Análisis de Series de Tiempo

Carrera de Especialización en Inteligencia Artificial

Clase 2

Ing. Magdalena Bouza, Ing. Carlos German Carreño Romano

Agenda

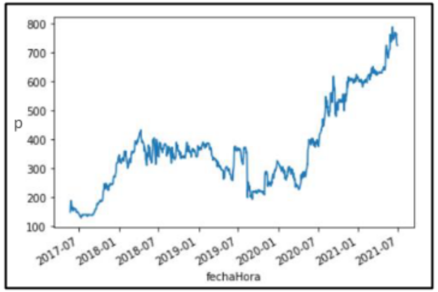
1. Revisión de actividad
2. Herramientas de preprocesamiento: transformación, diferenciación, promedio
3. Tendencia determinística
4. Explicación del Trabajo Práctico.
5. Modelos lineales, de promedio móvil y autorregresivos (AR, MA, ARMA)

Revisión actividad

Ejemplos actividad Clase 1

Ejercicio 1

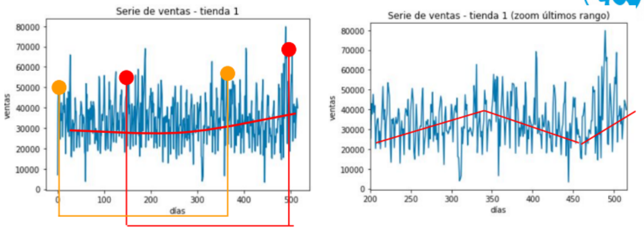
Graficar una serie a partir de un dataset relevante. Explicar observaciones



Bolsa y Mercado Argentino – Precio Cierre
23/05/2017 – 29/06/2021

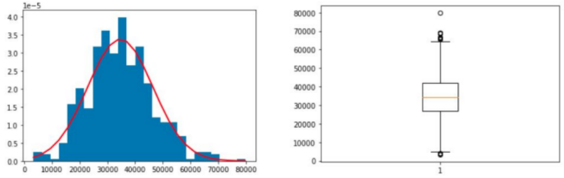
La evolución del último precio presenta cierta estabilidad hasta mediados del año 2020, entorno a un valor de 300. Luego, quizás por efecto de la pandemia, podemos ver un incremento del precio en forma constante, hasta llegar a mediados del 2021, donde toca su valor máximo histórico (en el tiempo analizado).

1 - Análisis visual



- La serie tiene un offset y parece tener una tendencia anual a incrementar las ventas.
- No parece estacionaria, y podría tener alguna estacionalidad del tipo semestral (vender más en una temporada que otra).

1 - Analizar la distribución



- La distribución tiene skew positiva, lo cual confirma que los valores incrementan con el tiempo.
- Dicho lo anterior, la serie no es estacionaria (varianza variable).

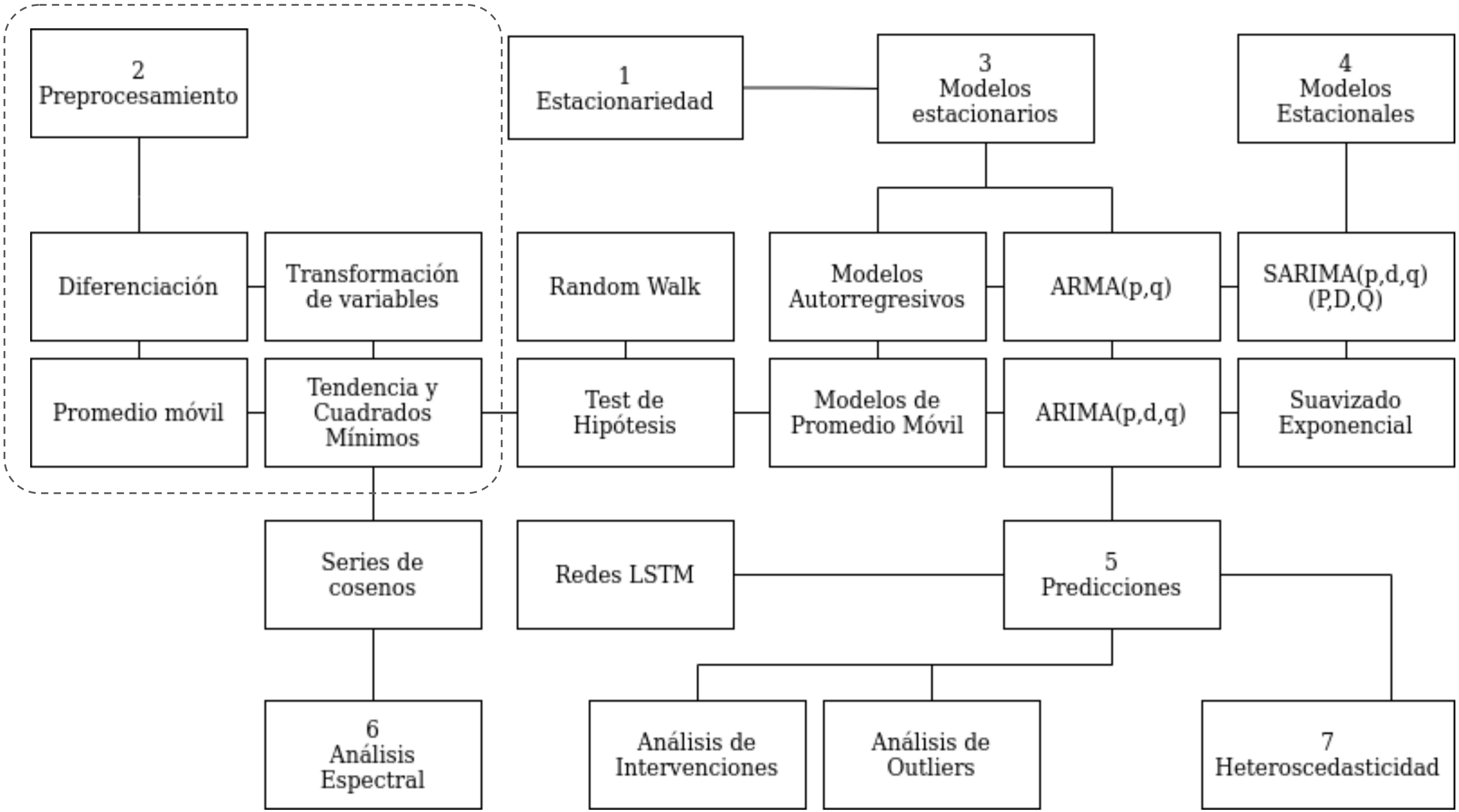
Preprocesamiento

Análisis de Series de Tiempo

Box and Jenkins plantean tres etapas de **modelado:**

1. **Especificaciones**
   a. Gráficos
   b. Estadísticas
   c. Contexto
2. **Ajuste**
   a. Parámetros
   b. Valores
3. **Diagnóstico**
   a. Testing

“everything should be made as simple as possible but no simpler”


2
Preprocesamiento
1
Estacionariedad
3
Modelos
estacionarios
4
Modelos
Estacionales
Diferenciación
Transformación
de variables
Random Walk
Modelos
Autorregresivos
ARMA(p,q)
SARIMA(p,d,q)
(P,D,Q)
Promedio móvil
Tendencia y
Cuadrados
Mínimos
Test de
Hipótesis
Modelos de
Promedio Móvil
ARIMA(p,d,q)
Suavizado
Exponencial
Series de
cosenos
Redes LSTM
5
Predicciones
6
Análisis
Espectral
Análisis de
Intervenciones
Análisis de
Outliers
7
Heteroscedasticidad

¿ Cuándo aplicar un preprocesamiento?

Cuando la distribución de la serie de tiempo cambia a lo largo del tiempo se dice que es **no estacionaria**.
En estos casos, pueden aplicarse transformaciones para que la serie de tiempo resultante sea aproximadamente estacionaria.

Algunos métodos:

1. Transformación de variables
2. Diferenciación
3. Promedio móvil

1. Transformación de variables

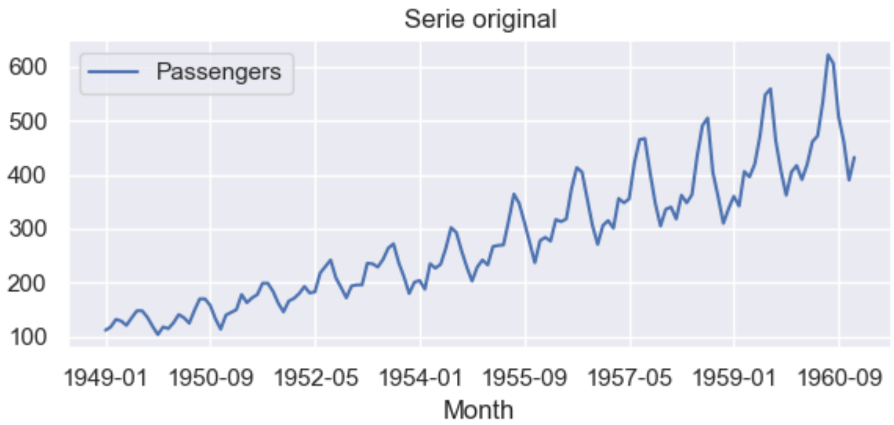
Muchas series de tiempo tienen la característica de que sus varianzas van aumentando a medida que avanza el tiempo.

Se pueden aplicar transformaciones a los puntos de la serie de tiempo.

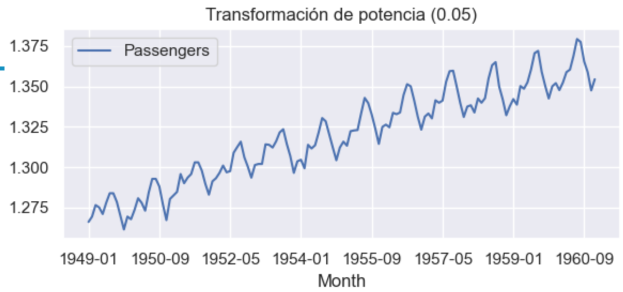
- Transformación Box-Cox

$$y_i^{(\lambda)} = \begin{cases} \frac{y_i^\lambda - 1}{\lambda} & si\ \lambda \neq 0, \\ \ln y_i & si\ \lambda = 0 \end{cases}$$

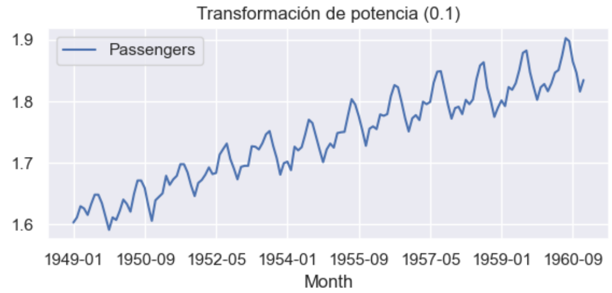
1. Ejemplos



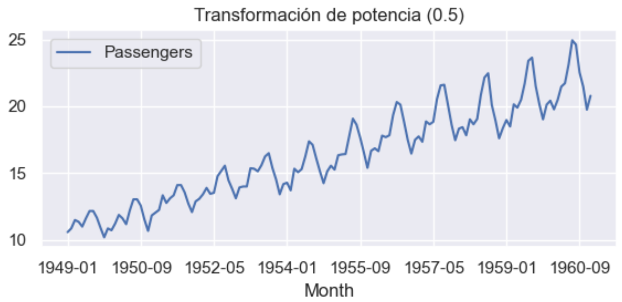
Serie original
Passengers
600
500
400
300
200
100
1949-01
1950-09
1952-05
1954-01
1955-09
1957-05
1959-01
1960-09
Month



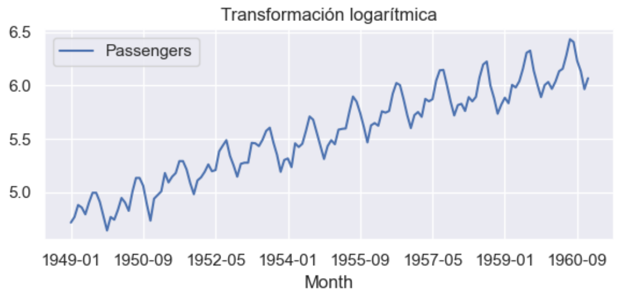
Transformación de potencia (0.05)
Passengers
1.375
1.350
1.325
1.300
1.275
1949-01
1950-09
1952-05
1954-01
1955-09
1957-05
1959-01
1960-09
Month



Transformación de potencia (0.1)
Passengers
1.9
1.8
1.7
1.6
1949-01
1950-09
1952-05
1954-01
1955-09
1957-05
1959-01
1960-09
Month



Transformación de potencia (0.5)
Passengers
25
20
15
10
1949-01
1950-09
1952-05
1954-01
1955-09
1957-05
1959-01
1960-09
Month



Transformación logarítmica
Passengers
6.5
6.0
5.5
5.0
1949-01
1950-09
1952-05
1954-01
1955-09
1957-05
1959-01
1960-09
Month

2. Diferenciación

Si la serie presenta un tendencia se puede analizar en su lugar la serie diferenciada. Dada una serie de tiempo y_n, $n = 0, 1, \ldots$ definimos

$$z_n = \Delta y_n = y_n - y_{n-1}$$

Motivación: si $y_n = an + b$, al diferenciar se obtiene una constante.

Observación: En general, si y_n se corresponde con un polinomio de grado n, diferenciando n veces se recupera una constante.

Ejemplos

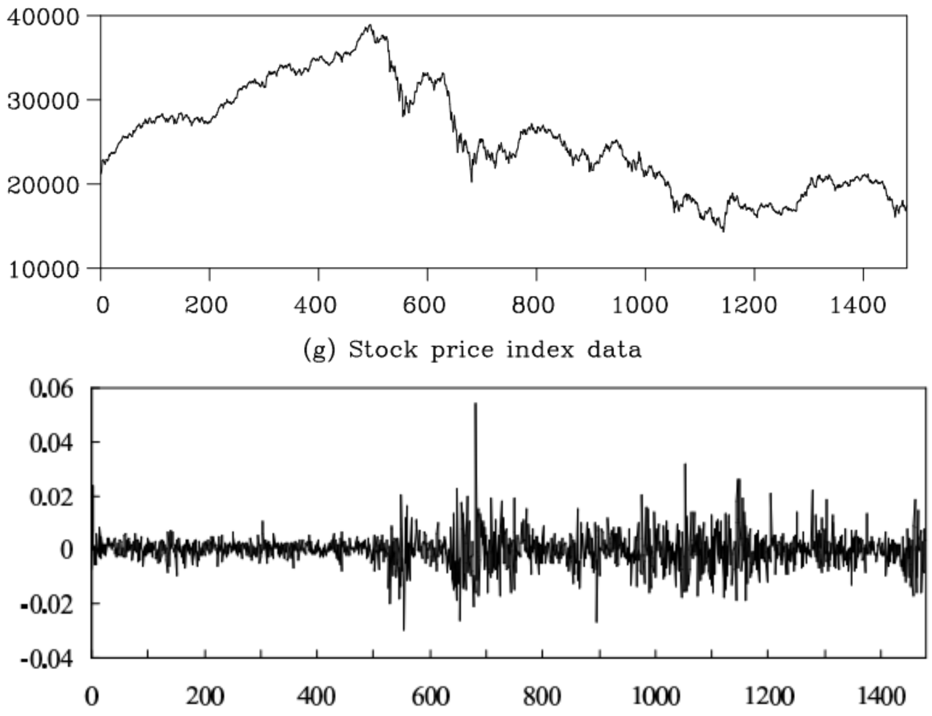


Figure 1.2: *Difference of the logarithm of the* Nikkei 225 *data.*

Ref.: Introduction to Time Series Modeling, *Genshiro Kiagawa*

3. Promedio móvil

Para una serie de tiempo y_n, el promedio móvil de $(2k+1)$ términos está dado por

$$T_n = \frac{1}{2k+1}\sum_{j=-k}^{k} y_{n+j}$$

Si modificamos la definición intercambiando la media por la mediana obtenemos la mediana móvil, definida como

$$T_n = \text{mediana}\{y_{-k}, \ldots, y_k\}$$

En general, la mediana móvil puede capturar cambios en la tendencia más rápido que el promedio móvil.

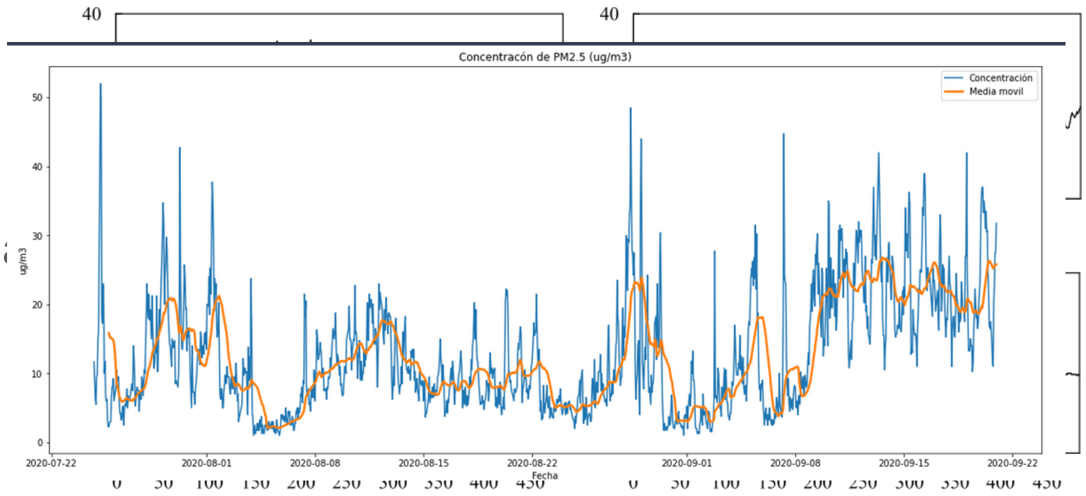


Figure 1.4 *Maximum temperature data and its moving average. Top left: original data, top right: moving average with $k = 5$, bottom left: $k = 17$, bottom right: $k = 29$.*

Ref.: Introduction to Time Series Modeling, *Genshiro Kiagawa*

Transformación logarítmica

¿Cuándo usar la transformación logarítmica?

- Cuando observamos que la varianza del proceso parece aumentar con el tiempo.
  En particular si $\mathbb{E}[Y_t] = \mu_t$ y $\sqrt{var(Y_t)} = \mu_t \sigma$, luego $\mathbb{E}[\log(Y_t)] \approx \log(\mu_t)$ y $var(\log(Y_t)) \approx \sigma^2$
- Si Y_t tiene cambios porcentuales relativamente estables entre un instante de tiempo y otro, y supongamos que $Y_t = (1 + X_t)Y_{t-1}$. Luego tomando el log

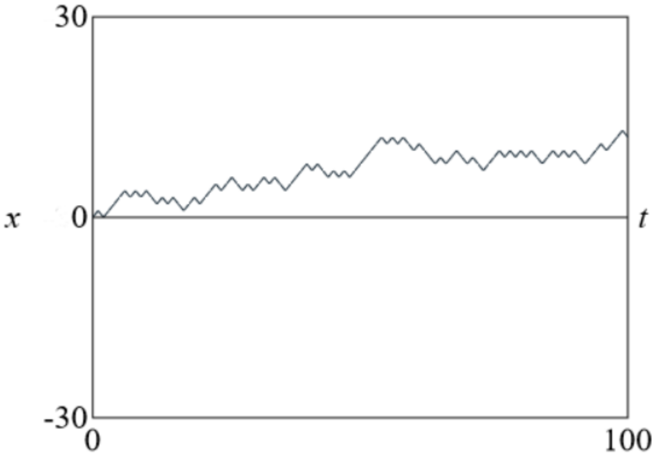
  $$\log(Y_t) = \log((1 + X_t)Y_{t-1}) = log(1 + X_t) + log(Y_{t-1}) \rightarrow \log(Y_t) - \log(Y_{t-1}) = \log\left(\frac{Y_t}{Y_{t-1}}\right) = \log(1 + X_t)$$

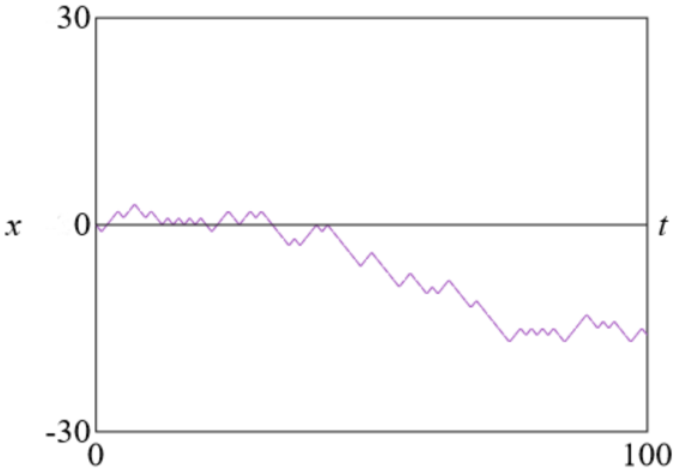
  Si además suponemos que X_t está acotado, $|X_t|<0.2$, sucede que $log(1 + X_t) \approx X_t$ y $\nabla(\log Y_t) \approx X_t$ va a ser relativamente estable y posiblemente se encuentre bien modelada por un poc. estacionario.

Análisis de tendencia

Tendencia estocástica vs. tendencia determinística

Si bien no hay una definición unificada de qué son las **tendencias estocásticas**, se puede decir que son aquellas que un observador podría hallar al analizar una realización de una serie de tiempo, pero que si se tiene una realización distinta esa tendencia cambia.





Tendencia estocástica vs. tendencia determinística

La **tendencia determinística** es aquella que viene dada por el modelo, y es fija para toda la serie de tiempo, sin importar que realización se tenga. *Por ejemplo las variaciones cíclicas a los largo de las distintas estaciones del año.*

En el caso de la tendencia determinística, podemos estimarla y descontarla de la serie de tiempo. Esta situación se puede modelar como

$$Y_t = X_t + \mu_t$$

donde μ_t es la tendencia determinística y X_t es una serie de tiempo de media cero alrededor de μ_t.

Algunos modelos comunes para tendencia

- Constante: $\mu_t = \mu \quad \forall t$
- Lineal: $\mu_t = \beta_0 + \beta_1 t \quad \forall t$
- Cuadrática: $\mu_t = \beta_0 + \beta_1 t + \beta_2 t^2 \quad \forall t$
- Cíclicas: $\mu_t = \mu_{t-T} \quad \forall t$, donde T es el período del ciclo. *Ejemplo: temperatura a lo largo del año tiene un período de T=12 meses*
- Coseno: $\mu_t = \beta \cos(2\pi f t + \phi) \quad \forall t$

¿Cómo estimar estas tendencias?

Suele emplearse el método de cuadrados mínimos para estimar los valores de los parámetros que describen la tendencia.

Si llamamos $f(t;\theta)$ a los modelos presentados anteriormente, buscamos hallar los parámetros θ que minimicen el ECM. Es decir, buscamos θ que minimice

$$Q(\theta) = \frac{1}{n}\sum_{i=1}^{n}(y_t - f(t;\theta))^2$$

Caso constante:

$$\begin{bmatrix} y_n \\ y_{n-1} \\ \vdots \\ y_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \mu + \begin{bmatrix} x_n \\ x_{n-1} \\ \vdots \\ x_1 \end{bmatrix}$$

Luego, la solución por c.m. es

$$\mu = \underbrace{\left(\begin{bmatrix} 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \right)}_{n}^{-1} \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} y_n \\ y_{n-1} \\ \vdots \\ y_1 \end{bmatrix} = \frac{1}{n} \sum_{t=1}^{n} y_t$$

Caso lineal

$$\begin{bmatrix} y_n \\ y_{n-1} \\ \vdots \\ y_1 \end{bmatrix} = \begin{bmatrix} 1 & n \\ 1 & n-1 \\ \vdots & \vdots \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} + \begin{bmatrix} x_n \\ x_{n-1} \\ \vdots \\ x_1 \end{bmatrix}$$

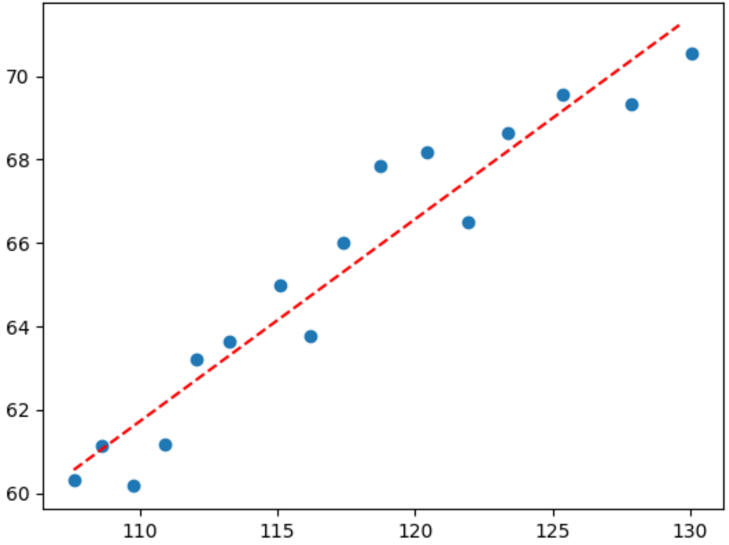
Y la solución por c.m resulta

$$\begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \end{bmatrix} = \left(\begin{bmatrix} 1 & n \\ 1 & n-1 \\ \vdots & \vdots \\ 1 & 1 \end{bmatrix}^T \begin{bmatrix} 1 & n \\ 1 & n-1 \\ \vdots & \vdots \\ 1 & 1 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & n \\ 1 & n-1 \\ \vdots & \vdots \\ 1 & 1 \end{bmatrix}^T \begin{bmatrix} y_n \\ y_{n-1} \\ \vdots \\ y_1 \end{bmatrix} =$$

$$\hat{\beta}_1 = \frac{\sum_{t=1}^{n} (y_t - \bar{y})(t - \frac{n+1}{2})}{\sum_{t=1}^{n} (t - \frac{n+1}{2})^2} \qquad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \frac{n+1}{2}$$

Caso lineal

```
from scipy.optimize import curve_fit
# define the true objective function
def objective(x, a, b):
    return a * x + b
```




```
dataframe = read_csv(url, header=None)
data = dataframe.values
# choose the input and output variables
x, y = data[:, 4], data[:, -1]
# curve fit
popt, _ = curve_fit(objective, x, y)
# summarize the parameter values
a, b = popt
print('y = %.5f * x + %.5f' % (a, b))
# plot input vs output
pyplot.scatter(x, y)
# define a sequence of inputs between the smalle
x_line = arange(min(x), max(x), 1)
# calculate the output for the range
y_line = objective(x_line, a, b)
# create a line plot for the mapping function
pyplot.plot(x_line, y_line, '--', color='red')
pyplot.show()
```

Caso cuadrático

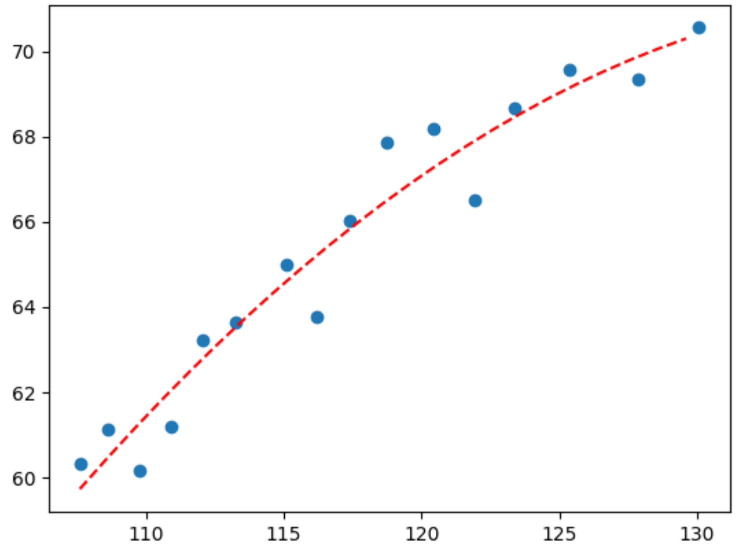
$$\begin{bmatrix} y_n \\ y_{n-1} \\ \vdots \\ y_1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & n & n^2 \\ 1 & n-1 & (n-1)^2 \\ \vdots & \vdots & \\ 1 & 1 & 1 \end{bmatrix}}_{A} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} x_n \\ x_{n-1} \\ \vdots \\ x_1 \end{bmatrix}$$

Y la solución por c.m. queda

$$\begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \hat{\beta}_2 \end{bmatrix} = (A^T A)^{-1} A^T \begin{bmatrix} y_n \\ y_{n-1} \\ \vdots \\ y_1 \end{bmatrix}$$

Caso cuadrático

```
# define the true objective function
def objective(x, a, b, c):
    return a * x + b * x**2 + c

# choose the input and output variables
x, y = data[:, 4], data[:, -1]
# curve fit
popt, _ = curve_fit(objective, x, y)
# summarize the parameter values
a, b, c = popt
print('y = %.5f * x + %.5f * x^2 + %.5f' % (a, b, c))
# plot input vs output
pyplot.scatter(x, y)
# define a sequence of inputs between the smallest and
x_line = arange(min(x), max(x), 1)
# calculate the output for the range
y_line = objective(x_line, a, b, c)
```

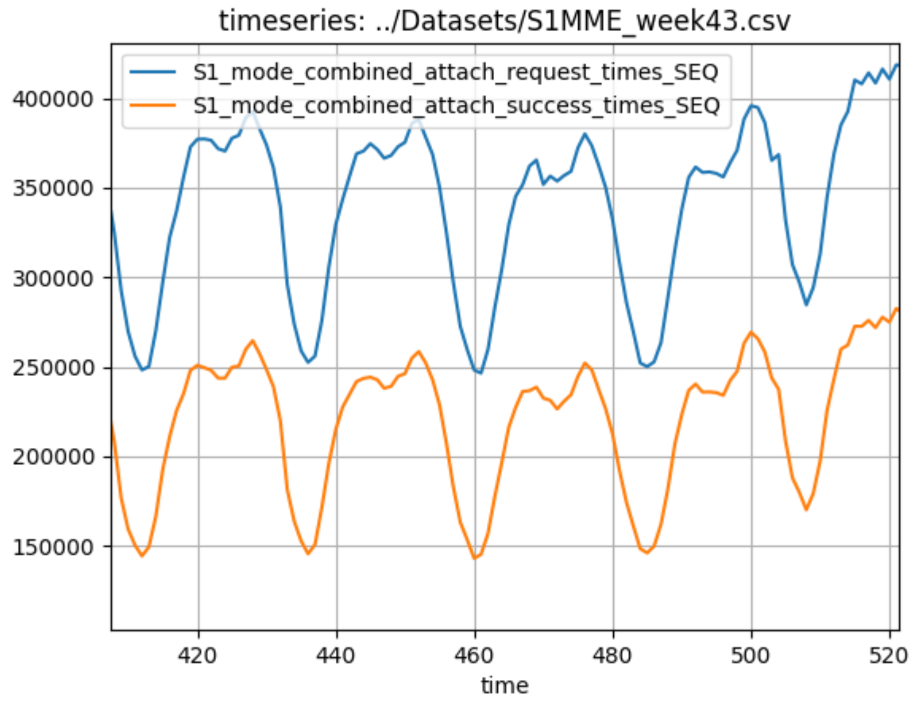


Caso cíclico

$$\mu_t = \begin{cases} \beta_1 & t = 1, 1+T, 1+2T, \dots \\ \beta_2 & t = 2, 2+T, 2+2T, \dots \\ \vdots & \\ \beta_T & t = T, 2T, 3T, \dots \end{cases}$$

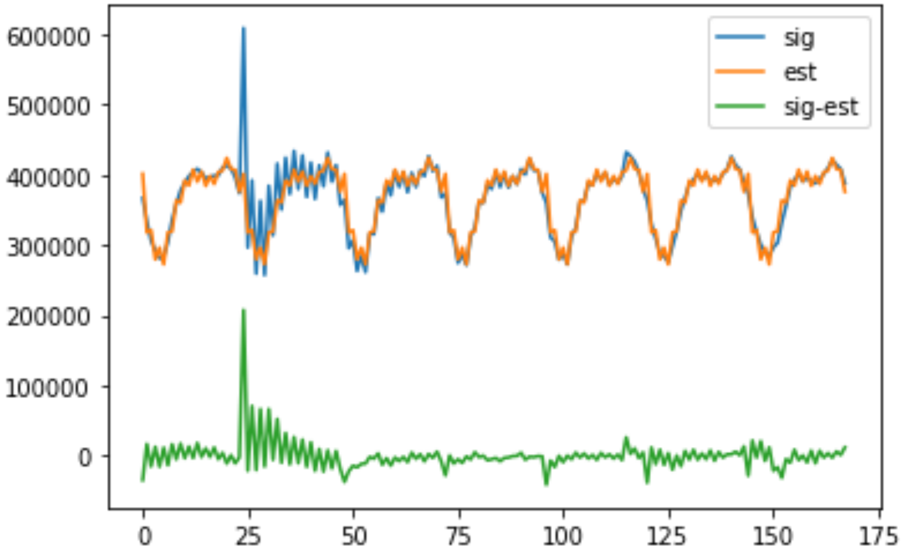
$$\begin{bmatrix} y_n \\ y_{n-1} \\ \vdots \\ y_1 \end{bmatrix} = A \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_T \end{bmatrix} + \begin{bmatrix} x_n \\ x_{n-1} \\ \vdots \\ x_1 \end{bmatrix}, \quad a_{i,j} = \begin{cases} 1 & (i \mod j) = 0 \\ 0 & (i \mod j) \neq 0 \end{cases}$$

Análisis de series de tiempo



Cada período presenta picos y valles bien marcados. La serie cae en sus valles a valores mínimos que suelen mantener un valor absoluto regular. En cada período se pueden observar dos picos a una distancia regular, donde el valor del segundo pico supera siempre al primero. Ambos picos tienen variaciones en su valor absoluto, a diferencia de los valles.

Ajuste por cuadrados mínimos: caso cíclico

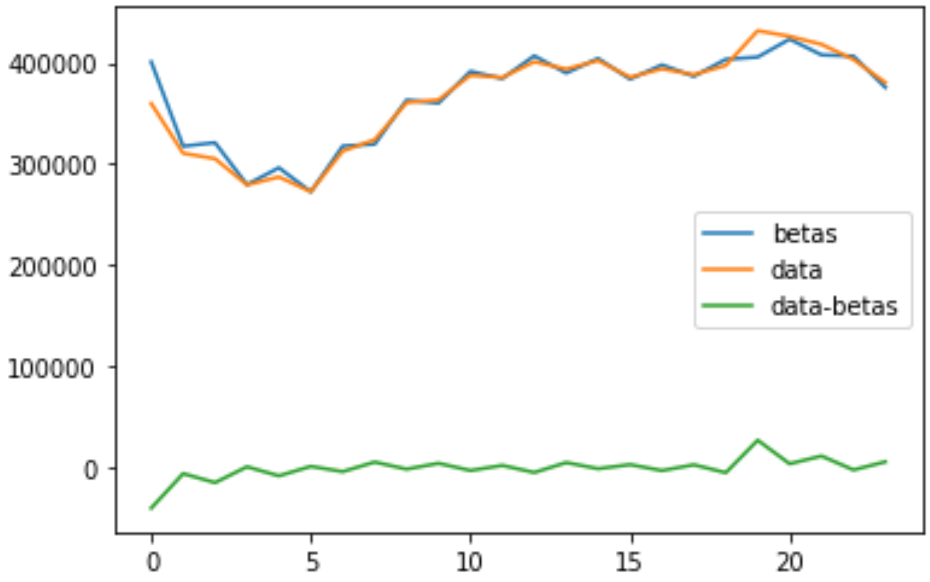



```python
dataset=sig
interval=1
diff = list()
for i in range(interval, len(dataset)):
```

```python
    value = dataset[i] - dataset[i - interval]
    diff.append(value)

plt.plot(sig)
plt.plot(diff)
plt.legend(['sig','diff'])
plt.show()

#repito la estimacion pero para la diferenciada
N=24 #hours
dataframe = pd.Series(pd.concat([pd.Series(diff[0]),pd
ts=pd.DataFrame(dataframe.values)
rows=int(len(ts)/N)
data = ts.values.reshape(rows,N)

betas=data.mean(axis=0)
```

[ver ejemplo: linearRegression24hs.py]

Ajuste por cuadrados mínimos: caso cíclico



Usando la técnica de cuadrados mínimos se pueden estimar los valores promedio de los períodos.

Luego si se resta a la serie original la serie estimada por cuadrados mínimos se obtiene una serie que puede servir para analizar estacionariedad.

Caso coseno

Reescribiendo la tendencia podemos llevarla a una expresión lineal:

$$\mu_t = \beta \cos(2\pi f t + \phi) = \beta_1 \cos(2\pi f t) + \beta_2 \cos(2\pi f t),$$
$$\beta_1 = \beta \cos(\phi),\ \beta_1 = \beta \sin(\phi)$$

$$\begin{bmatrix} y_n \\ y_{n-1} \\ \vdots \\ y_1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & \cos(2\pi f n) & \sin(2\pi f n) \\ 1 & \cos(2\pi f(n-1)) & \sin(2\pi f(n-1)) \\ \vdots & \vdots & \\ 1 & \cos(2\pi f 1) & \sin(2\pi f 1) \end{bmatrix}}_{A} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} x_n \\ x_{n-1} \\ \vdots \\ x_1 \end{bmatrix}$$

¡Cuidado!

Los resultados por cuadrados mínimos son válidos estrictamente sólo cuando Xt es un proceso blanco (las muestras temporales son independientes entre sí).

Sin embargo, se puede demostrar que si el proceso es estacionario los resultados son asintóticamente válidos, y sus varianzas coinciden con la de los estimadores de mínima varianza para los parámetros.

Estimando X_t

Los valores de X_t pueden ser estimados a partir de los residuos de la estimación de los parámetros:

$$\hat{x}_t = y_t - \hat{\mu}_t$$

Trabajo práctico

Trabajo Práctico

- Graficar una serie a partir de un dataset relevante. **Describir** observaciones.
- **Descomponer** una serie de tiempo usando el modelo aditivo de cuatro componentes.
- Extraer la **tendencia** y ajustar un **modelo determinístico**. Explicar su relación con el contexto. Obtener conclusiones acerca de la validez del modelo.
- **Entrega clase 4:** Evaluar si la serie original es **estacionaria**. Aplicar transformaciones (**preprocesamiento**), graficar autocorrelación, autocorrelación parcial y extraer conclusiones.

Trabajo Práctico

- **Entrega clase 6:** A partir de las transformaciones propuestas ajustar distintos **modelos (S)ARIMA**. Extraer orden, parámetros, coeficientes numéricos y análisis de la bondad del modelo.
- Ajustar y predecir usando **redes neuronales LSTM**. Comparar con **predicciones** usando SARIMA y extraer conclusiones.
- Reailzar el **análisis espectral** de la serie original. Hallar las frecuencias principales y comparar con las **componentes cíclica y estacional** usando la descomposición.
- **Entrega clase 8:** Presentación incluyendo introducción, gráficos, modelos propuestos, expresiones analíticas y conclusiones.

Modelos para series de tiempo estacionarias

Proceso lineal general

Sean $\{Y_t\}$ la serie de tiempo observada y $\{e_t\}$ una serie de ruido blanco no observable. $\{e_t\}$ es una secuencia de v.a. i.i.d, con media cero y varianza σ_e^2.

Un proceso lineal general es aquel que puede representarse como una combinación lineal de términos presentes y pasados del proceso de ruido blanco:

$$Y_t = e_t + a_1 e_{t-1} + a_2 e_{t-2} + \dots$$

Si la cantidad de términos a sumar es infinita se pide que $\sum_{i=1}^{\infty} a_i^2 < \infty$.

Se puede demostrar que:

- $\mathbb{E}[Y_t] = 0$
- $C_k = cov(Y_t, Y_{t-k}) = \sigma_e^2 \sum_{i=1}^{\infty} a_i a_{i+k}, \quad k \geq 0$

Modelo de promedios móviles

Un **modelo de promedio móvil (MA)** es un modelo lineal general donde existe una cantidad finita términos $a_i \neq 0$.

Diremos que $\{Y_t\}$ es un modelo de promedio móvil de orden q (MA(q)) si

$$Y_t = e_t - a_1 e_{t-1} - \ldots - a_q e_{t-q}$$

Los parámetros de este modelo son los pesos $a_1, \ldots, a_q$.

Para estos modelos,

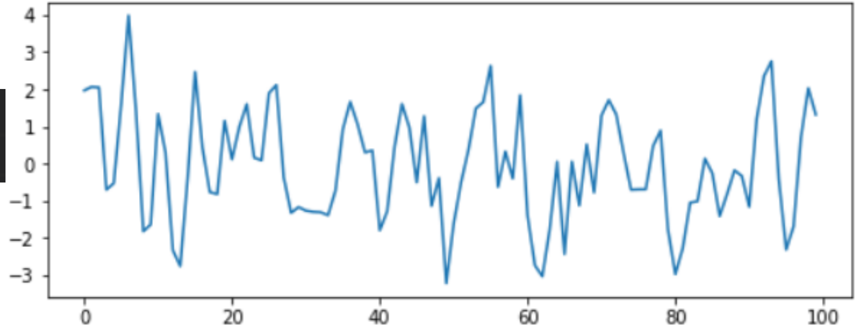
$$C_0 = (1 + a_1^2 + \ldots + a_q^2)\sigma_e^2$$

$$\frac{Cov(Y_t, Y_{t-k})}{C_0} \quad R_k = \begin{cases} \dfrac{-a_k + a_1 a_{k+1} + a_2 a_{k+2} + \ldots + a_{1-k} a_q}{1 + a_1^2 + \ldots + a_q^2} & k = 1, 2, \ldots q \\ 0 & k > q \end{cases}$$

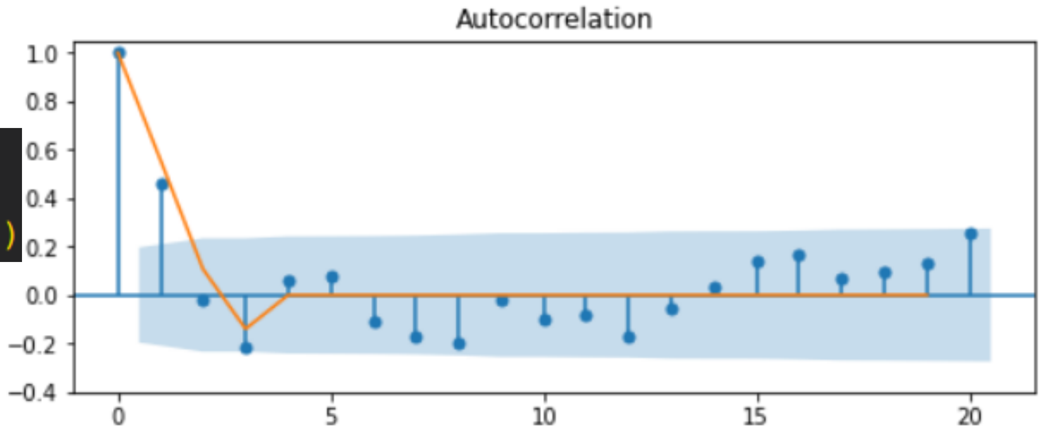
σ_e^2

Ejemplo!

```
ma_ts = arma_generate_sample(
    ar=[1], ma=np.r_[1,ma_coef], nsample =100)
```




```
plot_acf(ma_ts, ax= ax)
plt.plot(arma_acf(ar=[1],
            ma=np.r_[1,ma_coef], lags=20))
```



Modelo autoregresivo

Los modelos autorregresivos incluyen regresiones sobre sí mismos.

Diremos que Y_t es un modelo autorregresivo de orden p (AR(p)) si cumple que

$$Y_t = a_1 Y_{t-1} + \ldots + a_p Y_{t-p} + \boxed{e_t} \rightarrow \text{innovación}$$

Suponemos entonces que para cada t, e_t es independiente de $Y_{t-1}, Y_{t-2}, \ldots$

El modelo AR(p) tiene asociado su polinomio característico definido como:

$$\phi(x) = 1 - a_1 x + a_2 x^2 + \ldots + a_p x^p$$

Se puede demostrar que el proceso es AR(q) es estacionario si las raíces $\phi(x)$ se encuentran fuera del círculo unitario (módulo mayor a 1)

Modelo AR(1)

Si p=1, tenemos que $Y_t = a_1 Y_{t-1} + e_t$, luego $\mathbb{E}[Y_t] = 0$.

Puedo tomar la varianza miembro a miembro para obtener

$$var(Y_t) = a_1^2 var(Y_{t-1}) + \sigma_e^2 \rightarrow C_0 = a_1^2 C_0 + \sigma_e^2 \Rightarrow C_0 = \frac{\sigma_e^2}{1-a_1^2}$$

$$|a_1| < 1$$

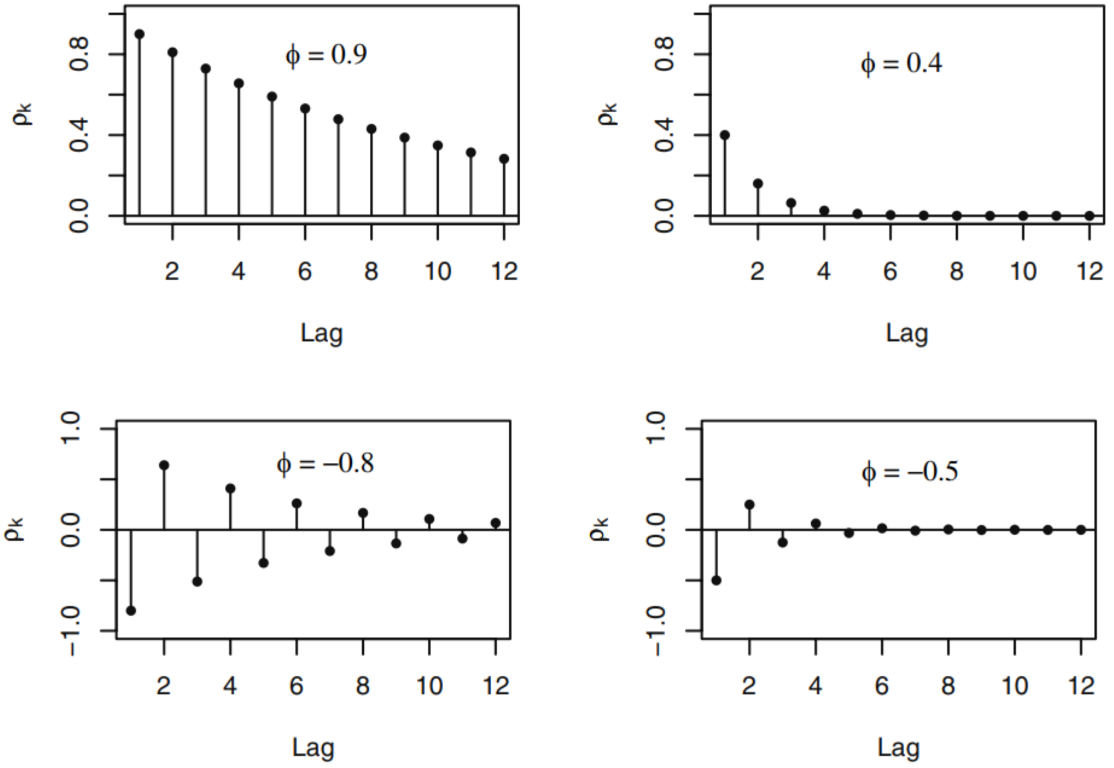
Además,

$$\begin{aligned} C_k = cov(Y_{t-k}, Y_t) &= cov(Y_{t-k}, a_1 Y_{t-1} + e_t) \\ &= a_1 \underbrace{cov(Y_{t-k}, Y_{t-1})}_{C_{k-1}} + \underbrace{cov(Y_{t-k}, e_t)}_{0} \\ &= a_1 C_{k-1} \end{aligned}$$

A partir del valor semilla de C_0 obtenemos de forma recursiva que

$$C_k = a_1^k \frac{\sigma_e^2}{1-a_1^2}$$

Modelo AR(1)



Modelo AR(1) como un proceso lineal general

Para comprender el modelo, la expresión dada para el modelo AR(1) es muy útil, sin embargo para muchas otras cosas es necesario llevarlo a la forma de un proceso general lineal.

Para eso comenzamos reemplazando Y_{t-1} por $a_1 Y_{t-2} + e_{t-1}$:

$$Y_t = a_1 (a_1 Y_{t-2} + e_{t-1}) + e_t = a_1^2 Y_{t-2} + a_1 e_{t-1} + e_t$$

aplicando la misma idea (k-1) veces:

$$Y_t = a_1^k e_{t-k} + a_1^{k-1} e_{t-k+1} + \ldots + a_1 e_{t-1} + e_t$$

Asumiendo $|a_1| < \infty$ e incrementando k sin límite tenemos que

$$Y_t = e_t + a_1 e_{t-1} + a_1^2 e_{t-2} + a_1^3 e_{t-3} + \ldots$$

$$Y_t = e_t + \psi_1 e_{t-1} + \psi_2 e_{t-2} + \psi_3 e_{t-3} + \ldots$$

Modelo AR(p)

Se puede mostrar que una condición necesaria (pero no suficiente) para que el proceso sea estacionario es que $a_1 + a_2 + \ldots + a_p < 1$ y $|a_p| < 1$.

Al igual que como hicimos para el AR(1), podemos calcular la función de autocovarianza y autocorrelación:

$$\begin{aligned} C_k = cov(Y_t, Y_{t-k}) &= cov(a_1 Y_{t-1}, Y_{t-k}) + \ldots + cov(a_p Y_{t-p}, Y_{t-k}) + cov(e_t, Y_{t-k}) \\ &= a_1 C_{k-1} + \ldots + a_p C_{k-p} \end{aligned}$$

$$R_k = \frac{cov(Y_t, Y_{t-k})}{C_0} = a_1 R_{k-1} + \ldots + a_p R_{k-p}$$

Modelo AR(p)

Evaluando para k=1,...,p y recordando que $R_0 = 1$ y $R_k = R_{-k}$ obtenemos las **ecuaciones de Yule-Walker (Y-W)**:

$$\begin{cases} R_1 = a_1 + a_2 R_1 + \ldots + a_p R_{p-1} \\ \quad \vdots \\ R_p = a_1 R_{p-1} + a_2 R_{p-2} + \ldots + a_p R_p \end{cases}$$

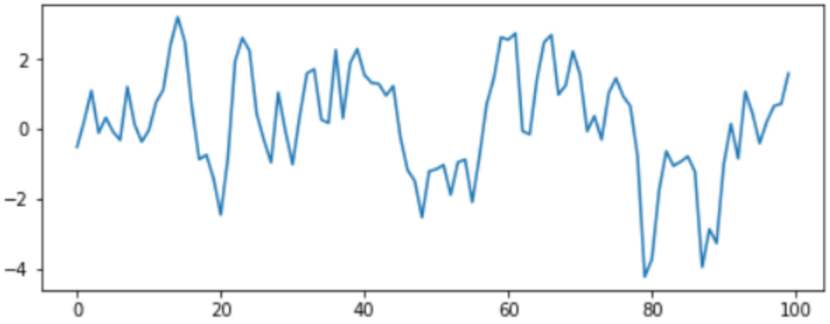
Dado los valores de $a_1, \ldots, a_p$, se puede resolver el sistema de ecuaciones para hallar $R_1, \ldots, R_p$

Finalmente, podemos usar estos valores para hallar $C_0 = a_1 C_1 + \ldots + a_p C_p + \sigma_e^2$ observando que

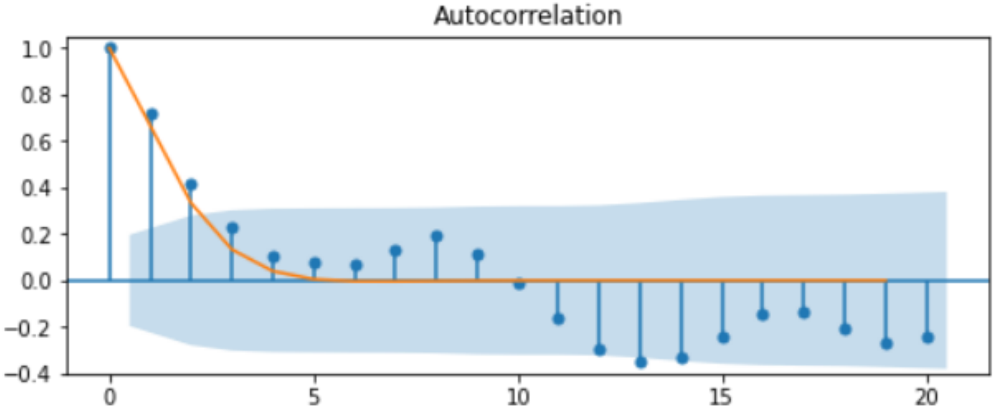
$$C_0 = a_1 R_1 C_0 + \ldots + a_p R_p C_0 + \sigma_e^2 \Rightarrow C_0 = \frac{\sigma_e^2}{1 - a_1 R_1 - \ldots - a_p R_p}$$

Ejemplo!

```
ar_coef  = np.array([0.8,-0.2])
ar_ts = arma_generate_sample(ar=np.r_[1,-ar_coef],
                             ma=[1], nsample =100)
```




```
plot_acf(ar_ts, ax=ax)
plt.plot(arma_acf(ar=np.r_[1,-ar_coef],
         ma=[1], lags=20))
```



Modelo ARMA

El modelo arma es una combinación de un proceso AR con un MA. Diremos que $\{Y_t\}$ sigue un modelo ARMA(p,q) si

$$Y_t = a_1 Y_{t-1} + \ldots + a_p Y_{t-p} + e_t - b_1 e_{t-1} - \ldots - b_q e_{t-q}$$

Si se satisfacen las condiciones de estacionariedad, el modelo ARMA(p,q) puede reescribirse como un proceso lineal general con coeficientes $\psi_1, \psi_2, \ldots$ dados por:

$$\begin{cases} \psi_0 = 1 \\ \psi_1 = -b_1 + a_1 \\ \psi_2 = -b_2 + a_2 + a_1 \psi_1 \\ \vdots \\ \psi_j = -b_j + a_p \psi_{j-p} + a_{p-1} \psi_{j-p+1} + \ldots + a_1 \psi_{j-1} \end{cases}$$

$$\psi_j = 0, si\ j < 0\ y\ b_j = 0\ si\ j > q$$

ARMA(p,q)

Se puede ver que la función de autocorrelación está dada por:

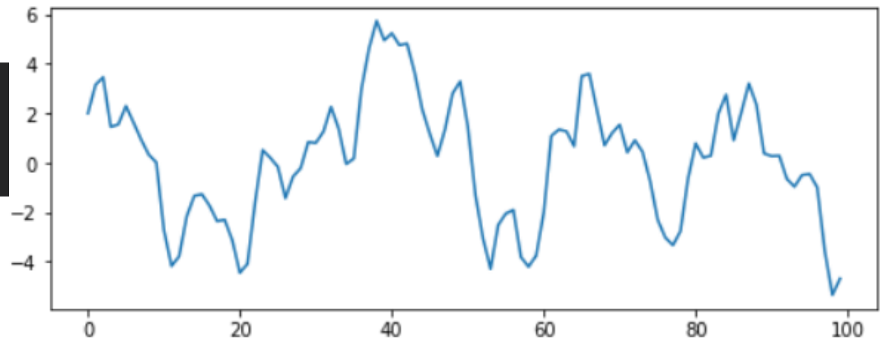
$$\begin{cases} C_0 = a_1 C_1 + a_2 C_2 + \ldots - \sigma_e^2 (b_0 + b_1 \psi_1 + \ldots + b_q \psi_q) \\ C_1 = a_1 C_0 + a_2 C_1 + \ldots + a_p C_{p-1} - \sigma e^2 (b_1 + b_2 \psi_1 + \ldots + b_1 \psi_{q-1}) \\ \quad \vdots \\ C_p = a_1 C_{p-1} + a_2 C_{p-2} + \ldots + a_p C_0 - \sigma_e^2 (b_p + b_{p+1} \psi_1 + \ldots + b_q \psi_{q-p}) \end{cases}$$

Si k>q entonces la expresión puede simplificarse como:

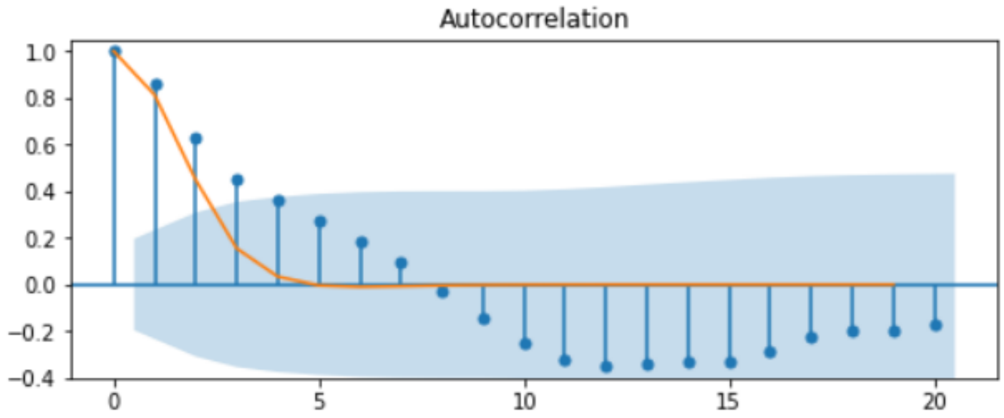
$$C_k = a_1 C_{k-1} + a_2 C_{k-2} + \ldots + a_p C_{k-p}$$

Ejemplo!

```
arma_ts = arma_generate_sample(
    ar=np.r_[1,-ar_coef], ma=np.r_[1,ma_coef],
    nsample =100)
```



```
plot_acf(arma_ts, ax=ax)
plt.plot(arma_acf(ar=np.r_[1, -ar_coef],
        ma=np.r_[1,ma_coef], lags=20))
```



Conclusiones

- Hemos visto un camino para modelar series de tiempo basado en **tres partes: identificación, ajuste y test**.
- Usamos **transformaciones,** extraemos información y **proponemos un modelo,** estocástico o determinista
- Vimos modelos con **tendencia determinística** con ejemplos: **lineal, cuadrático, cíclico y coseno**
- Hicimos ajustes por **cuadrados mínimos** para encontrar los **valores de los parámetros** a estimar
- En la segunda parte, vimos **modelos estocásticos: MA, AR** y ARMA
- Estudiamos las propiedades a partir de la **Autocorrelación y Autocovarianza**
- **Generalizamos** los resultados **de forma analítica**
- Buscamos **identificar los parámetros** y sus valores para modelos estocásticos
- Nos vamos a concentrar la próxima clase en probar y medir la **bondad del modelo** propuesto