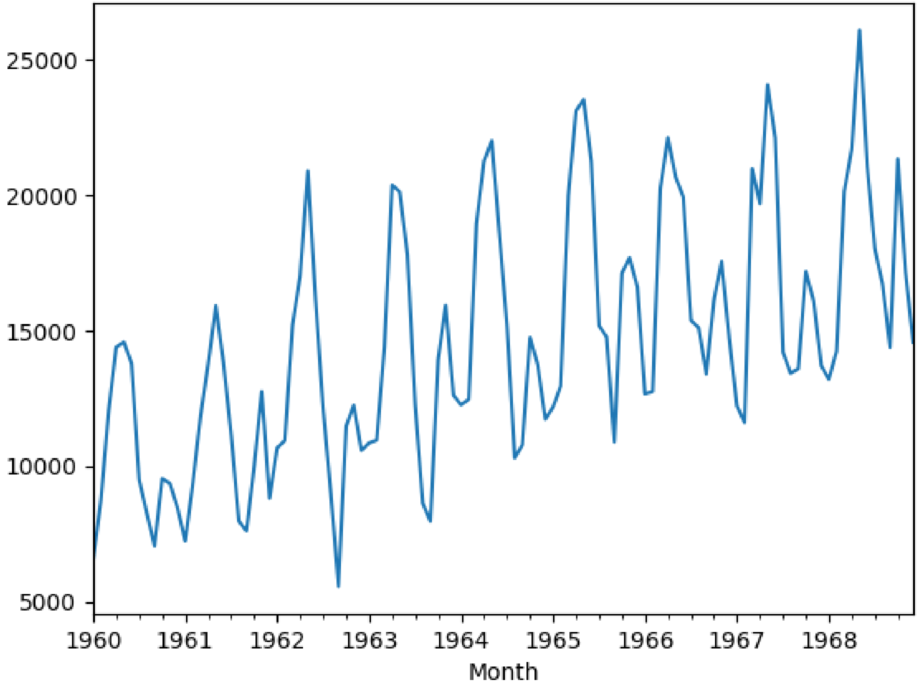
Um ponto de Atenção!





Qual a diferença fundamental entre estes dois gráficos?.

Univariate Time Series

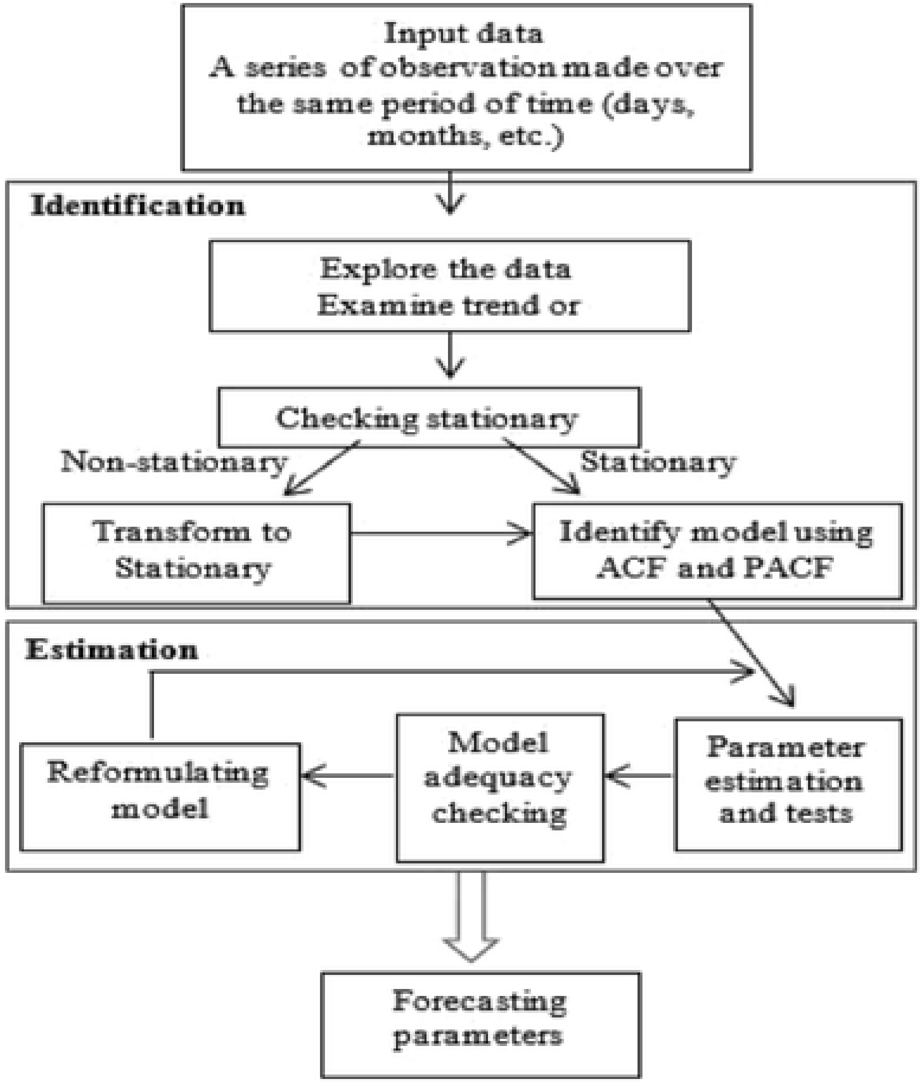


Modelos de Séries Temporais Univariados

Séries Temporais Univariadas. O termo "série temporais univariadas" refere-se às séries temporais que consistem de observações únicas (escalares) registradas sequencialmente durante incrementos iguais de tempo.

Identificação Modelo de Box-Jenkins

Identificação Modelo de Box-Jenkins



https://link.springer.com/article/10.1007/s40808-016-0164-0

6.4.4.6. Identificação do Modelo de Box-Jenkins

Estacionariedade e Sazonalidade	O primeiro passo no desenvolvimento de um modelo de Box-Jenkins é determinar se a série é estacionária e se existe qualquer sazonalidade significativa que precisa ser modeladda.
Detectando Estacionariedade	A estacionariedade pode ser avaliada de um gráfico de sequência de execução. O gráfico de sequência de execução deverá se mostrar constante em localização e escala. Ela pode também ser detectada de um gráfico de autocorrelação. Especificamente, a não estacionariedade é frequentemente indicada por um gráfico de autocorrelação com um decaimento muito lento.
Detectando sazonalidadae	Sazonalidade (ou periodicidade) pode ser geralmente estimada de umgráfico de autocorrelação, um gráfico da subsérie sazonal, ou um gráfico espectral.
Diferenciação para se atingir a estacionariedade	Box and Jenkins recomendaram a abordagem de diferenciação para se atingir a estacionariedade. Entretanto, ajustar uma curva e subtrair os valores ajustados dos dados originais pode também ser usado no contexto dos modelos de Box-Jenkins.
Diferenciação sazonal	No estágio de identificação do modelo, nossa meta é detectar sazonalidade, se existir, e identificar a ordem dos termos de média móvel sazonal e dos termos autoregressivos. Para muitas séries, o período é conhecido e um único termo de sazonalidade é suficiente. Por exemplo, para dados mensais tipicamente incluiríamos um termo AR 12 sazonal ou um termo MA 12 sazonal. Para modelos Box-Jenkins, não removemos explicitamente a sazonalidadae antes de ajustar o modelo. Em vez disso, incluímos a ordem dos termos sazonais na especificação do modelo apara a estimação do software ARIMA. Entretanto, pode ser útil aplicar uma diferença sazonal aos dados e gerar novamente os gráficos de autocorrelação parcial e autocorrelação. Isto pde ser útil na identificação do componente não sazonal do modelo. Em alguns casos, a diferenciação sazonal pode remover a maioria ou todos os efeitos de sazonalidade.

http://www.bertolo.pro.br/MetodosQuantitativos/Previsao/pmc446.htm

Identificação Modelo de Box-Jenkins

Article Talk | Read Edit View history | Search Wikipedia

Box–Jenkins method

From Wikipedia, the free encyclopedia

In time series analysis, the **Box–Jenkins method**,[1] named after the statisticians George Box and Gwilym Jenkins, applies autoregressive moving average (ARMA) or autoregressive integrated moving average (ARIMA) models to find the best fit of a time-series model to past values of a time series.

Contents [hide]

1 Modeling approach
2 Box–Jenkins model identification
2.1 Stationarity and seasonality
2.1.1 Detecting stationarity
2.1.2 Detecting seasonality
2.1.3 Differencing to achieve stationarity
2.1.4 Seasonal differencing
2.2 Identify *p* and *q*
2.2.1 Autocorrelation and partial autocorrelation plots
3 Box–Jenkins model estimation
4 Box–Jenkins model diagnostics
4.1 Assumptions for a stable univariate process
5 References
6 Further reading
7 External links

Modeling approach [edit]

The original model uses an iterative three-stage modeling approach:

1. *Model identification and model selection*: making sure that the variables are stationary, identifying seasonality in the dependent series (seasonally differencing it if necessary), and using plots of the autocorrelation (ACF) and partial autocorrelation (PACF) functions of the dependent time series to decide which (if any) autoregressive or moving average component should be used in the model.
2. *Parameter estimation* using computation algorithms to arrive at coefficients that best fit the selected ARIMA model. The most common methods use maximum likelihood estimation or non-linear least-squares estimation.
3. *Statistical model checking* by testing whether the estimated model conforms to the specifications of a stationary univariate process. In particular, the residuals should be independent of each other and constant in mean and variance over time. (Plotting the mean and variance of residuals over time and performing a Ljung–Box test or plotting autocorrelation and partial autocorrelation of the residuals are helpful to identify misspecification.) If the estimation is inadequate, we have to return to step one and attempt to build a better model.

The data they used were from a gas furnace. These data are well known as the Box and Jenkins gas furnace data for benchmarking predictive models.

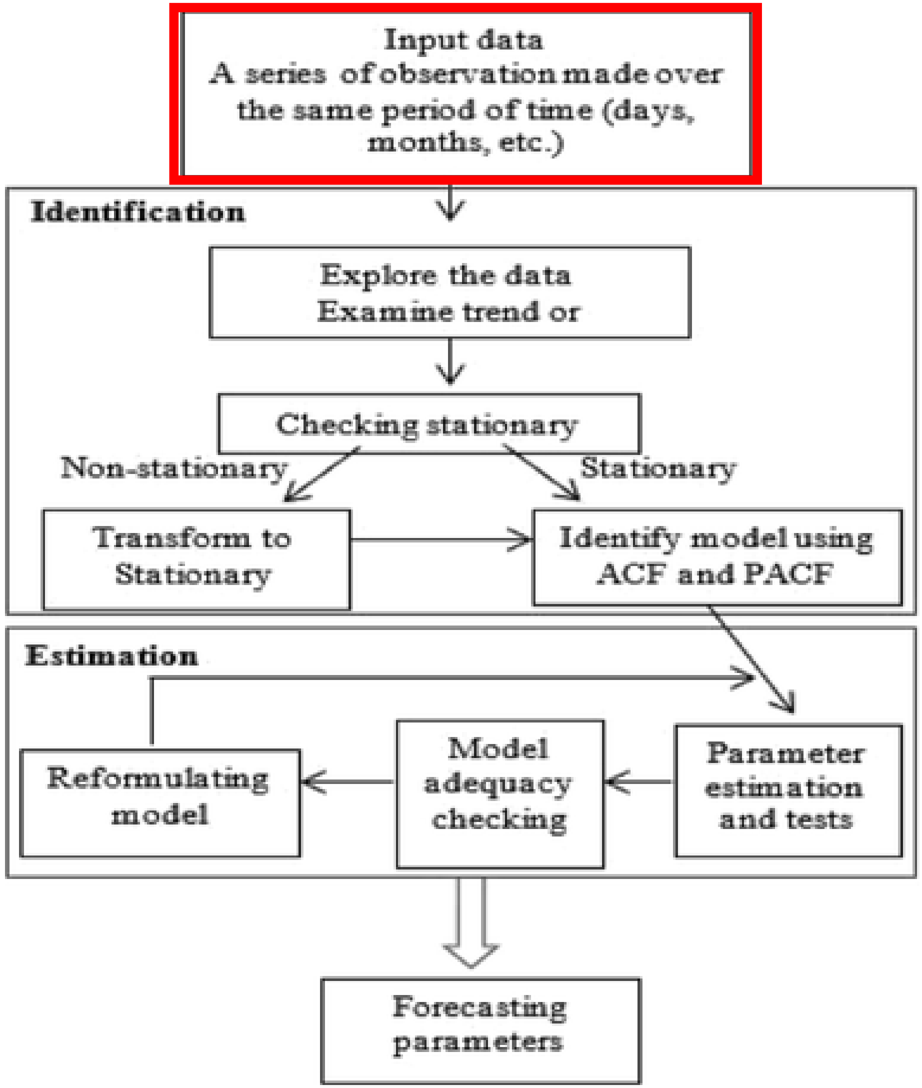
Commandeur & Koopman (2007, §10.4)[2] argue that the Box–Jenkins approach is fundamentally problematic. The problem arises because in "the economic and social fields, real series are never stationary however much differencing is done". Thus the investigator has to face the question: how close to stationary is close enough? As the authors note, "This is a hard question to answer". The authors further argue that rather than using Box–Jenkins, it is better to use state space methods, as stationarity of the time series is then not required.

https://en.wikipedia.org/wiki/Box%E2%80%93Jenkins_method

Passo 1

Integridade dos Dados Temporais

Identificação Modelo de Box-Jenkins



https://link.springer.com/article/10.1007/s40808-016-0164-0

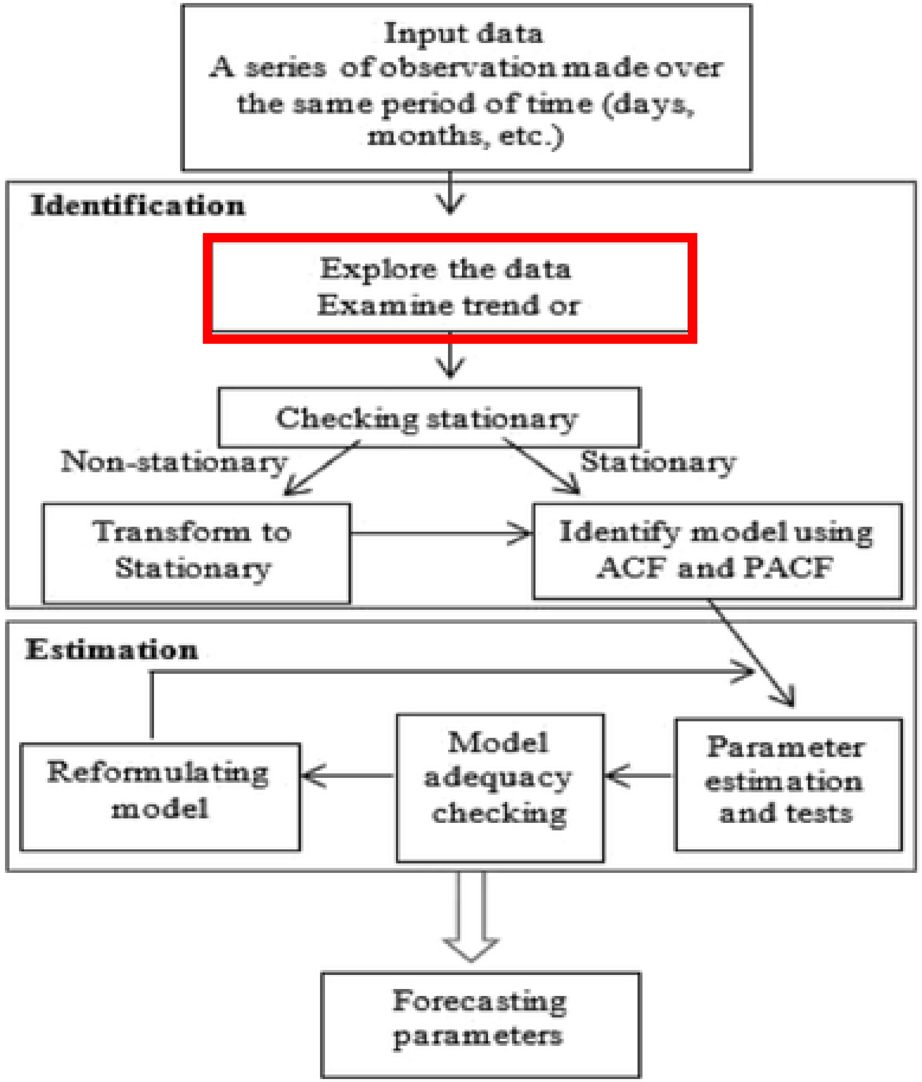
Passo 1

- Garantir que de fato seus dados correspondem a uma série temporal e que temos todos dados “preenchidos” e em um mesmo intervalo de tempo. Caso isto não seja possível pode ser necessário utilizar alguma técnica de interpolação.

Passo 2

Explorar e Analisar Tendências

Identificação Modelo de Box-Jenkins

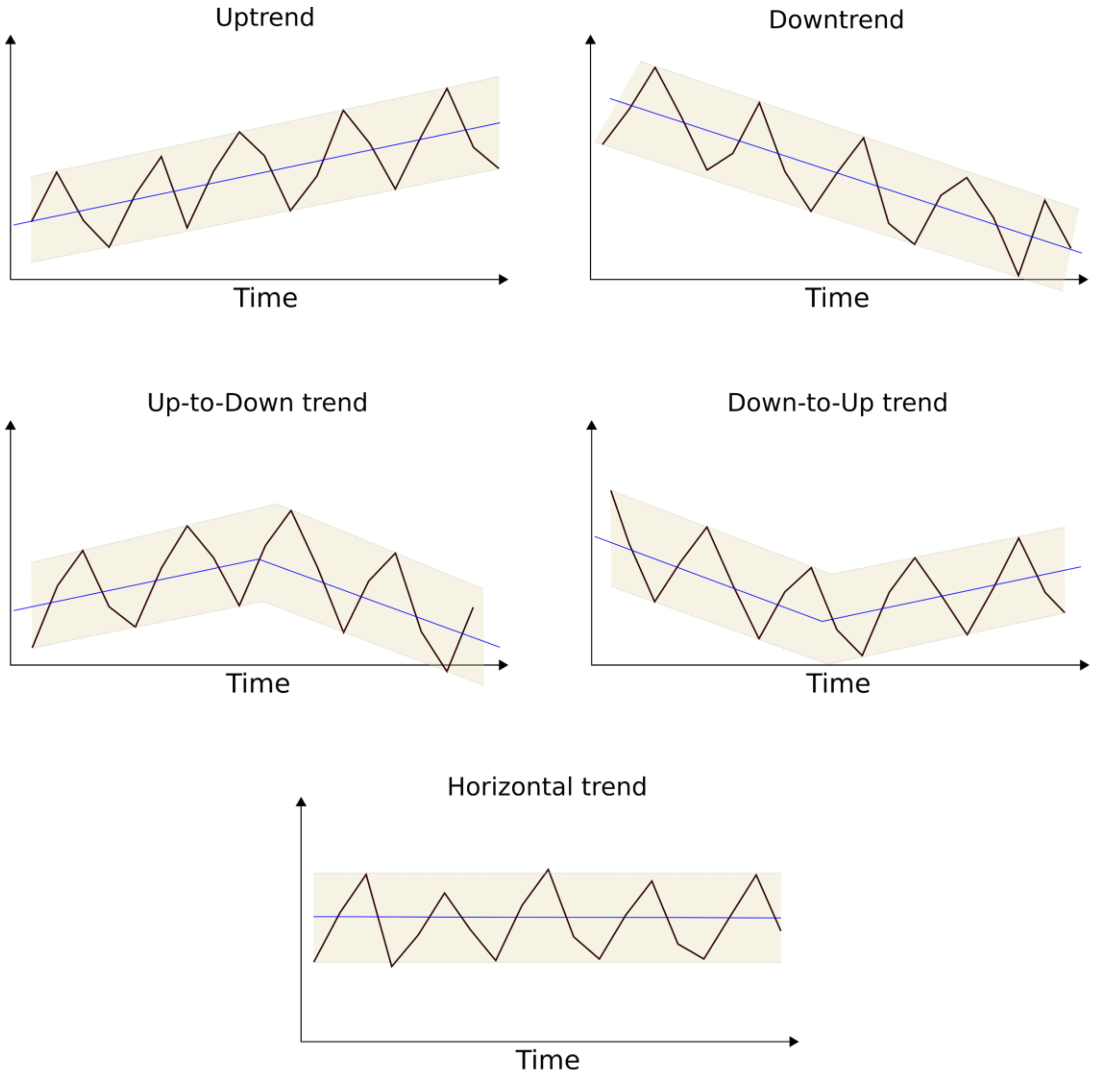


https://link.springer.com/article/10.1007/s40808-016-0164-0

Passo 1

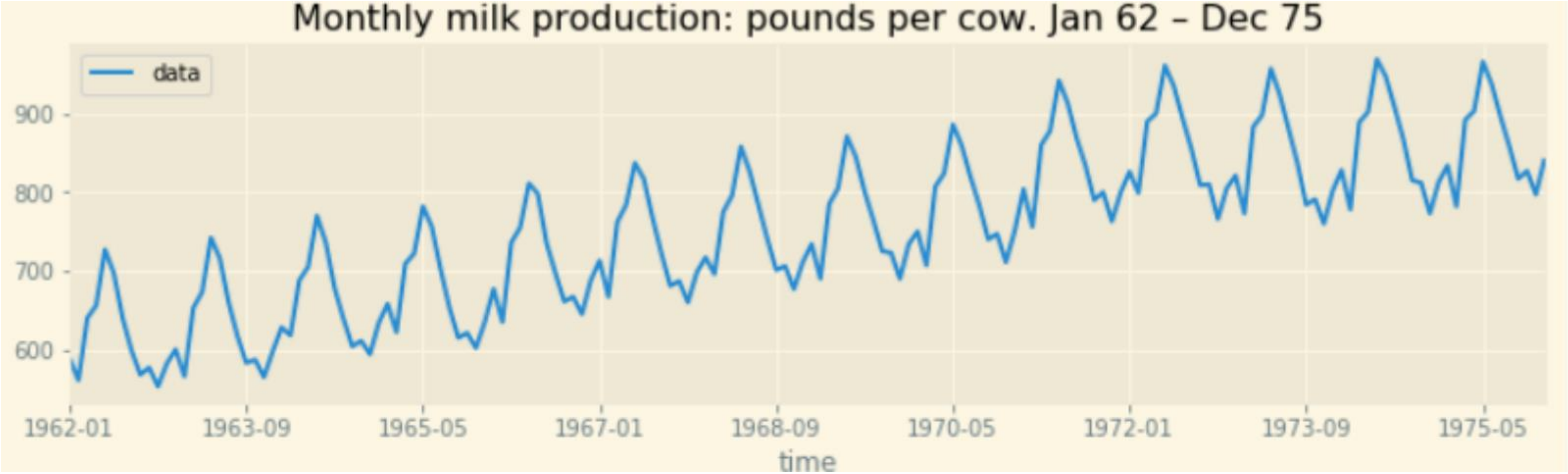
- Explorar os dados em questão e verificar se há alguma tendência nos dados.

Tendência



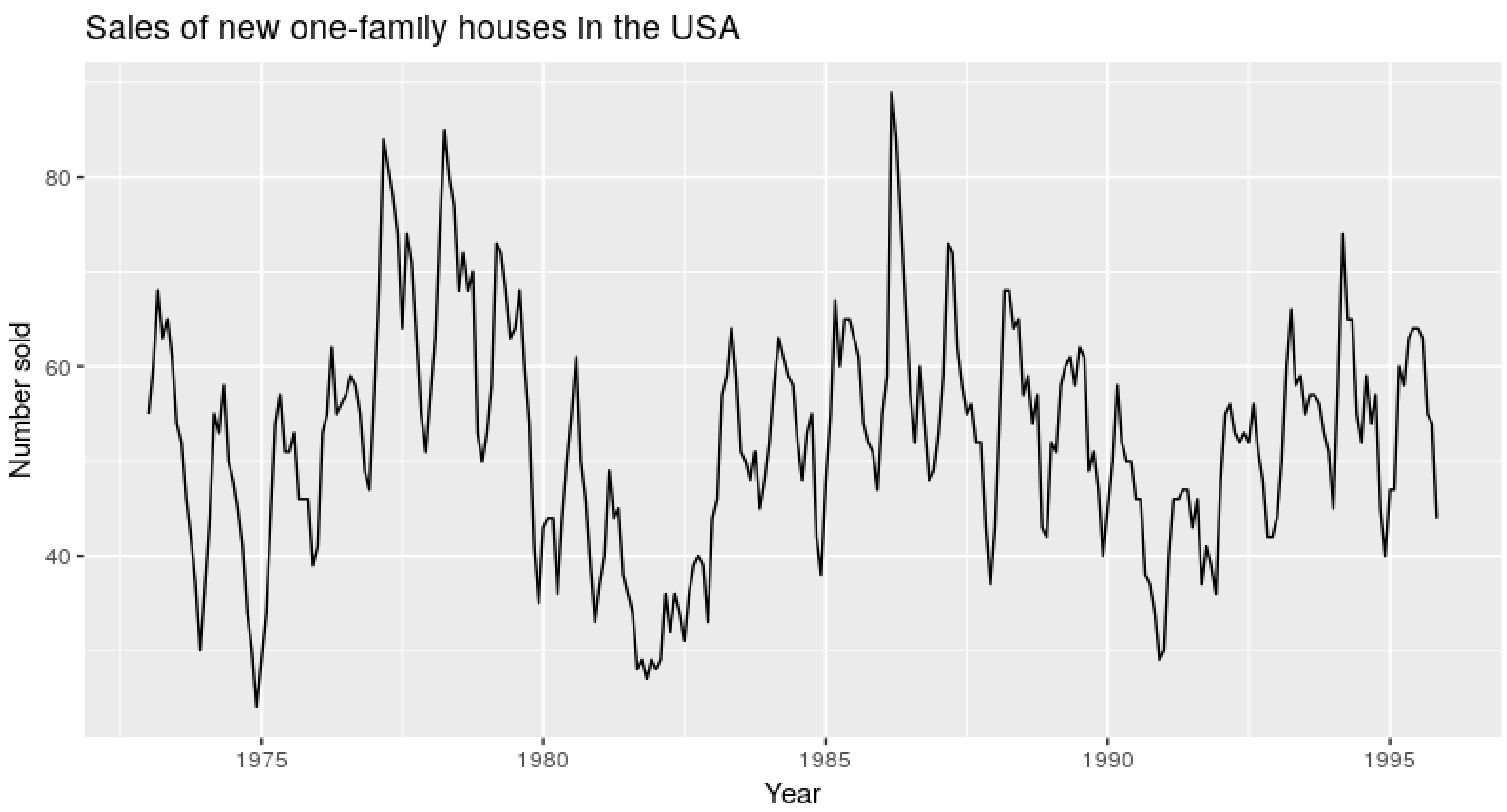
https://medium.com/better-programming/fundamentals-of-time-series-data-and-forecasting-15e9490b2618

Sazonalidade



Sazonalidade indica um intervalo regular em que o comportamento dos dados se repete.

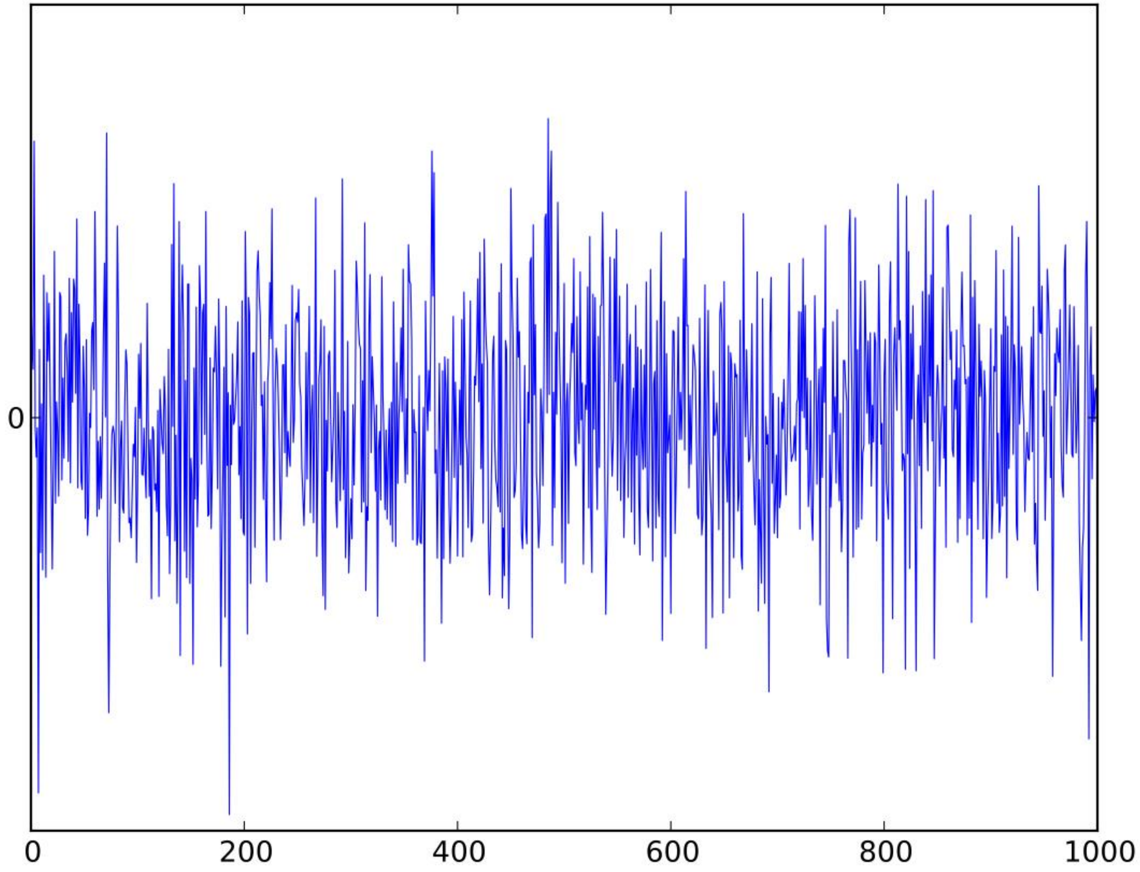
Seasonal x Cyclical



https://medium.com/better-programming/fundamentals-of-time-series-data-and-forecasting-15e9490b2618

Importante fazer a diferenciação em relação a **Ciclos** – que mostram alguma repetição ao longo do tempo, e me maior escala, mas que não são completamente regulares.

Ruído – *White Noise*

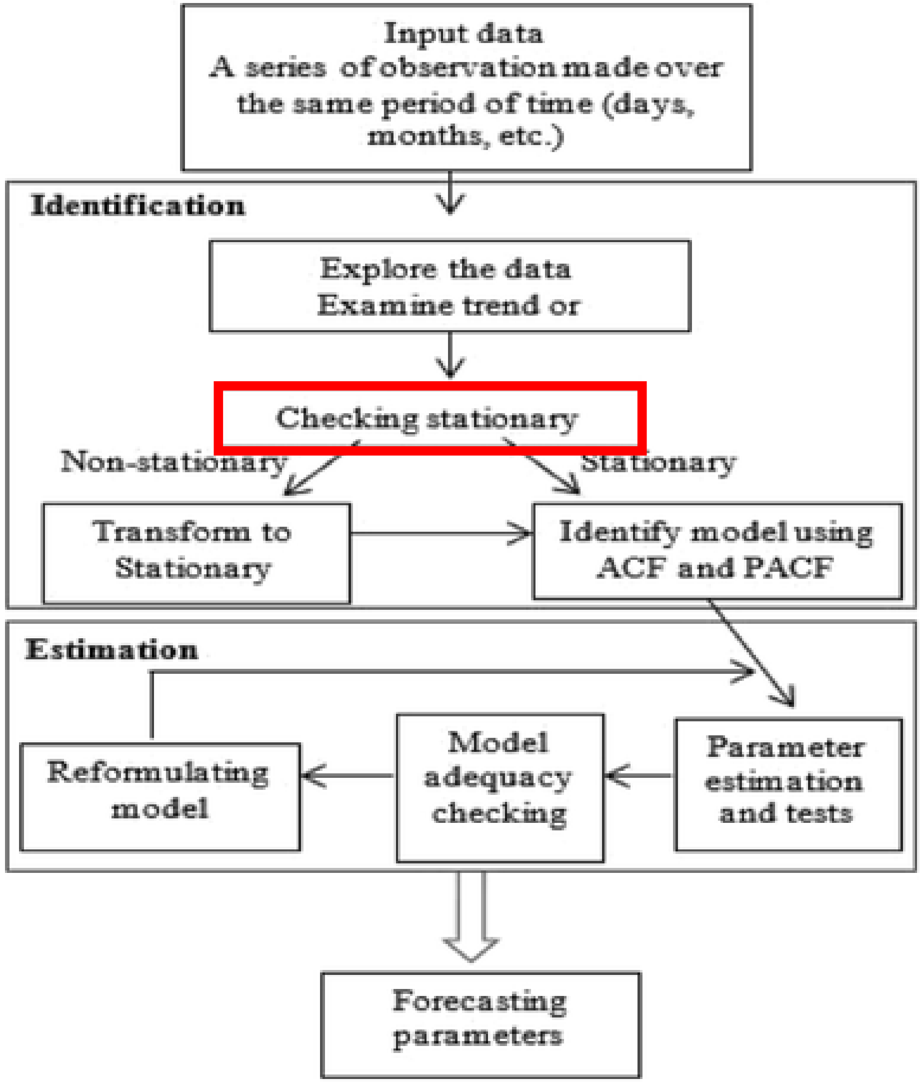


O ruído branco é um sinal discreto cujas amostras são vistas como uma seqüência de variáveis
aleatórias não autocorrelacio nadas
com média zero
e variância finita

Passo 3

Verificar Estacionariedade

Identificação Modelo de Box-Jenkins

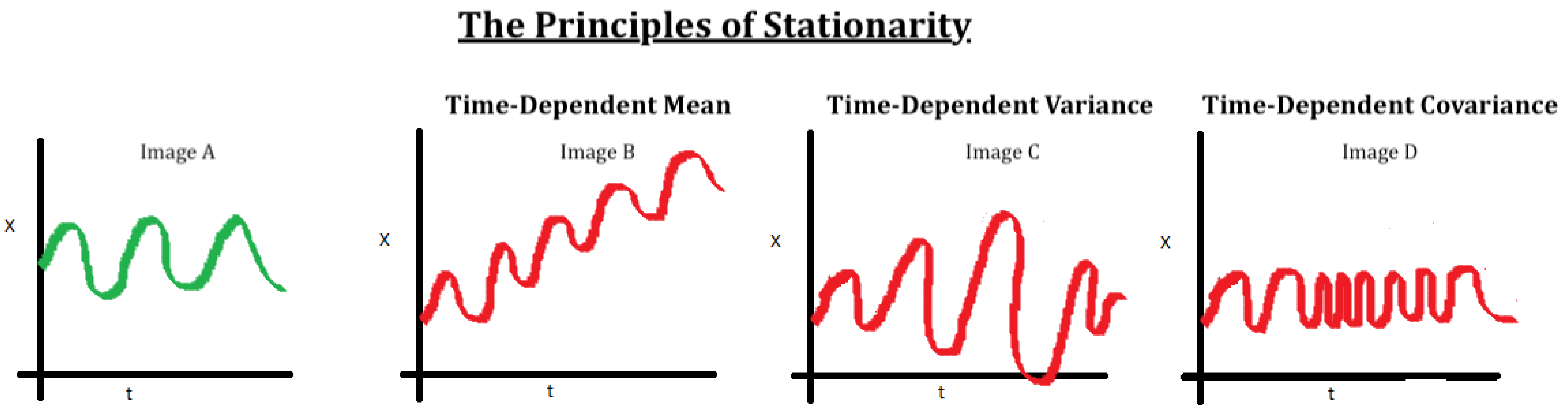


https://link.springer.com/article/10.1007/s40808-016-0164-0

Passo 1

- Verificar se os dados são estacionários.

Estacionariedade



Stationary series

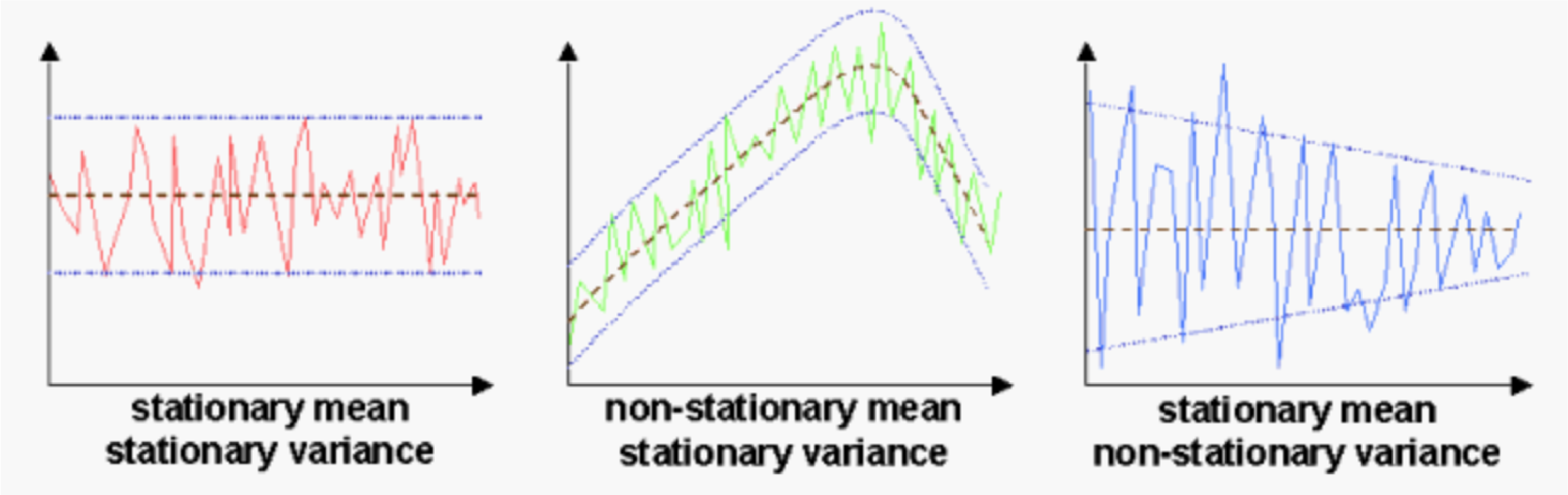
Non-Stationary series

Non-Stationary series

Non-Stationary series

Uma suposição comum em muitas técnicas de séries temporais é que os dados sejam estacionários.
Um processo estacionário tem a propriedade de que a média, variância e estrutura de autocorrelação não mudam no decorrer do tempo

Estacionariedade



https://towardsdatascience.com/stationarity-in-time-series-analysis-90c94f27322

How to make TS stationary?

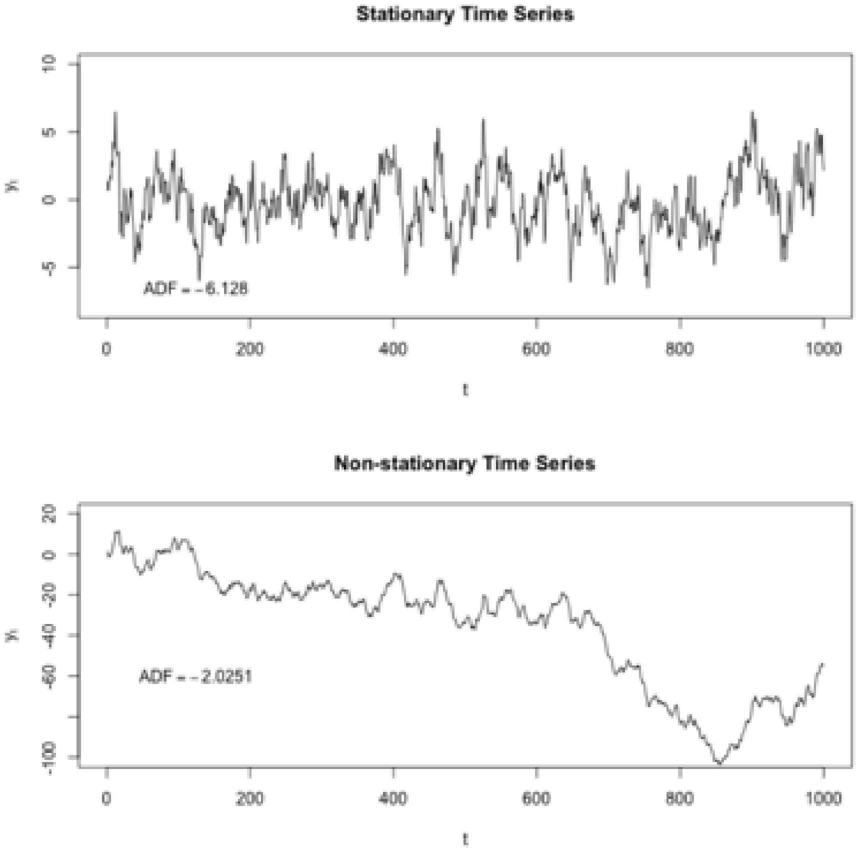
•Constant mean – Average
•Constant variance – Distance from mean
•Auto covariance that does not depend on time

https://beingdatum.com/time-series-forecasting/

Estacionariedade – Como Testar

HOW TO TEST STATIONARITY?

- Rolling Statistics: **Visual technique**
- **Augmented Dickey Fuller Test**:
  Here the null hypothesis is that the TS is non-stationary.
  If ‘Test Statistics’ < ‘Critical Value’, then we can reject the hypothesis & conclude that TS is stationary.
- If the test statistic is less than the critical value, we can reject the null hypothesis (so the series is stationary).
  Therefore, n=1 in Dickey Fuler Test means p >0.05 and we cannot reject the null hypothesis.



Estacionariedade – Referência

DigitalHouse>
Coding School

Stationarity in time series analysis

A review of the concept and types of stationarity

Shay Palachy Follow
Apr 8, 2019 · 17 min read ★

This post is meant to provide a concise but comprehensive overview of the concept of stationarity and of the different types of stationarity defined in academic literature dealing with time series analysis.

Future posts will aim to provide similarly concise overviews of detection of non-stationarity in time series data and of the different ways to transform non-stationary time series into stationary ones.[1]

. . .

Why is stationarity important?

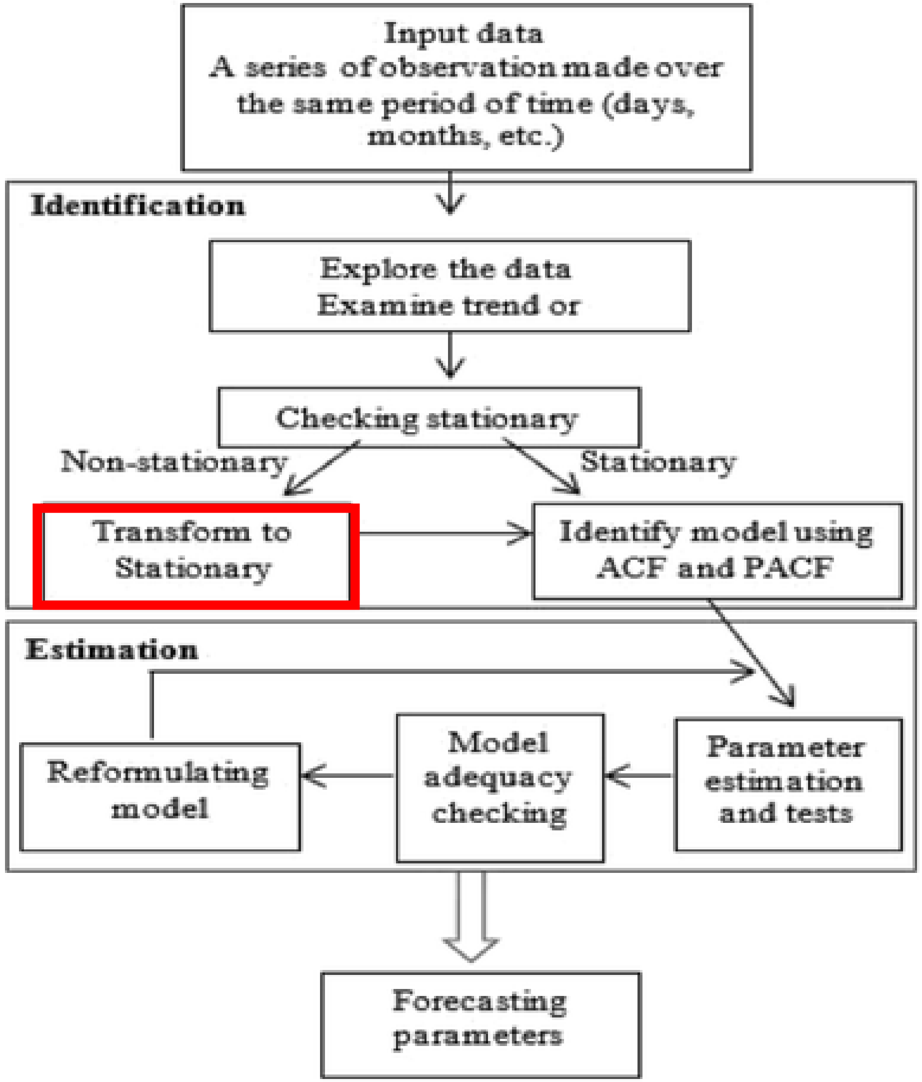
Before diving into formal definitions of stationarity, and the related concepts upon which it builds, it is worth considering why the concept of stationarity has become important in time series analysis and its various applications.

https://towardsdatascience.com/stationarity-in-time-series-analysis-90c94f27322

Passo 4

Transformar para Estacionário (se necessário)

Identificação Modelo de Box-Jenkins

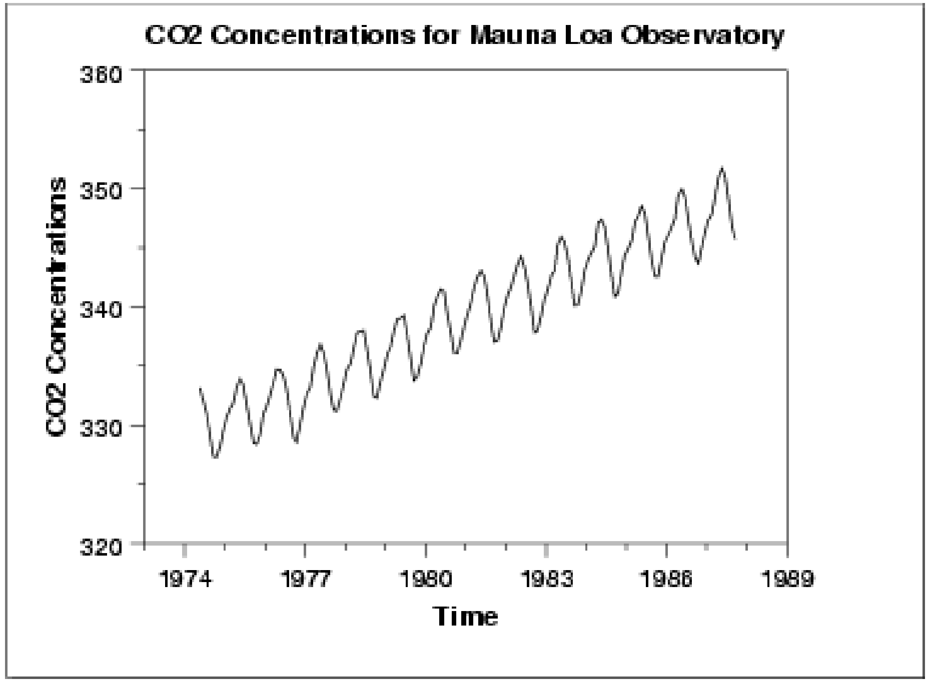


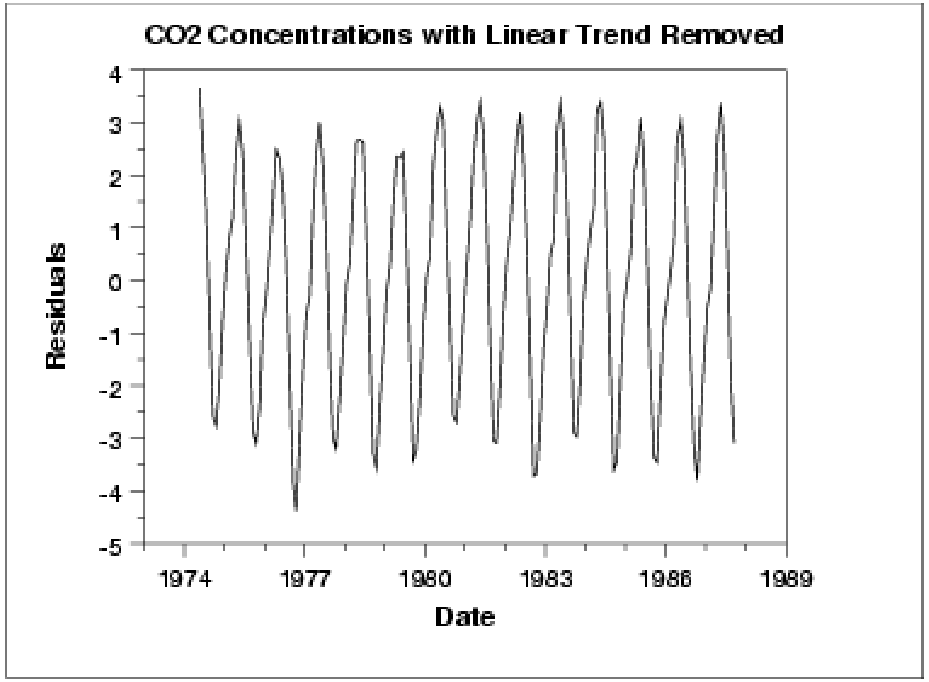
https://link.springer.com/article/10.1007/s40808-016-0164-0

Passo 4

- Transformar os dados em estacionários.

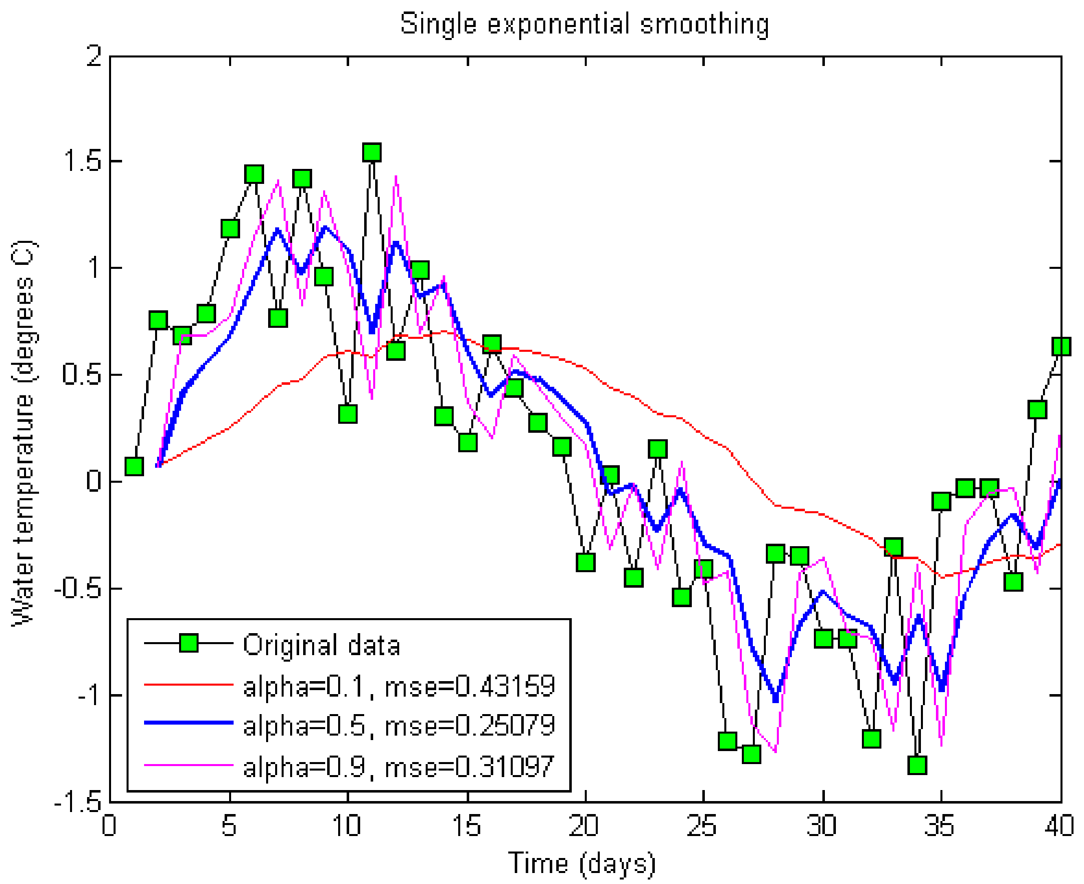
Differencing – Tornar Estacionário





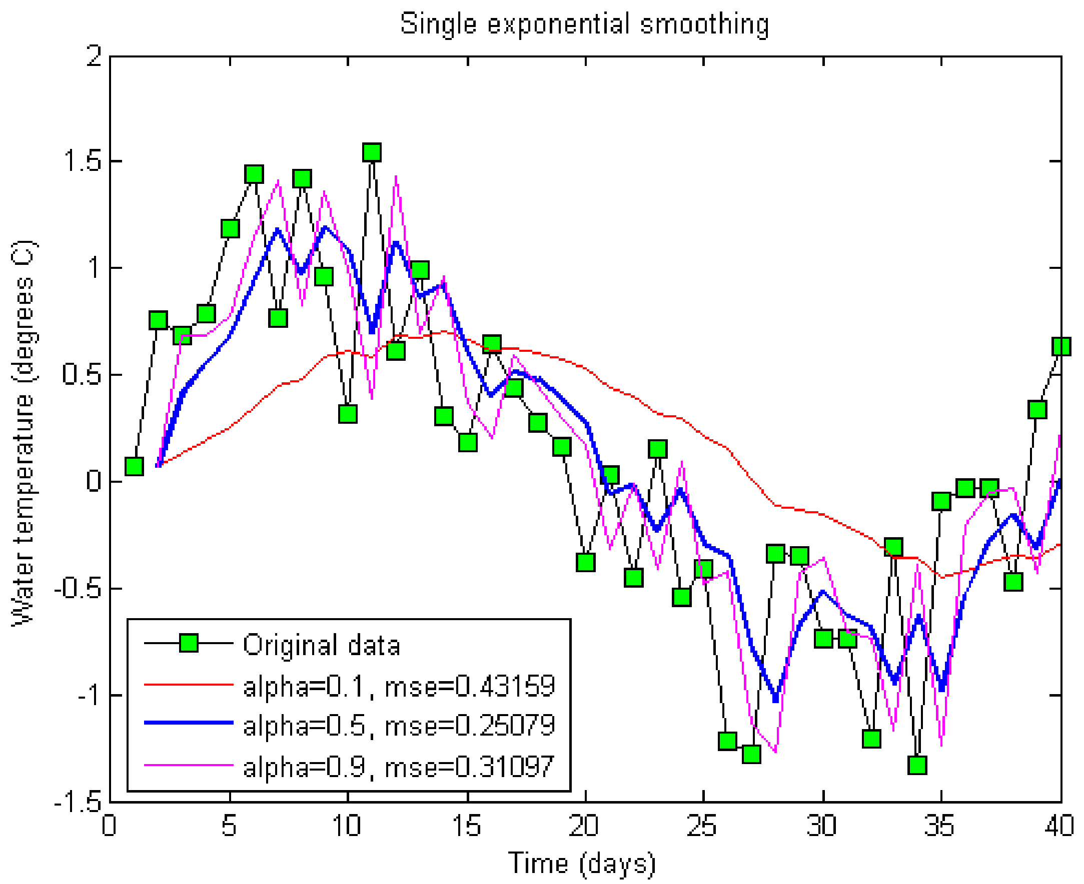
Uma das técnicas para tornar uma série estacionária é fazer o *differencing* – que consiste em subtrair valores entre tempos adjacentes da série. Desta maneira o gráfico irá representar a variação daquilo que estamos medindo.

Suavização - *Smoothing*



Inerente na coleção de dados tomados durante o tempo é alguma forma de variação aleatória. Existem métodos para redução do cancelamaento do efeito devido à variação aleatória. Uma técnica frequentemente usada na indústria é a "suavização".

Suavização – Média Móvel

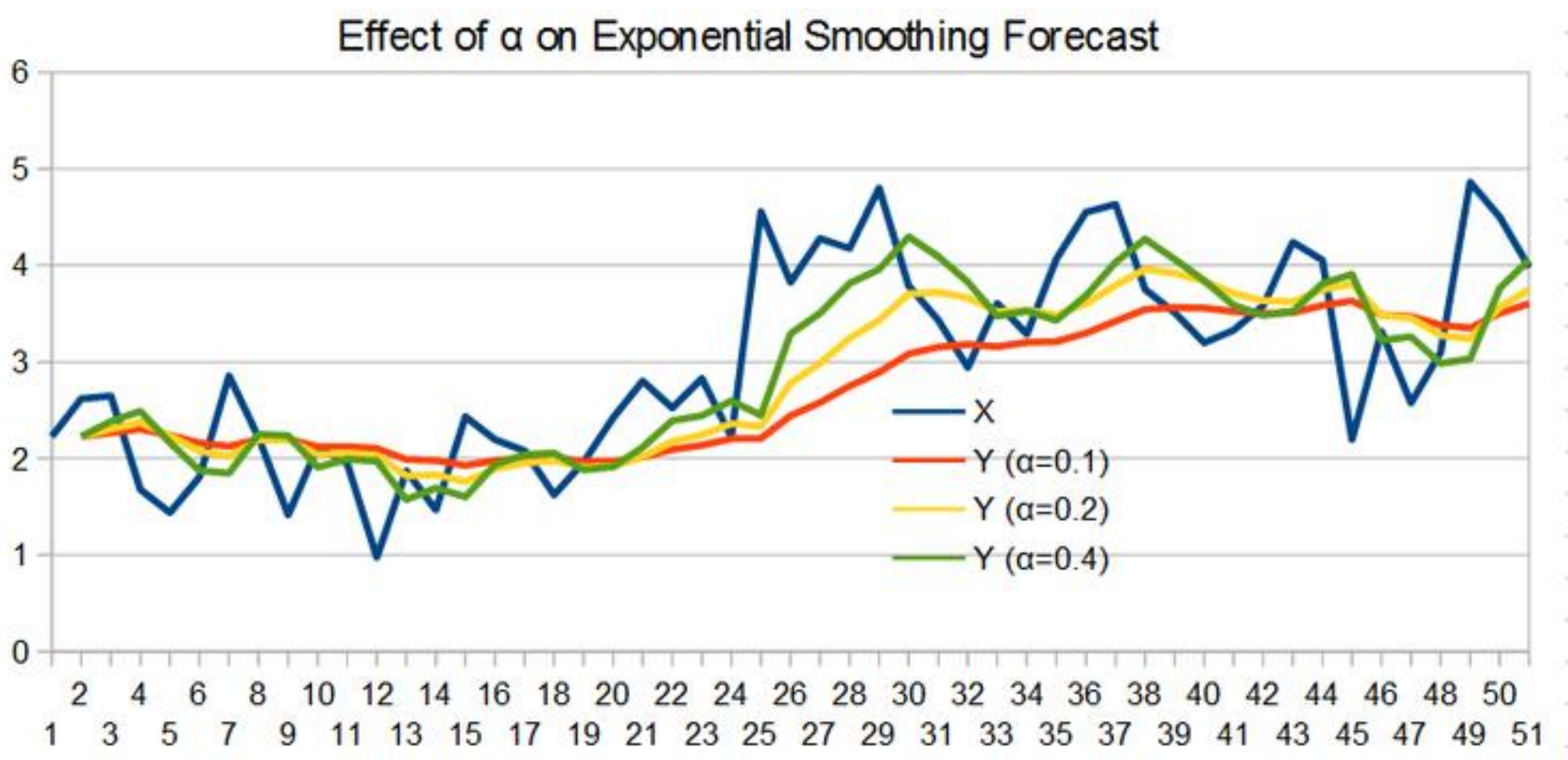


Inerente na coleção de dados tomados durante o tempo é alguma forma de variação aleatória. Existem métodos para redução do cancelamaento do efeito devido à variação aleatória. Uma técnica frequentemente usada na indústria é a "suavização".

Suavização – Média Móvel

Quanto maior o intervalo de dias selecionado, mais o valor se aproximará da média dos dados com um todo.

Suavização – Exponencial



Enquanto na Média Móvel Simples as observações passadas são ponderadas igualmente, a Suavização Exponencial atribui pesos decrescentes exponencialmente quando a observação fica mais velha.

DigitalHouse>
Coding School

Suavização – Exponencial Tripla

6.4.3.5. Suavização Exponencial Tripla

O que acontece se os dados apresentarem tendência **e** sazonalidade?

Para manusear sazonalidade, teremos de adicionar um terceiro parâmetro

Neste caso a suavização dupla não funcionará. Introduzimos agora uma terceira equação que leva em conta a sazonalidade (algumas vezes chamada periodicidade). O conjunto resultante de equações é chamado de método de "Holt-Winters" (HW) que são os nomes de seus inventores.

As equações básicas para o seu método são dadas por:

$$S_t = \alpha \frac{y_t}{I_{t-L}} + (1-\alpha)(S_{t-1} + b_{t-1})$$ OVERALL SMOOTHING

$$b_t = \gamma(S_t - S_{t-1}) + (1-\gamma)b_{t-1}$$ TREND SMOOTHING

$$I_t = \beta \frac{y_t}{S_t} + (1-\beta) I_{t-L}$$ SEASONAL SMOOTHING

$$F_{t+m} = (S_t + m b_t) I_{t-L+m}$$ FORECAST

onde

- y é a observação
- S é a observação suavizzada
- b é o fator de tendência
- I é o índice sazonal
- F é a previsão em m períodos adiante
- t é um índice denotando um período de tempo

e α, β, e γ são constantes que devem ser estimadas de tal modo que o MSE dos erros seja minimizado. Isto fica melhor se deixado para um bom pacote de software.

Estação completya necessária

Para inicializar o método HW precisamos no mínimo dados de uma estação completa para determinar estimativas inciais dos índices de sazonalidade I_{t-L}.

http://www.bertolo.pro.br/MetodosQuantitativos/Previsao/pmc435.htm

Recomendada quando os dados apresentarem tendência e sazonalidade.

Suavização – Exponencial Dupla

6.4.3.3. Suavização Exponencial Dupla

A suavização exponencial dupla usa duas constantes e é melhor para se manipular tendências

Como foi observado anteriormente, a Suavização Simples não é excelente para seguir os dados quando houver tendência. Esta situação pode ser melhorada pela introdução de uma segunda equação com uma segunda constante, γ, que deve ser escolhida conjuntamente com α.

Aqui estão as duas equações associadas com a Suavização Exponencial Dupla:

$$S_t = \alpha y_t + (1-\alpha)\,(S_{t-1} + b_{t-1}) \qquad 0 \le \alpha \le 1$$

$$b_t = \gamma\,(S_t - S_{t-1}) + (1-\gamma)\, b_{t-1} \qquad 0 \le \gamma \le 1$$

Note que o valor atual da série foi usado para calcular seu valor suavizado trocado na suavização exponencial.

Valores Iniciais

Vários métodos para se escolher os valores iniciais

Como no caso da suavização simples, existem uma variedade de esquemas para definir os valores iniciais para S_t e b_t na suavização dupla.

S_1 é em geral definido como y_1. Aqui estão três sugestões para b_1:

$$b_1 = y_2 - y_1$$

$$b_1 = [(y_2 - y_1) + (y_3 - y_2) + (y_4 - y_3)]/3$$

$$b_1 = (y_n - y_1)/(n - 1)$$

http://www.bertolo.pro.br/MetodosQuantitativos/Previsao/pmc435.htm

Recomendada quando os dados não apresentarem tendência.

Suavização – Exponencial Simples

DigitalHouse>
Coding School

6.4.3.1. Suavização exponencial Simples

A suavização exponencial simples pondera as observações passadas com pesos decrescentes exponencialmente para previsão de valores futuros

Este esquema de suavização começa definido S_2 para y_1, onde S_i representa a observação suavizada ou EWMA, e y representa a observação original. Os subscritos se referem aos períodos de tempo, 1, 2, ..., n. Para o terceiro período, $S_3 = \alpha\, y_2 + (1-\alpha)\, S_2$; e assim por diante. Não há S_1; a série suavizada inicia com a versão suavizada da segunda observação.

Para qualquer período de tempo t, o valor suavizado S_t é encontrado calculando

$$S_t = \alpha y_{t-1} + (1-\alpha) S_{t-1} \qquad 0 < \alpha \leq 1 \qquad t \geq 3$$

Esta é a *equação básica da suavização exponencial* e a constante ou parâmetro α é chamado de *constante de suavização*.

Nota: Existe uma abordagem alternativa para suavização exponencial que troca y_{t-1} na equação básica com y_t, a observação atual. Esta formulação, devida a Roberts (1959), é descrita na seção sobre Gráficos de controle EWMA. A formulação aqui segue Hunter (1986).

Definindo o primeiro EWMA

A primeira previsão é muito importante

O EWMA inicial faz um importante papel no cálculo de todos os EWMA's subsequentes. Definindo S_2 como y_1 é um método de inicialização. Outra maneira é defini-lo como alvo do processo.

Ainda outra possibilidade seria fazer a média das quatro ou cinco primeiras observações.

http://www.bertolo.pro.br/MetodosQuantitativos/Previsao/pmc435.htm

Recomendada quando os dados não apresentarem tendência nem sazonalidade.

Passo 5

Identificar modelo usando ACF e PACF

Baseline – Predições *Naive*

É comum inicialmente fazermos uma estimativa muito simples - *Naive* (ingênua) para servir de baseline para os demais modelos que iremos testar. Exemplos:

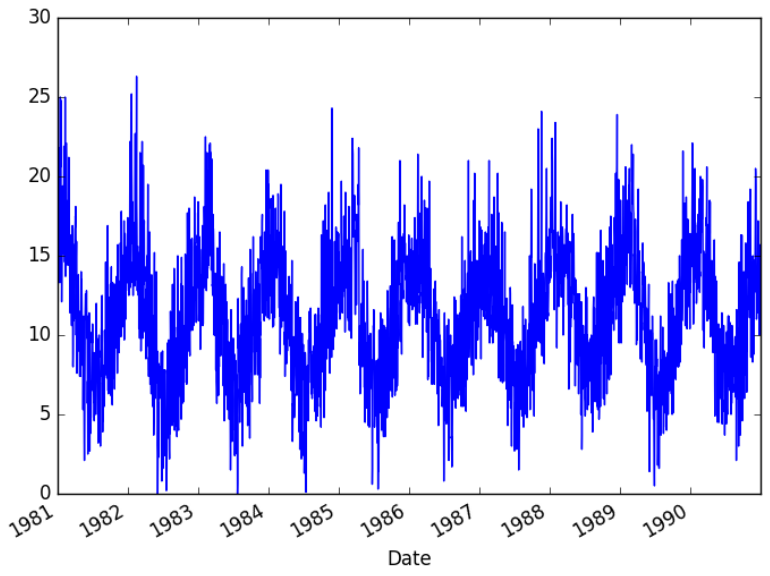
- Repetir o último valor
- Utilizar o valor médio
- Utilizar uma média rolante
- Valor médio
- etc

ACF - Auto Correlation Function

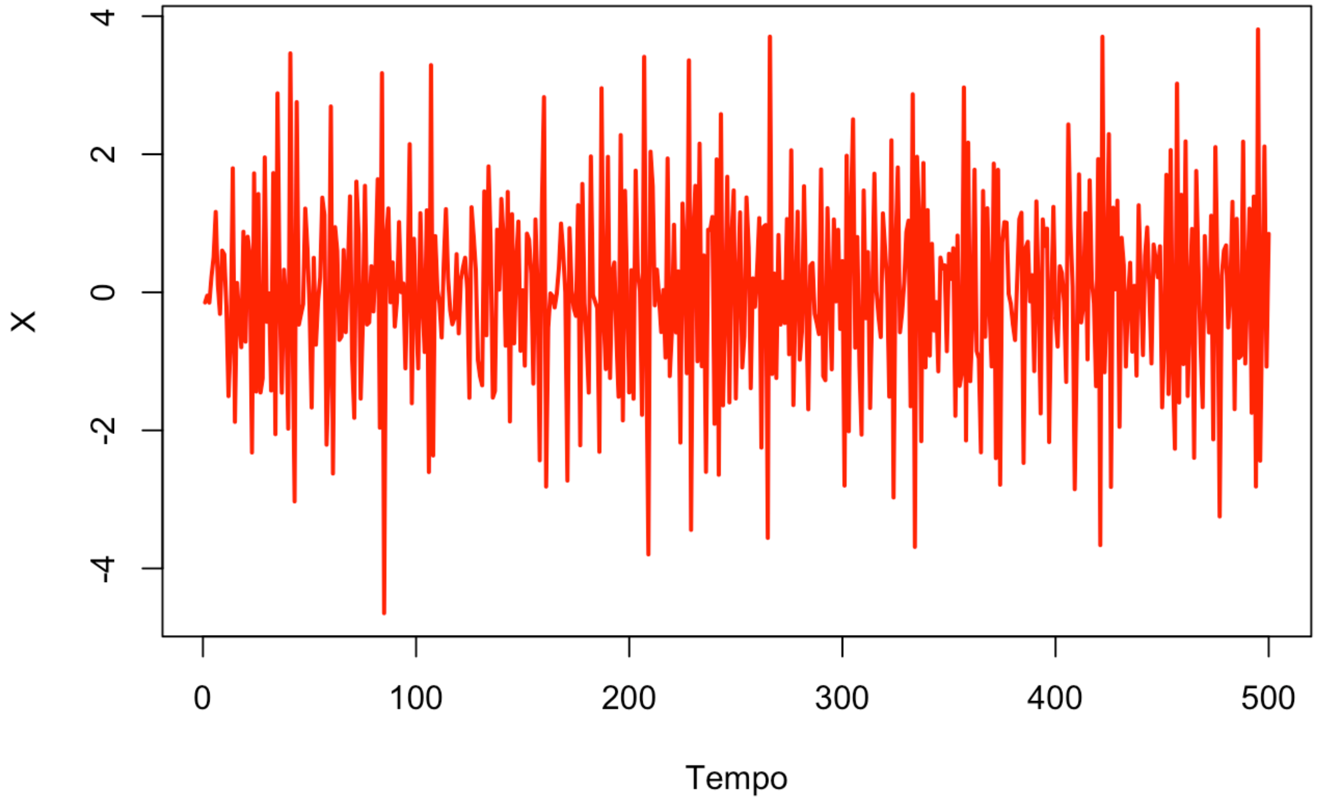
O coeficiente de correlação entre dois valores em uma Time Series é chamada de **Auto Correlation function** (**ACF**). Por exemplo, o ACF de uma série temporal é dado por:

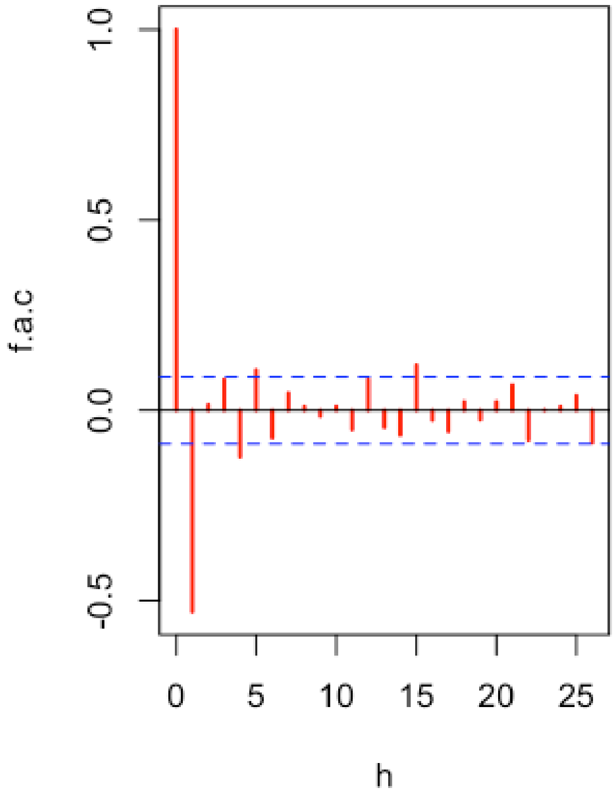
$$\mathbf{ACF = Corr(yt,yt{-}k)}$$

Onde: k é intervalo de tempo entre os dois valores que queremos medir.



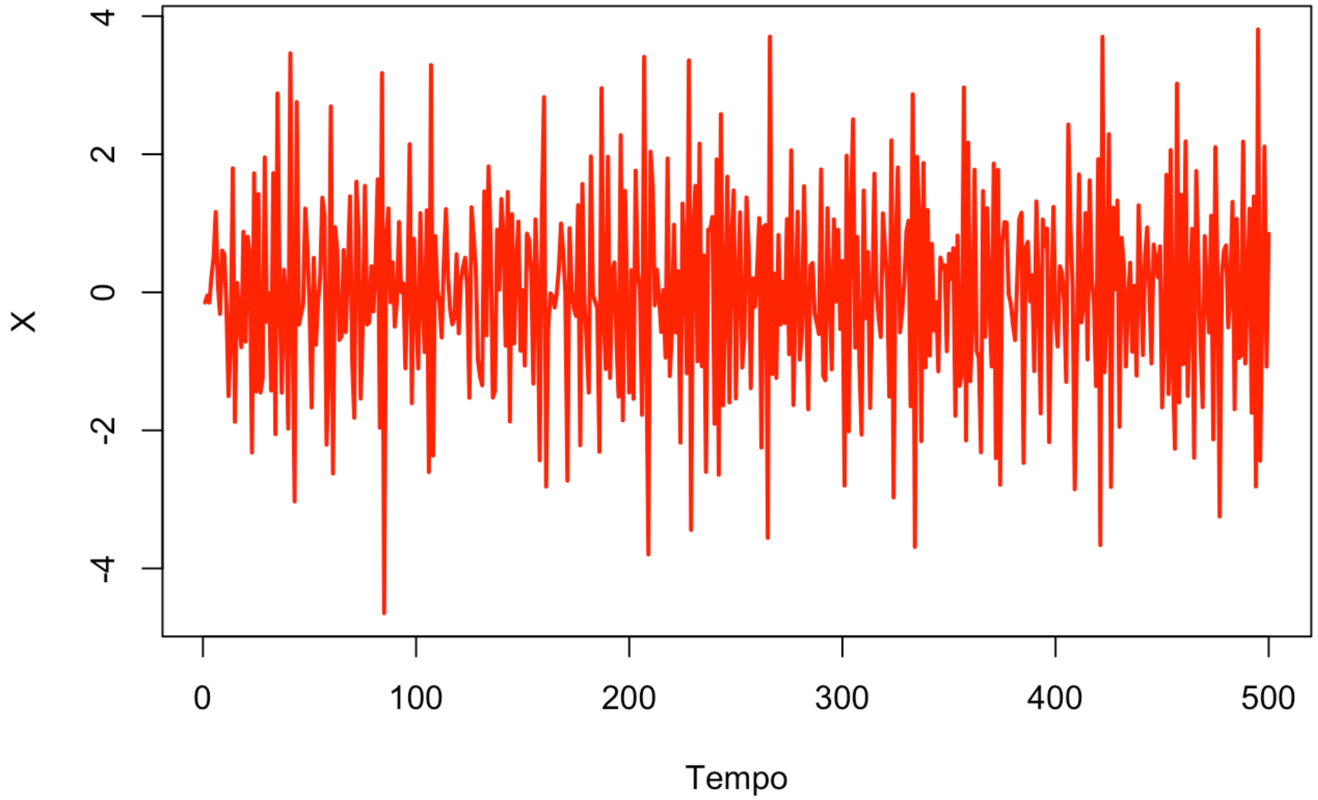
ACF - Auto Correlation Function

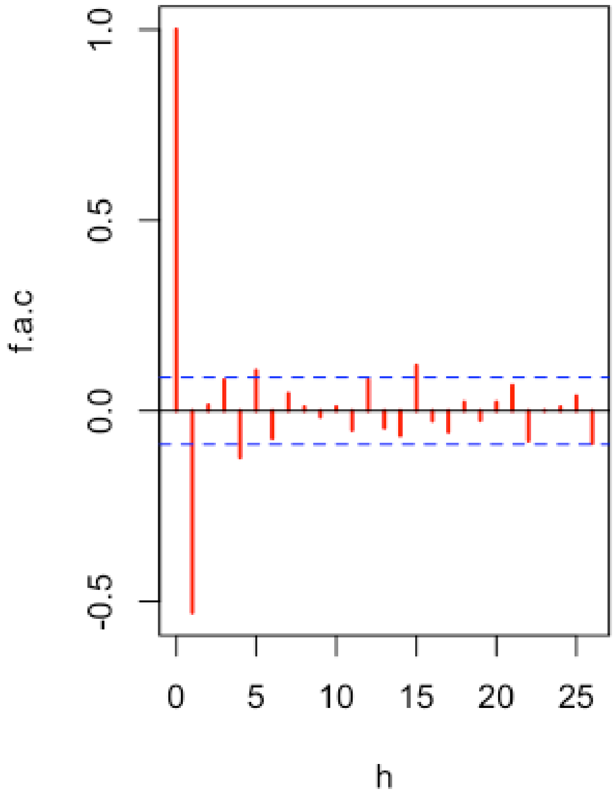




O resultado do ACF é a correlação (positiva ou negativa) deste intervalo K – considerando toda a série.

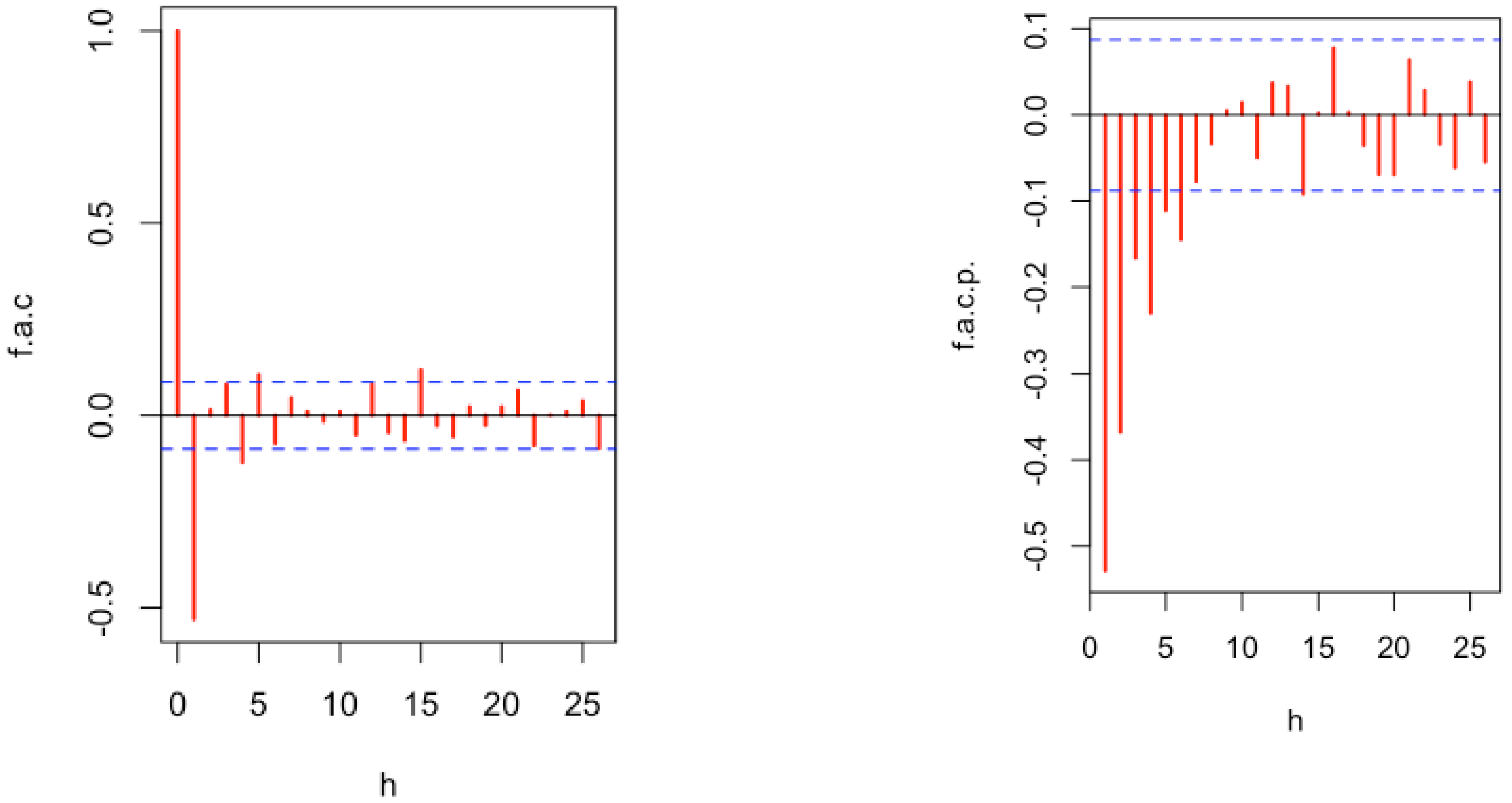
PACF – Partial Auto Correlation Function





O resultado do ACF é a correlação (positiva ou negativa) deste intervalo K – considerando toda a série. São levados em consideração os pontos intermediários entre K.

PACF – Partial Auto Correlation Function



Também há o PACF – que mede a correlação de maneira análoga, mas que **não leva em consideração os pontos intermediários.**

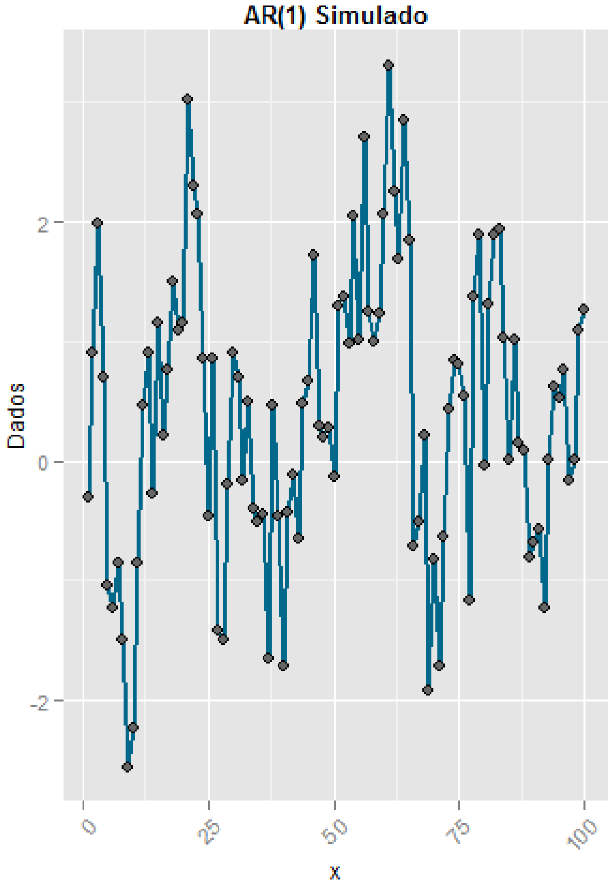
Passo 6

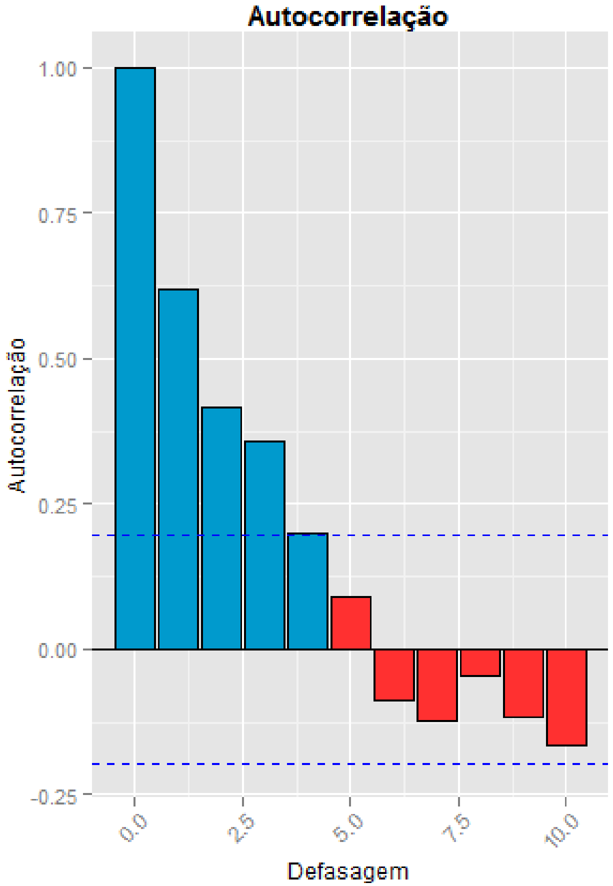
Modelos e parâmetros

Modelo MA – *Moving Average*

É o mais “naive” dos modelos, simplesmente aplica a média aos dados do passado considerando-se uma janela de tempo **q**.

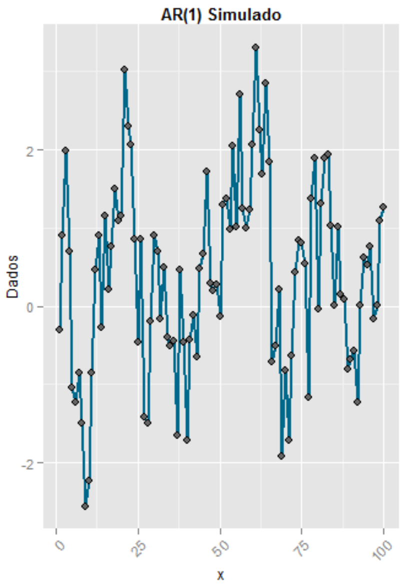
Modelo AR – *Auto Regressive*

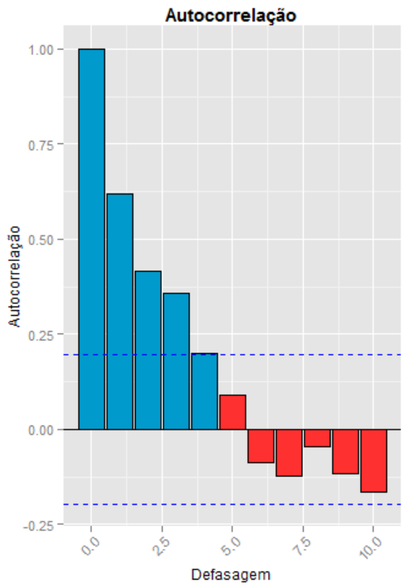




A principal ideia dos modelos auto regressivos (AR)
é que o resultado de cada ponto no futuro é função de uma regressão dos valores no passado. $y_t=\beta_0+\beta_1 y_{t-1}+\epsilon_t$. Tem como parâmetro a quantidade de pontos **p** que serão usados na Regressão.

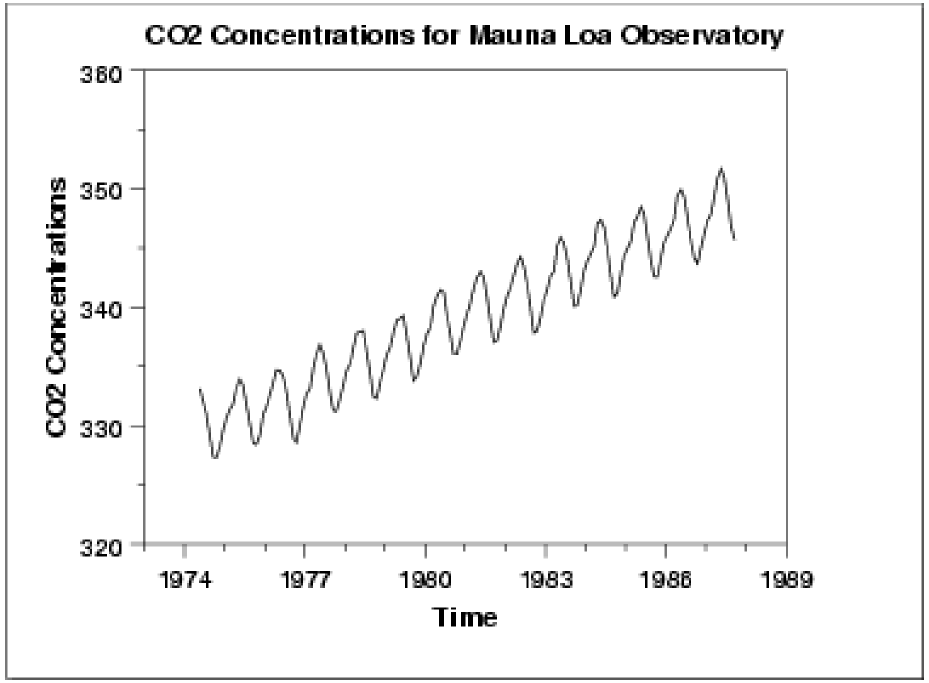
Modelo ARMA (AR + MA)

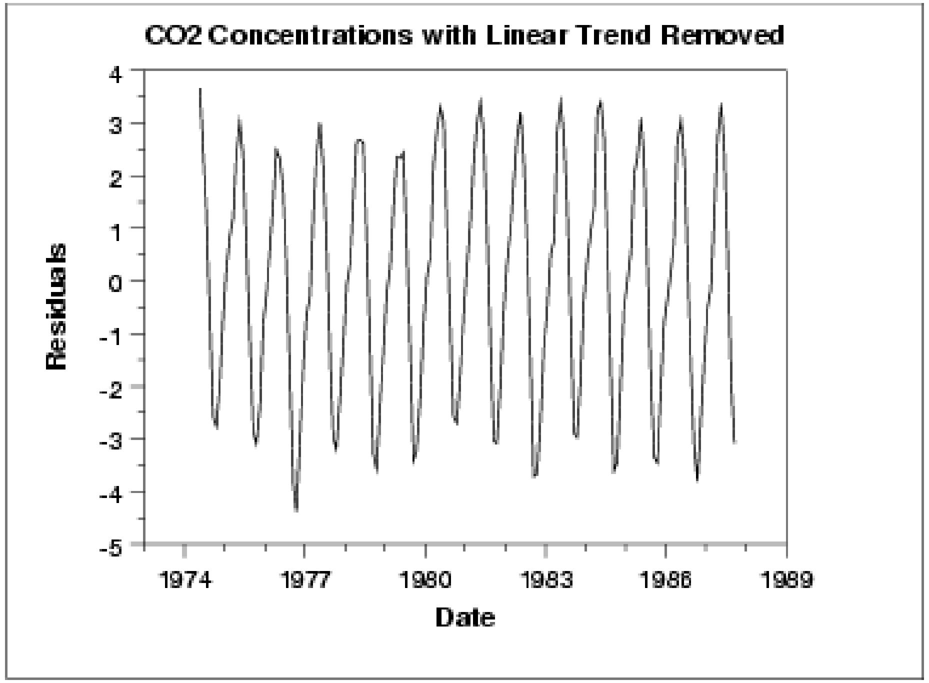




Combina os modelos AR e MA e se utiliza dos mesmos parâmetros **q** e **p.**

Modelo ARIMA (AR+ Integrated + MA)





O ARIMA consegue tratar séries não estacionárias, com seu fator **d** – de diferenciação.

Modelo ARIMA (AR+ Integrated + MA)

Terminologias em ARIMA

O modelo ARIMA pode ser basicamente resumido por três fatores:

p = o número de termos autorregressivos

- **p** é o número de termos auto-regressivos (parte AR). Permite incorporar o efeito de valores passados em nosso modelo. Intuitivamente, isso seria semelhante ao afirmar que é provável que esteja quente amanhã se estiver quente nos últimos três dias.

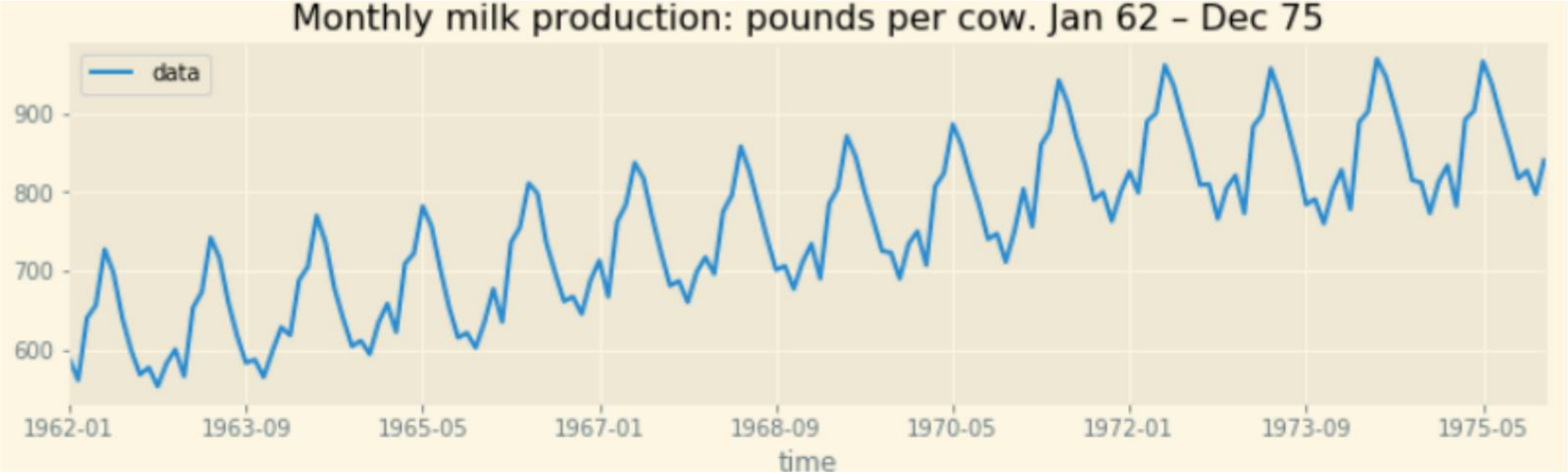
d = o número é o fator que vamos aplicar para resolver a estacionaridade

- **d** é o número de diferenças não sazonais necessárias para a estacionariedade. Intuitivamente, isso seria semelhante ao afirmar que provavelmente haverá a mesma temperatura amanhã se a diferença de temperatura nos últimos três dias tiver sido muito pequena.

q = o número de termos da média móvel

- **q** é o número de erros de previsão atrasados na equação de previsão (parte MA). Isso nos permite definir o erro do nosso modelo como uma combinação linear dos valores de erro observados em momentos anteriores no passado.
- Estes são os três números inteiros (p, d, q) usados para parametrizar os modelos ARIMA. Portanto, isso é chamado de modelo "ARIMA (p, d, q)".

Modelo SARIMA – (Seasonal + ARIMA)



O SARIMA consegue tratar séries sazonais e não estacionárias e possui o parâmetro **s** que indica em quanto tempo a série se repete, a sazonalidade se manifesta.

Métricas de Qualidade

- R squared: coeficiente de determinação (em econometria, isso pode ser interpretado como o percentual de variação explicado pelo modelo, $(-\infty, 1]$

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

```
sklearn.metrics.r2_score
```

- Mean Absolute Error: essa é uma métrica interpretável porque tem a mesma unidade de medida que a série inicial, $[0, +\infty)$

$$MAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n}$$

```
sklearn.metrics.mean_absolute_error
```

- Median Absolute Error: novamente, uma métrica interpretável que é particularmente interessante porque é robusta para valores extremos, $[0, +\infty)$

$$MedAE = median(|y_1 - \hat{y}_1|, \ldots, |y_n - \hat{y}_n|)$$

```
sklearn.metrics.median_absolute_error
```

- Mean Squared Error: a métrica mais usada que concede uma penalidade mais alta a erros grandes e vice-versa, $[0, +\infty)$

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

```
sklearn.metrics.mean_squared_error
```

Métricas de Qualidade

DigitalHouse>
Coding School

- Mean Squared Logarithmic Error: praticamente, é o mesmo que MSE, mas adotamos o logaritmo da série. Como resultado, também damos mais importância a pequenos erros. Isso geralmente é usado quando os dados têm tendências exponenciais, $[0, +\infty)$

$$MSLE = \frac{1}{n}\sum_{i=1}^{n}(log(1+y_i) - log(1+\hat{y}_i))^2$$

```
sklearn.metrics.mean_squared_log_error
```

- Mean Absolute Percentage Error: é o mesmo que o MAE, mas é calculado como uma porcentagem, o que é muito conveniente quando você deseja explicar a qualidade do modelo ao gerenciamento,, $[0, +\infty)$

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{y_i}$$

```
def mean_absolute_percentage_error(y_true, y_pred):
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100
```

Modelos Multivariate

Modelo ARIMAX

https://www.youtube.com/watch?v=GrIZ5ArKDnA

O ARIMAX funciona como o ARIMA, porém ele possui a capacidade de tratar e considerar em suas predições variáveis eXplanatórias.

Modelo SARIMAX

https://www.youtube.com/watch?v=FgGzZOGg-8c

De maneira análoga o SARIMAX funciona como o SARIMA e possui a capacidade de tratar e considerar em suas predições variáveis eXplanatórias.

Outros Modelos e Ferramentas utilizadas

em Time Series

Facebook PROPHET

PROPHET

Facebook Prophet's logo

Implementing Facebook Prophet efficiently

Ruan van der Merwe Follow
Nov 14, 2018 · 12 min read

If you have ever worked with time series predictions, I am quite sure you are well aware of the strains and pains that come with them. One moment you think you have cracked the stock market, the next moment you are lying in the bath crying and cursing your inaccurate models (I really don't recommend that you try and predict the stock market, you will most likely not reap the benefits you think you will). What I am trying to say is that time series predictions are difficult and always require a very specialized data scientist to implement it.

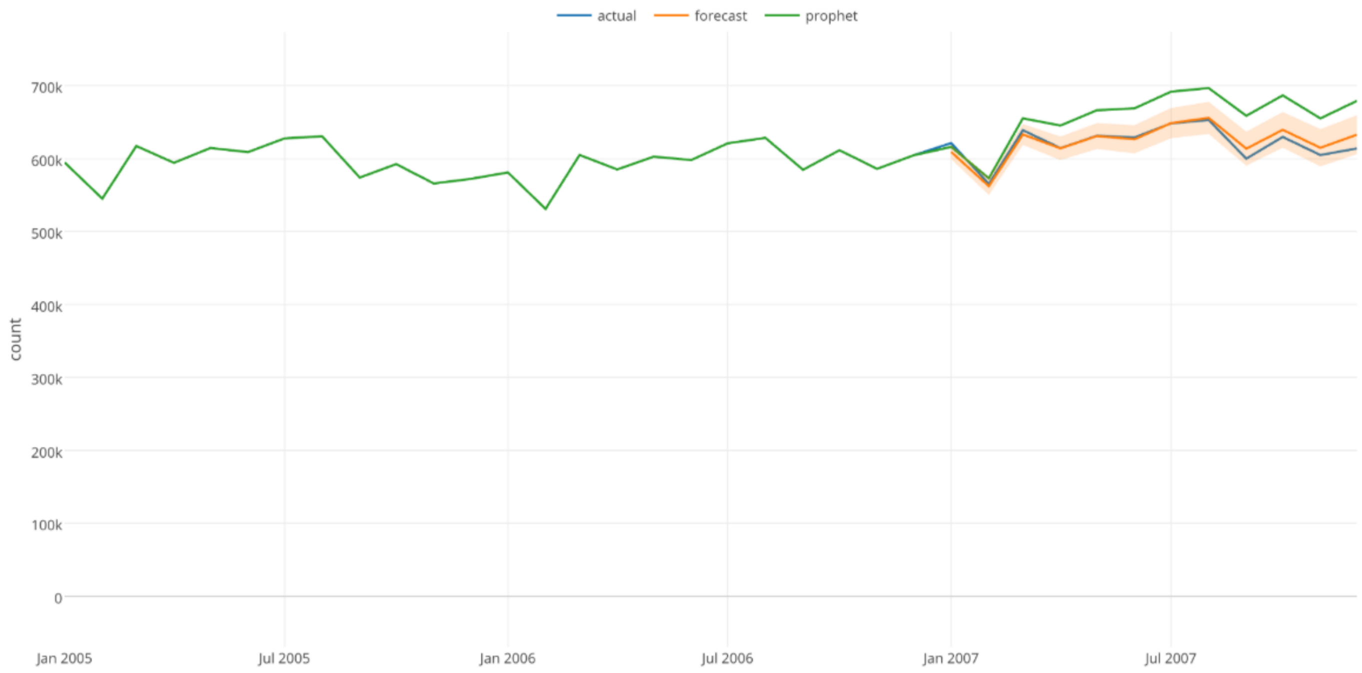
https://towardsdatascience.com/implementing-facebook-prophet-efficiently-c241305405a3

PROPHET x ARIMA

DigitalHouse>
Coding School

Is Prophet Really Better than ARIMA for Forecasting Time Series Data?

Hideaki Hayashi Follow
Oct 17, 2017 · 11 min read



https://blog.exploratory.io/is-prophet-better-than-arima-for-forecasting-time-series-fa9ae08a5851

VAR - Vector Autoregression

Vector Autoregression (VAR) – Comprehensive Guide with Examples in Python

by Selva Prabhakaran |

Vector Autoregression (VAR) is a forecasting algorithm that can be used when two or more time series influence each other. That is, the relationship between the time series involved is bi-directional. In this post, we will see the concepts, intuition behind VAR models and see a comprehensive and correct method to train and forecast VAR models in python using `statsmodels`.

https://www.machinelearningplus.com/time-series/vector-autoregression-examples-python/

O VAR implementa um modelo vetorial que funciona bem quando temos mais de uma time series interagindo entre si.

Reinforcement Learning

Open access peer-reviewed chapter

Training Deep Neural Networks with Reinforcement Learning for Time Series Forecasting

By Takashi Kuremoto, Takaomi Hirata, Masanao Obayashi, Shingo Mabu and Kunikazu Kobayashi

Submitted: June 14th 2018 Reviewed: February 26th 2019 Published: April 3rd 2019
DOI: 10.5772/intechopen.85457

https://www.intechopen.com/books/time-series-analysis-data-methods-and-applications/training-deep-neural-networks-with-reinforcement-learning-for-time-series-forecasting

ARCH e GARCH Modeling

DigitalHouse>
Coding School

ARCH vs. GARCH Models

The ARCH(q) Model

The ARCH(1) model can be extended to such that the conditional variance depends on more than one lagged realisation. For instance:

ARCH(*1*): $h_t = b_0 + b_1 \boldsymbol{u_{t-1}^2}$

ARCH(q): $h_t = b_0 + b_1 \boldsymbol{u_{t-1}^2} + b_2 \boldsymbol{u_{t-2}^2} + \cdots + b_q \boldsymbol{u_{t-q}^2}$

ARCH(q): $h_t = b_0 + \sum_{i=1}^{q} b_i \boldsymbol{u_{t-i}^2}$

The ARCH(q) model shows that the variance or volatility in a given period depends on the magnitudes of the squared errors in the past q periods. ***simultaneously*** examines the mean and variance of a variable.

Remember: $b_0 > 0$, $0 \leq b_1 < 1$ but the ARCH(q) model frequently yields negative coefficients of the lagged periods of the squared error.

Subscribe; Comment; Like; Share

CrunchEconometrix YouTube Channel

The GARCH(p, q) Model

Modify the ARCH(q) model by imposing a geometric lag structure of this form: $b_s = b_1 \theta_1^{s-1}$,

$$h_t = b_0 + b_1 u_{t-1}^2 + \theta_1 b_1 u_{t-2}^2 + \theta_1^2 b_1 u_{t-3}^2 + \cdots$$

Next add and subtract $\theta_1 b_0$ and rearrange the terms:

$$h_t = (b_0 - \theta_1 b_0) + \theta_1 (b_0 + b_1 u_{t-2}^2 + \theta_1^2 b_1 u_{t-3}^2 + \cdots) + b_1 u_{t-1}^2$$

Since $h_{t-1} = b_0 + \theta_1 b_1 u_{t-3}^2 + \theta_1^2 b_1 u_{t-4}^2 + \cdots + b_1 u_{t-2}^2$, this simplifies to:

$$\boldsymbol{h_t = \varphi + \theta_1 h_{t-1} + b_1 u_{t-1}^2}$$

Where, $\varphi = (b_0 - \theta_1 b_0)$

This is a GARCH(*1*, *1*) model: contains 1 lagged term of the conditional variance (h) and 1 lagged term of the squared error (u^2).

Os modelos ARCH e GARCH tem mostrado resultados expressivos em cenários que temos muita volatilidade.

https://www.youtube.com/watch?v=B-fYft6b4nE

Long Short Term Memory

DigitalHouse>
Coding School

Multivariate Time Series Forecasting with LSTMs in Keras

by **Jason Brownlee** on August 14, 2017 in **Deep Learning for Time Series**

Tweet Share Share

Last Updated on August 5, 2019

Neural networks like Long Short-Term Memory (LSTM) recurrent neural networks are able to almost seamlessly model problems with multiple input variables.

This is a great benefit in time series forecasting, where classical linear methods can be difficult to adapt to multivariate or multiple input forecasting problems.

In this tutorial, you will discover how you can develop an LSTM model for multivariate time series forecasting in the Keras deep learning library.

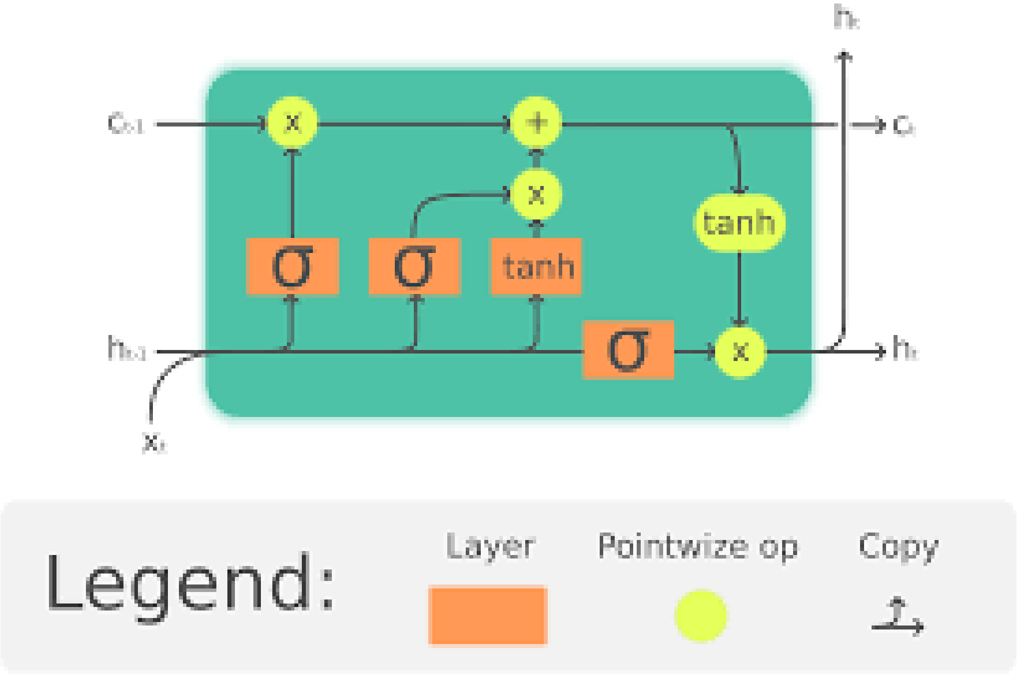
After completing this tutorial, you will know:

- How to transform a raw dataset into something we can use for time series forecasting.
- How to prepare data and fit an LSTM for a multivariate time series forecasting problem.
- How to make a forecast and rescale the result back into the original units.

Discover how to build models for multivariate and multi-step time series forecasting with LSTMs and more in my new book, with 25 step-by-step tutorials and full source code.

Let's get started.

- **Updated Aug/2017**: Fixed a bug where yhat was compared to obs at the previous time step when calculating the final RMSE. Thanks, Songbin Xu and David Righart.
- **Update Oct/2017**: Added a new example showing how to train on multiple prior time steps due to popular demand.
- **Update Sep/2018**: Updated link to dataset.



https://machinelearningmastery.com/multivariate-time-series-forecasting-lstms-keras/

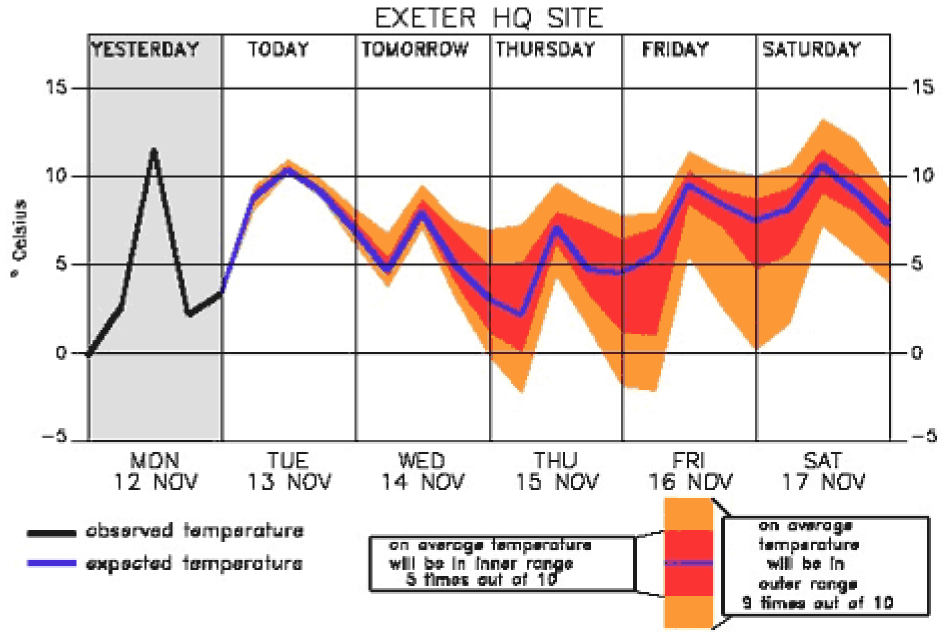
Ensembles

Using ensemble forecasts in decision-making

Ensemble prediction allows the uncertainty in forecasts to be assessed quantitatively.

Ensemble prediction allows the uncertainty in forecasts to be assessed quantitatively, and attaching numbers to the confidence or uncertainty can allow the user to assess the risks more accurately.

Ensemble forecasts contain a huge amount of information. Using the 12 forecast members shown on the right, at over a day ahead we can be confident that there will be showers and bands of heavy rain around the UK, but there is considerable uncertainty about the location and extent of the heavy rain (shown in the yellow and orange colours).



How we use the ensembles to help decision-making

Our forecasters often like to see the individual forecasts, but for other users we need to find efficient ways to summarise the information. One way is using probability forecasts.

To make best use of a probability forecasts, users must choose a probability threshold which gives the correct balance of alerts and false alarms for their particular application.

https://www.metoffice.gov.uk/research/weather/ensemble-forecasting/decision-making

Obrigado!