



SKILLING
CENTER

TECMILENIO



Aplicaciones Financieras

Modelado de series de tiempo





El senior Armando Rodríguez es director de ventas de una empresa productora de cerveza, y tiene entre sus funciones estimar la demanda para el próximo trimestre para preparar los insumos necesarios, producir las cantidades adecuadas y tener el mínimo de faltantes o excedentes de producto, pues ya sabe que en estos casos se incurre en costos y posible deterioro en la preferencia de los clientes.

¿Cuáles son los principales factores que debe considerar el Sr. Rodríguez para hacer sus estimaciones? ¿Le serán de utilidad los datos históricos de ventas de trimestres anteriores para pronosticar próximos periodos? ¿Deberá enfocar el análisis y las estimaciones por estaciones del año?





Series de tiempo

- Las series de tiempo son frecuentemente utilizadas en los negocios para identificar patrones de cambios de alguna variable sobre el tiempo. Tienen información oculta que a simple vista no es fácil identificar.
- Representan una secuencia de valores en orden cronológico, regularmente en intervalos igualmente espaciados.
- El análisis técnico en finanzas es una aplicación típica de series de tiempo.
- Shumway y Stoffer (2017) mencionan que las series de tiempo son un proceso para describir estadísticamente el carácter de los datos que están fluctuando de una manera aleatoria en el tiempo.
- El interés en modelar las series de tiempo es para generar pronósticos a partir de información contenida en ellas.



Descomposición de series de tiempo

- La frecuencia de los datos (diario, semanal, mensual, etc.) depende de la naturaleza de la variable y del análisis que se pretende realizar.
- La descomposición de series de tiempo es para entender e interpretar fuerzas involucradas y para hacer pronósticos del estado futuro del proceso.
- Pal y Prakash (2017) mencionan que en la exploración de las series de tiempo se encuentran ciertas cualidades que ayudan a comprender su comportamiento y desempeño:
 - Tendencia general
 - Estacionalidad
 - Cambios cíclicos
 - Variaciones inesperadas





- Las series de tiempo pueden ser representadas de la siguiente manera:

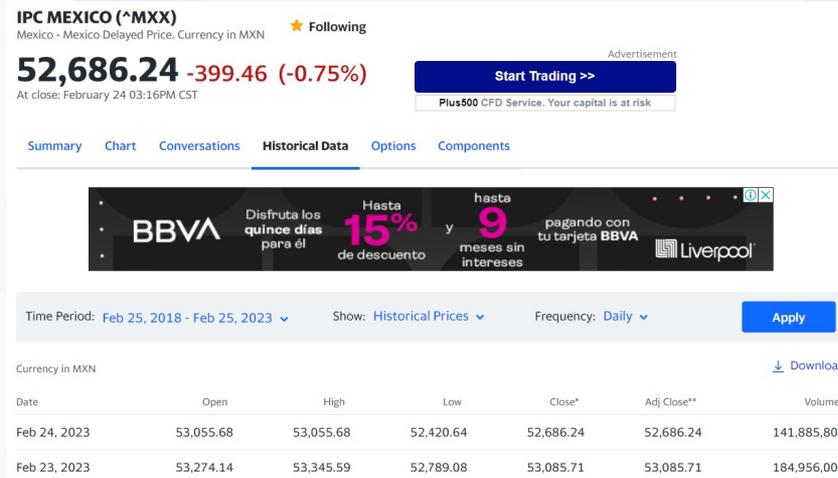
$$Y_t = T_t + S_t + C_t + R_t$$

En donde:

- Y_t = Conjunto de las cualidades.
- t : periodos de tiempo, $t = 1,2,3...N$ sucesivos e igualmente espaciados.
- T_t : tendencia general.
- S_t : estacionalidad, patrones de comportamiento repetitivos en determinados periodos de tiempo.
- C_t : ciclicidad, se refiere a periodos con fluctuaciones sobre la tendencia, debido a condiciones propias del sistema o externas.
- R_t : cambios inesperados. Puede incluir efectos aleatorios y efectos causados por eventos extraordinarios como huracanes o epidemias.

Tendencia general

- Puede ser de crecimiento o decrecimiento, aunque en periodos más cortos tenga fluctuaciones, en general se identifica como positiva o negativa. Es mejor observarla en una gráfica.
- Por ejemplo, se pretende analizar la serie de tiempo del índice de la Bolsa Mexicana de Valores. De Yahoo Finance se descarga la información del símbolo ^MXX con un período de 5 años con frecuencia diaria.



Fuente: Yahoo Finance. (s.f.). Recuperado de <https://finance.yahoo.com/quote/%5EMXX/history?p=%5EMXX>

- Para este ejemplo, el archivo descargado se guarda como ^MXX.csv, y se va a utilizar Python para leerlo y analizarlo.

- El código Python para analizar la tendencia de los valores es el siguiente:
 - **# nombre del archivo file_name**
 - **file_name = "^MXX.csv"**
 - **# importar las librerías a utilizar**
 - **import pandas as pd**
 - **import matplotlib.pyplot as plt**
 - **Import numpy as np**
 - **# lectura del archivo con la librería pandas**
 - **df = pd.read_csv(file_name)**
 - **df.dropna(inplace=True)**
 - **# impresión de los primeros 5 fechas de los registros**
 - **print(df.head())**

Los primeros cinco registros muestran la fecha, el precio de apertura, el precio máximo del día, el precio mínimo del día, el precio de cierre, el precio de cierre ajustado y el volumen operado.

	Date	Open	High	Low	Close	Adj Close	Volume
0	2018-02-22	48552.28906	49086.62891	48538.89844	48969.64063	48969.64063	174797900
1	2018-02-23	49176.30859	49176.30859	48488.17969	48643.42969	48643.42969	118771200
2	2018-02-26	48662.28906	48713.98828	48240.32031	48463.44141	48463.44141	142142400
3	2018-02-27	48481.89844	48532.55078	47886.35938	47970.19141	47970.19141	162311300
4	2018-02-28	48055.39063	48062.12891	47231.78906	47437.92969	47437.92969	268429100

Un resumen estadístico del precio de cierre (Close) se obtiene con el comando `.describe()`

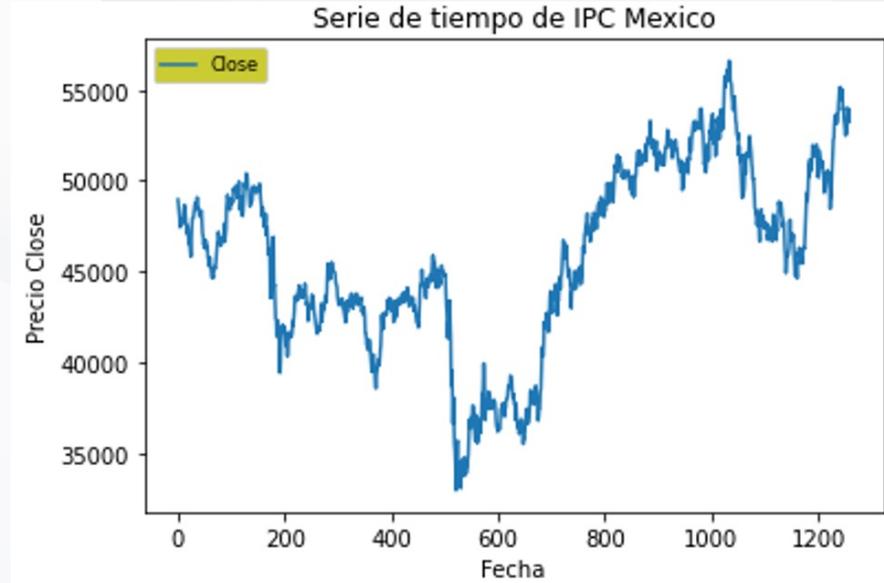
- `df["Close"].describe()`

Como se puede ver en la siguiente tabla, hay 1259 registros, con un valor promedio de 45,895 unidades, desviación estándar de 5,071 unidades, el valor mínimo de 32,964. Ordenados los valores de menor a mayor, el 25 % de los datos (el primer cuartil) resulta en 42,912, el 50 % (la mediana) es de 46,294, el 75 % (el tercer cuartil) es 50,029, el valor máximo de 56,609 y el tipo de datos es numérico con decimales flotante.

count	1259.000000
mean	45895.600847
std	5071.366396
min	32964.218750
25%	42912.695315
50%	46294.429690
75%	50029.576175
max	56609.539060
Name: Close, dtype: float64	

La gráfica de los valores del índice con frecuencia diaria se genera con el siguiente código:

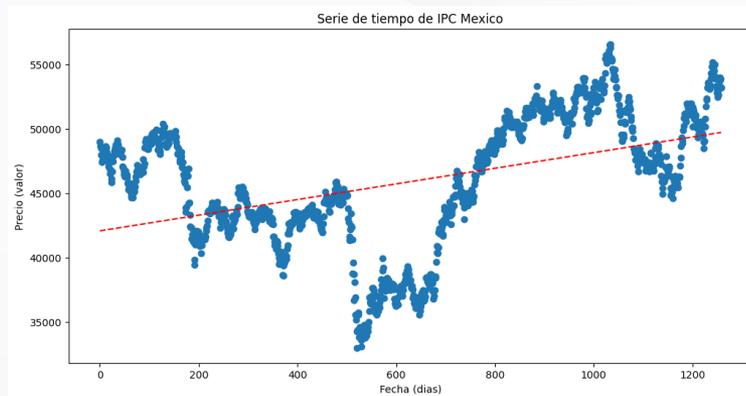
- **# grafica de la serie de tiempo**
- **plt.plot(df["Close"])**
- **plt.legend(['Close'],loc="upper left", facecolor="y",fontsize=8)**
- **plt.xlabel("Fecha")**
- **plt.ylabel("Precio Close")**
- **plt.title("Serie de tiempo de IPC Mexico")**



Fuente: Yahoo Finance. (s.f.). Recuperado de <https://finance.yahoo.com/quote/%5EMXX/history?p=%5EMXX>

- La tendencia es mixta en períodos cortos, pero positiva en general.
- Por esta razón, para revelar la tendencia general se necesitan periodos de tiempo largos.

- En la siguiente gráfica se agrega una línea de regresión lineal, en la cual, visualmente se observa la inclinación de la línea de tendencia.
- Esta línea de tendencia general es un modelo de la serie de tiempo que puede ser utilizado para hacer predicciones de largo plazo.
- El código Python para generar la gráfica de valores y la línea de tendencia es el siguiente:
 - **# grafica de valores en el tiempo**
 - **plt.figure(figsize=(12, 6))**
 - **plt.scatter(df.index, df['Close'])**
 - **plt.xlabel('Fecha (dias)')**
 - **plt.ylabel('Precio (valor)')**
 - **plt.title("Serie de tiempo de IPC Mexico")**
 - **# ajustar la linea de tendencia**
 - **z = np.polyfit(df.index, df['Close'], 1)**
 - **p = np.poly1d(z)**
 - **plt.plot(df.index, p(df.index), "r--")**
 - **plt.show()**



- De acuerdo con Pal y Prakash (2017), la diferencia entre los valores reales y los valores en la línea de tendencia general se denominan residuales, los cuales pueden ser utilizados en análisis posterior para identificar y confirmar sus propiedades como la estacionalidad, comportamiento cíclico y variaciones irregulares.

Fuente: Yahoo Finance. (s.f.). Recuperado de <https://finance.yahoo.com/quote/%5EMXX/history?p=%5EMXX>



Estacionalidad

- Se identifica por patrones repetitivos en periodos particulares de la serie de tiempo.
- Por ejemplo, las ventas de una empresa constantemente se incrementan o decrecientan en determinadas estaciones del año.
- Una serie de tiempo puede tener tanto estacionalidad como tendencia.
- Analytics University (s.f.) menciona que una serie de tiempo es estrictamente estacional si el comportamiento de los datos es idéntico cuando la serie de tiempo es desfasada en h periodos, $v_{1+h}, v_{2+h}, \dots, v_{k+h}$, en donde h puede representar los períodos trimestrales del año.
- Para identificar la estacionalidad, las propiedades que se buscan son los siguientes:
 - a) Movimientos de promedios por periodos estacionales de la serie.
 - b) Cambios en los valores y en la varianza.
 - c) Presencia de outliers o datos extraordinarios.
- Como ejemplo, se analizan los precios de cierre diario de la acción de Walmart (WMT), desde el 1 de enero del 2020 al 24 de febrero del 2023.
- Se transforman los datos para generar nuevas series de tiempo.
- Las siguientes gráficas se integran en un dashboard mediante el software Tableau.





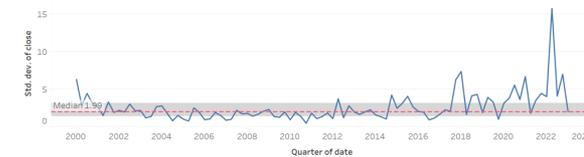
- En la gráfica superior izquierda se observa la tendencia positiva de los precios diarios.
- La gráfica superior derecha muestra los precios promedio de cada trimestre. Cada trimestre tiene su propio color de línea.
- La tercera gráfica muestra la desviación estándar de los precios por trimestre. Aquí es en donde se comienza a observar estacionalidad, con valores alrededor de 1.99, incrementando su volatilidad a partir del 2015.
- Las gráficas box plot muestran la dispersión tanto de los precios promedio como de la desviación estándar para cada trimestre. El precio promedio está entre 72 y 75, y la desviación estándar entre 1.9 y 2.7.
- La gráfica inferior muestra las diferencias en porcentaje del precio diario con respecto al día anterior. Regularmente el porcentaje promedio tiende a ser valor cero, lo cual confirma la estacionalidad de esta serie de tiempo transformada.

Análisis de estacionalidad de la acción de Walmart (WMT)

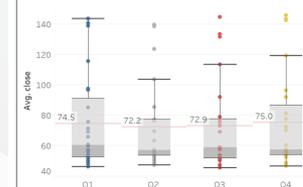
Datos históricos diarios del 1 de enero 2000 al 24 de febrero 2023



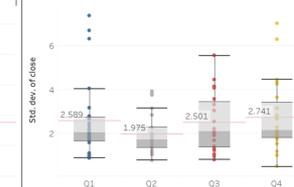
Desviación estándar de los precios diarios por trimestre



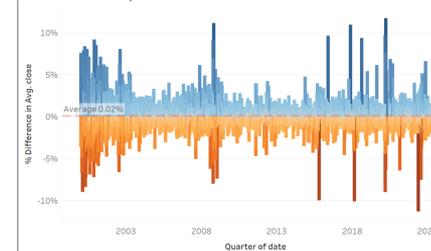
Concentración de precios promedio



Concentración de la desviación estándar



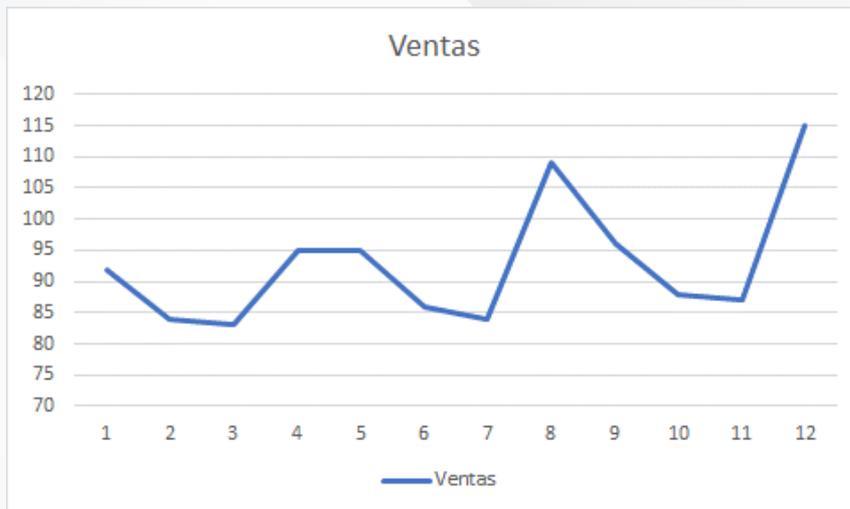
% de cambio del precio diario



Esta pantalla se obtuvo directamente del software que se está explicando en la computadora, para fines educativos.

Corrección de estacionalidad en series de tiempo

- La corrección de estacionalidad se utiliza para propósitos de pronosticar valores futuros, determinando el componente de tendencia de la serie de tiempo sin el efecto de estacionalidad.
- Se estandarizan los datos con base en el promedio del periodo completo en donde están los periodos estacionales.
- Considera la siguiente información de ventas trimestrales con propiedades de tendencia y estacionalidad.



Esta pantalla se obtuvo directamente del software que se está explicando en la computadora, para fines educativos.

year	quarter	sales
2019	Q1	72
2019	Q2	64
2019	Q3	63
2019	Q4	75
2020	Q1	75
2020	Q2	66
2020	Q3	64
2020	Q4	89
2021	Q1	76
2021	Q2	68
2021	Q3	67
2021	Q4	95

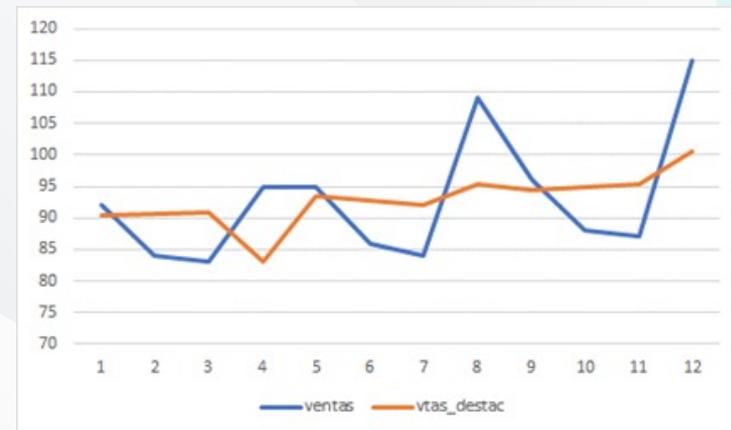
- Procedimiento:
 - a) Calcula el promedio de ventas de cada año con la variable `avg_year`.
 - b) Divide los valores de ventas entre su respectivo promedio del año, y se obtiene el índice de estacionalidad para cada trimestre, con la variable `index_stat`.
 - c) Calcula el índice estacional promedio de cada trimestre, en la variable `avg_index_stat`. Por ejemplo, para el Q1 es $(1.0511 + 1.0204 + 0.9935) / 3 = 1.0217$.

year	quarter	sales	avg_year	index_stat
2019	Q1	72	68.5	1.0511
2019	Q2	64	68.5	0.9343
2019	Q3	63	68.5	0.9197
2019	Q4	75	68.5	1.0949
2020	Q1	75	73.5	1.0204
2020	Q2	66	73.5	0.8980
2020	Q3	64	73.5	0.8707
2020	Q4	89	73.5	1.2109
2021	Q1	76	76.5	0.9935
2021	Q2	68	76.5	0.8889
2021	Q3	67	76.5	0.8758
2021	Q4	95	76.5	1.2418

quarter	avg_index_stat
Q1	1.0217
Q2	0.9071
Q3	0.8888
Q4	1.1825

- Procedimiento (continuación):
 - d) Por último, la corrección de la serie estacional se determina dividiendo los valores originales entre sus respectivos índices estacionales promedio, en la variable `vtas_nostac`. Por ejemplo, el valor de 2019 Q1 se obtiene $72 / 1.0217 = 70.4738$.
 - e) La gráfica muestra la serie original y la serie con corrección de estacionalidad.

year	quarter	sales	avg_year	index_stat	vtas_nostac
2019	Q1	72	68.5	1.0511	70.4738
2019	Q2	64	68.5	0.9343	70.5583
2019	Q3	63	68.5	0.9197	70.8855
2019	Q4	75	68.5	1.0949	63.4231
2020	Q1	75	73.5	1.0204	73.4102
2020	Q2	66	73.5	0.8980	72.7632
2020	Q3	64	73.5	0.8707	72.0106
2020	Q4	89	73.5	1.2109	75.2620
2021	Q1	76	76.5	0.9935	74.3891
2021	Q2	68	76.5	0.8889	74.9682
2021	Q3	67	76.5	0.8758	75.3861
2021	Q4	95	76.5	1.2418	80.3359



Esta pantalla se obtuvo directamente del software que se está explicando en la computadora, para fines educativos.



Para incrementar tu comprensión del tema, realiza lo siguiente:

1. Descarga los precios históricos diarios de la acción de la empresa Exxon Mobil Corporation, en un periodo de 3 años.
2. Desarrolla el código Python para resolver los siguientes incisos:
 - a) Muestra los cinco últimos registros de la serie.
 - b) Muestra los valores estadísticos descriptivos del precio de cierre.
 - c) Elabora la gráfica de la cotización de cierre diario y la línea de tendencia.
 - d) Elabora la gráfica de box plot para mostrar la dispersión de los valores con el promedio trimestral del precio de cierre.
 - e) Elabora la gráfica de cambios porcentuales diario.
3. Presenta un reporte con los resultados y elabora tus comentarios y observaciones del comportamiento de la acción en el periodo de análisis.



Con este tema estás iniciando el aprendizaje para analizar el comportamiento histórico de variables, identificando sus propiedades, como la tendencia, estacionalidad, ciclicidad y valores extraordinarios, lo cual te será de mucha utilidad para el siguiente paso, que son los pronósticos de valores futuros.

Así también, con la información que el Sr. Armando Rodríguez obtiene mediante el análisis de los datos históricos de las ventas de cerveza, ha logrado identificar el comportamiento estacional de las ventas, además de tener una valiosa herramienta para hacer las estimaciones de futuras ventas con mayor precisión, lo cual también utilizará como soporte y recomendación para el departamento de producción.





- Analytics University. (2016, 18 de marzo). *Stationary Process | Strict Stationarity & Weak Stationarity || Time Series* [Archivo de video]. Recuperado de <https://www.youtube.com/watch?v=eKa7veMDUzo>
- Pal, A., y Prakash, P. (2017). *Practical time series analysis*. PACKT.
- Shumway, R., y Stoffer, D. (2017). *Time series analysis and its applications* (4a ed.). Springer International Publishing AG.
- Yahoo Finance. (s.f.). Recuperado de <https://finance.yahoo.com/quote/%5EMXX/history?p=%5EMXX>

Aplicaciones Financieras

Modelado de series de tiempo
(segunda parte)





La señora Domínguez es propietaria de una fábrica de calzado y gracias a su visión y dedicación, modelos confortables (porque son como guantes para los pies, como ella los denomina) y precios competitivos, actualmente está interesada en consolidar su crecimiento, para lo cual está contemplando hacer análisis de series de tiempo para desarrollar pronósticos de las ventas en periodos futuros, y con esto, afinar sus presupuestos y hacer más eficiente su operación.

- ¿Qué consideraciones debe tomar en cuenta para realizar su análisis de las ventas para modelarlas como series de tiempo?
- ¿Qué periodo de tiempo de ventas y con qué periodicidad le conviene considerar en su análisis?
- ¿De qué forma puede validar que sus pronósticos son precisos?





- Las series de tiempo a simple vista ocultan mucha información valiosa que se puede identificar aplicando diversos métodos estadísticos para explorar e interpretar determinados comportamientos.
- Entendiendo estos comportamientos es posible pronosticar valores futuros con considerable precisión.
- En el ambiente financiero, específicamente el análisis técnico, utiliza diversas herramientas estadísticas para obtener señales de compra/venta de instrumentos financieros, como estrategia para administrar portafolios de inversión.
- Como expresan Hyndman y Athanasopoulos (2021), los datos en las series de tiempo pueden mostrar una variedad de patrones, y es de mucha utilidad el dividirlos en algunos componentes, cada uno representando una categoría de patrón subyacente.
- Por ejemplo, la tendencia de los valores originales en el tiempo se puede determinar con una regresión lineal o incluso polinomial.



Modelado de series de tiempo con métodos de suavizado

- El método de suavización de los valores originales como una segunda línea, facilita hacer pronósticos de valores futuros a corto plazo.
- La suavización de series de tiempo es por medio de promedio simple o ponderado, asignando pesos iguales o diferentes a los valores históricos.



Promedio móvil simple

- El promedio móvil simple determina el promedio de un número específico k de valores anteriores a cada punto del tiempo hasta el final de los puntos. Al valor de k se le conoce como ancho de ventana.
- Valores mayores de k hacen mayor suavización dado que nuevos datos tienen menos influencia relativa sobre el total de datos.
- El cálculo del promedio móvil simple es mediante la siguiente fórmula:

$$SMA = \frac{1}{K} \sum x_i$$

En donde SMA (*simple moving average*) es la nueva serie de tiempo con el promedio de k observaciones anteriores y x_i son los valores originales incluidos en la ventana k .



Como ejemplo, con los valores del tipo de cambio peso/dólar de 12 meses, del 8 de marzo del 2022 al 3 de marzo del 2023, el cálculo en Python para ventana de 10 y 30 días es el siguiente:

```
- # nombre del archivo con file_name
- file_name = "MXN=X_12M.csv"
- # importar las librerías a utilizar
- import pandas as pd
- import matplotlib.pyplot as plt
- # lectura del archivo
- df = pd.read_csv(file_name)

- # calculo del promedio movil con ventana k = 10
- sma_close = df['Close'].rolling(10).mean()
- df['sma10_close'] = sma_close

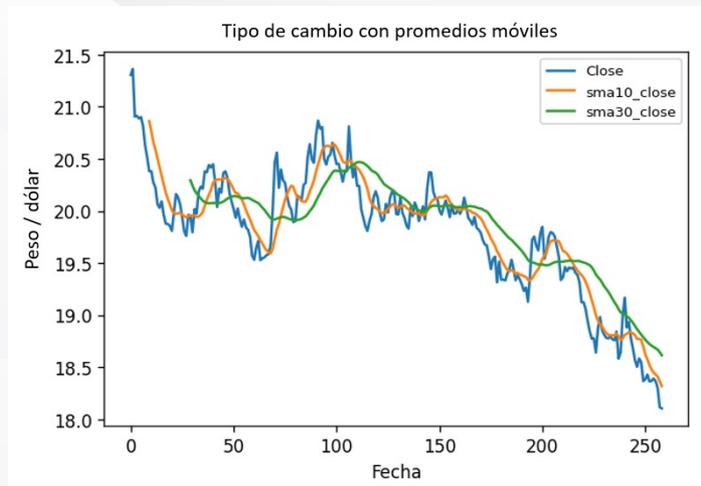
- # calculo del promedio movil con ventana k = 30
- sma_close = df['Close'].rolling(30).mean()
- df['sma30_close'] = sma_close
- sma_window = df[['Date','Close','sma10_close','sma30_close']]
- # impresion de los ultimos valores de la serie original y los promedios moviles simples
- print(sma_window.tail(5))

- # grafica de los promedios moviles
- plt.figure(figsize=(6, 4))
- plt.plot(df["Close"])
- plt.plot(df["sma10_close"])
- plt.plot(df["sma30_close"])
- plt.legend(['Close','sma10_close','sma30_close'],loc="upper right",
facecolor="w",fontsize=8)
- plt.xlabel("Fecha")
- plt.ylabel("Pesos / dolar")
- plt.title("Tipo de cambio con promedios moviles")
```

Los resultados son los siguientes:

Últimos valores originales y los promedios móviles

	Date	Close	sma10_close	sma30_close
254	2023-02-27	18.391331	18.451015	18.694663
255	2023-02-28	18.366899	18.431201	18.680963
256	2023-03-01	18.300940	18.410815	18.669611
257	2023-03-02	18.115971	18.363957	18.644341
258	2023-03-03	18.105301	18.319615	18.615094



Esta pantalla se obtuvo directamente del software que se está explicando en la computadora, para fines educativos.



Promedio móvil ponderado exponencialmente

- Se asignan diferentes pesos a los promedios móviles de acuerdo con su relativa importancia de valores más recientes o antiguos sobre el resultado actual y futuro.
- La fórmula del promedio móvil ponderado es la siguiente:

$$WMA = \frac{1}{K} \sum w_i x'_t$$

En donde WMA (*weighted moving average*) es la nueva serie de tiempo con las ponderaciones w del promedio móvil x' .

- La suavización exponencial aplica determinada constante de suavización α al valor actual y un peso suavizado $(1-\alpha)$ al promedio móvil anterior, y se expresa así:

$$EMA = \alpha x_t + (1 - \alpha)x'_{t-1}$$

En donde EMA (*exponential moving average*) es la nueva serie y α es el factor de suavización con valores entre 0 y 1, y controla la razón de decrecimiento de los pesos.



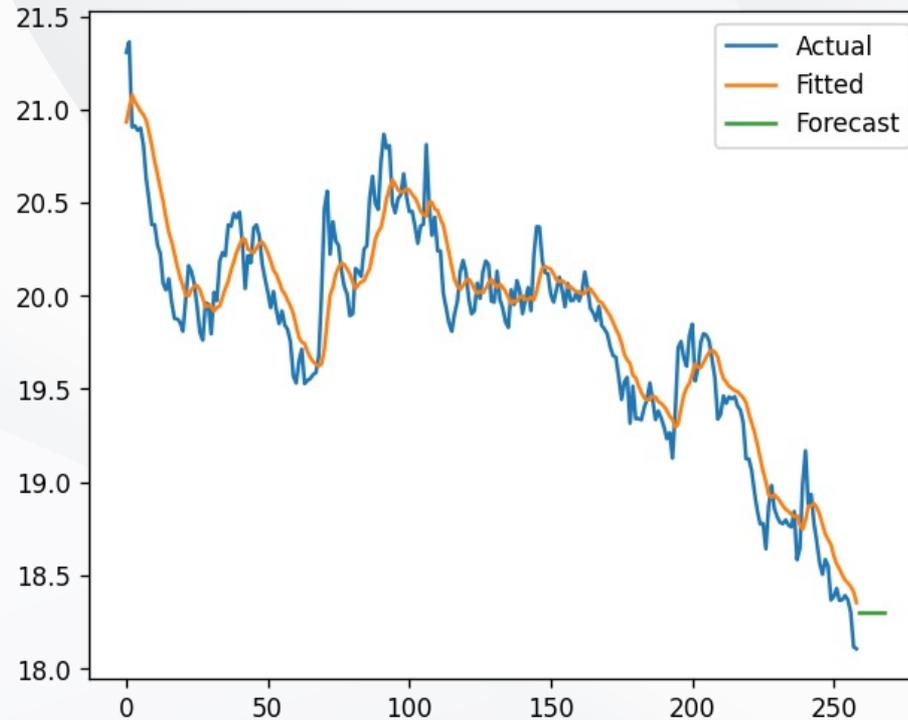
- En Python se utiliza el módulo de modelos estadísticos, en particular el SimpleExpSmoothing para determinar la nueva serie de tiempo con suavización exponencial y pronosticar 10 días.
 - **import csv**
 - **import pandas as pd**
 - **import numpy as np**
 - **import matplotlib.pyplot as plt**
 - **from statsmodels.tsa.api import SimpleExpSmoothing**
 - **# obtener datos del archivo**
 - **df = pd.read_csv('MXN=X_12M.csv')**
 - **data = df['Close']**

 - **# desarrollo del modelo de suavizacion exponencial**
 - **model = SimpleExpSmoothing(data)**
 - **fit = model.fit(smoothing_level=0.2)**

 - **# determinar pronostico para los proximos 10 dias**
 - **forecast = fit.forecast(10)**
 - **# grafica de la serie original, la linea suavizada y pronostico**
 - **plt.figure(figsize=(6, 5))**
 - **plt.plot(data, label='Actual')**
 - **plt.plot(fit.fittedvalues, label='Fitted')**
 - **plt.plot(forecast, label='Forecast')**
 - **plt.legend()**
 - **plt.show()**



- El resultado se muestra en la siguiente gráfica, que incluye la serie original, la serie suavizada y el pronóstico de los próximos 10 días.



Esta pantalla se obtuvo directamente del software que se está explicando en la computadora, para fines educativos.

Modelado de series con autoregresión

- El modelo ARIMA (*Autoregressive Integrated Moving Average*) es un proceso más estricto en las series de tiempo.
- Consta de tres componentes:

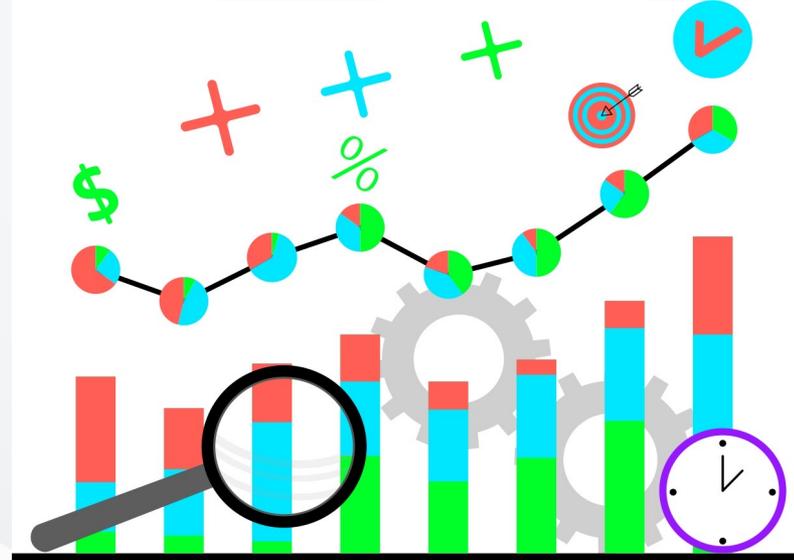


- Un modelo ARIMA es típicamente denotado como ARIMA (p,d,q), en donde p representa el número de valores anteriores que se quieren incluir en la regresión, d representa la cantidad de valores para determinar las diferencias y q representa el tamaño de la ventana del promedio móvil de los residuales.



Pronósticos usando modelos ARIMA

- Dependiendo del orden de p , d y q , el modelo se comporta diferente. En este caso, se sigue utilizando el tipo de cambio del peso/dólar de Estados Unidos, en un periodo de 12 meses.



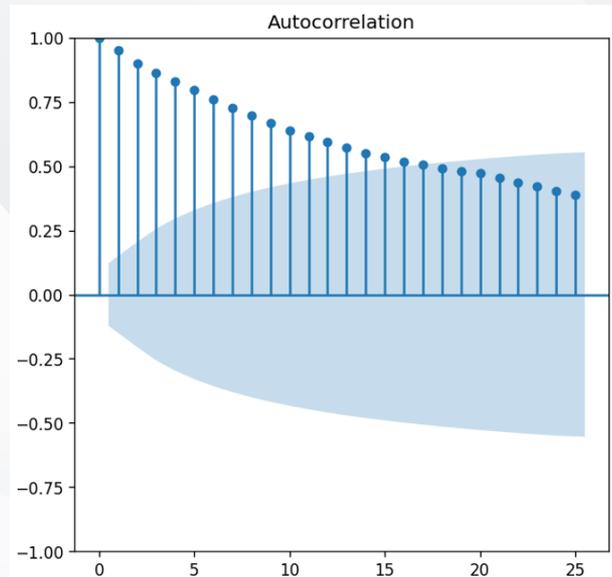


Desarrollo del código Python para el modelo ARIMA:

- **# importar las librerías**
- **import pandas as pd**
- **import matplotlib.pyplot as plt**
- **# lectura del archivo**
- **df = pd.read_csv('MXN=X_12M.csv', names=['Close'], header=0)**

- **# determina la funcion de autocorrelacion (ACF)**
- **from statsmodels.graphics.tsaplots import plot_acf**
- **plt.figure(figsize=(6, 5))**
- **plot_acf(df.Close)**

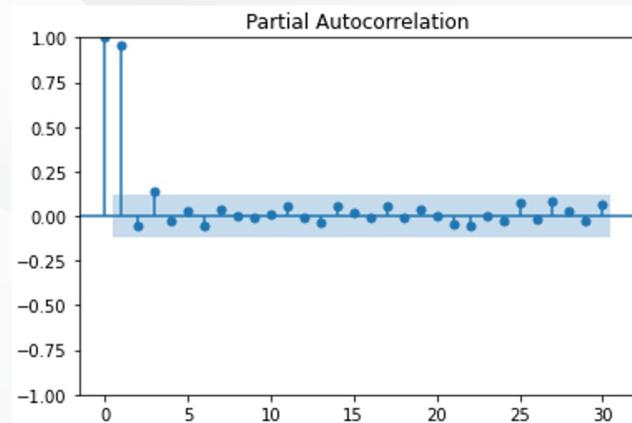
- En la gráfica resultante de autocorrelación, el eje vertical es el grado de correlación $[-1,1]$ y en el eje horizontal son diferentes periodos anteriores y posteriores (k) notando, como era de esperarse, que valores más recientes (menores valores de k) tienen mayor índice de correlación.



Esta pantalla se obtuvo directamente del software que se está explicando en la computadora, para fines educativos.

- Adicionalmente, la gráfica muestra intervalos de confianza del 95% en la curva sombreada. La autocorrelación fuera de estos intervalos de confianza es estadísticamente significativa, mientras que los que están dentro de estos intervalos son influenciados por situaciones de aleatoriedad. Se observa que valores de k mayores de 17 (eje horizontal), el índice de correlación está dentro de los intervalos de confianza, los cuales van aumentando a medida que aumenta k , lo cual es una señal de no estacional.

- Por lo tanto, se continúa el análisis con la autocorrelación parcial que resuelve este problema, midiendo la correlación entre $x(t)$ y $x(t+h)$ cuando la influencia de variables intermedias ha sido eliminada.
- La siguiente gráfica muestra la autocorrelación parcial:
 - **from pandas import read_csv**
 - **from matplotlib import pyplot**
 - **from statsmodels.graphics.tsaplots import plot_pacf**
 - **series = read_csv('MXN=X_12M_Close.csv', header=0, index_col=0)**
 - **plot_pacf(series, lags=30)**
 - **pyplot.show()**

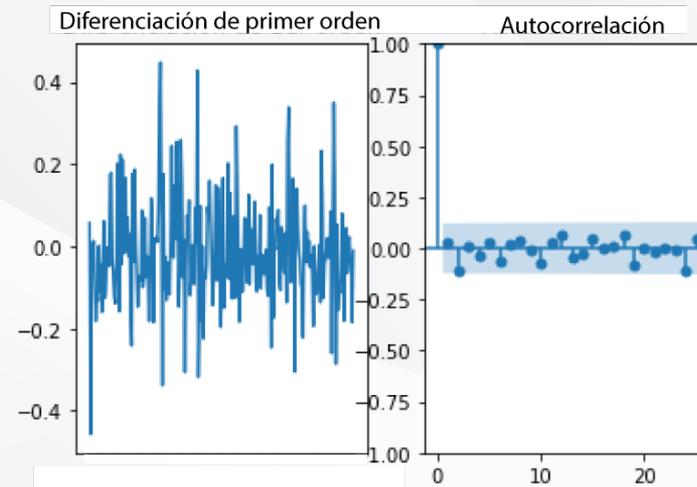


Esta pantalla se obtuvo directamente del software que se está explicando en la computadora, para fines educativos.

- La correlación parcial cuando $k=0$ (eje horizontal) siempre es 1.0. En la gráfica se observa que solamente cuando $k=1$ es estadísticamente significativa, mientras que el resto de los retrasos (lags) están dentro de los intervalos de confianza. Por lo tanto, para el periodo de tiempo del análisis, el tipo de cambio peso/dólar tiene modelos AR de primer orden.

- Se continúa con el siguiente componente, que es la integración. Este determina las diferencias sucesivas de valores actuales con el anterior, lo cual, es el componente de integración de primer orden, las cuales se espera tengan promedio y varianza constante y, por lo tanto, pueden ser tratadas como serie estacional, de lo contrario, se procede con la diferenciación de segundo orden, y así sucesivamente.
- Desarrollo del código Python para el segundo componente del modelo ARIMA:
 - **# diferenciacion de 1er orden**
 - **f = plt.figure()**
 - **ax1 = f.add_subplot(121)**
 - **ax1.set_title('Diferenciacion de 1er orden')**
 - **ax1.plot(df.Close.diff())**

 - **# Autocorrelacion de las diferencias**
 - **ax2 = f.add_subplot(122)**
 - **plot_acf(df.Close.diff().dropna(), ax=ax2)**
 - **plt.figure(figsize=(5,4))**
 - **plt.show()**



Esta pantalla se obtuvo directamente del software que se está explicando en la computadora, para fines educativos.

- El siguiente paso es determinar los parámetros óptimos para el modelo ARIMA para el tipo de cambio peso/dólar.
 - **# determima los parametros optimos del modelo ARIMA(p,d,q)**
 - **from statsmodels.tsa.arima_model import ARIMA**
 - **!pip install pmdarima**
 - **import pmdarima as pm**
 - **val = df['Close'].values**
 - **model = pm.auto_arima(val, start_p=1, start_q=1,**
 - **test='adf', # prueba adf para determinar el valor optimo para 'd'**
 - **max_p=3, max_q=3, # maximo valor de p y q**
 - **m=1, # frecuencia de las series**
 - **d=None, # el modelo determina 'd'**
 - **seasonal=False, # no estacionalidad**
 - **start_P=0,**
 - **D=0,**
 - **trace=True,**
 - **error_action='ignore',**
 - **supress_warnings=True,**
 - **stepwise=True)**
 - **print(model.summary())**
- El resultado del modelo ARIMA(p,d,q) óptimo es ARIMA(1,1,1), es de primer orden en los tres componentes.

```
Best model: ARIMA(1,1,1)(0,0,0)[0] intercept
Total fit time: 3.057 seconds

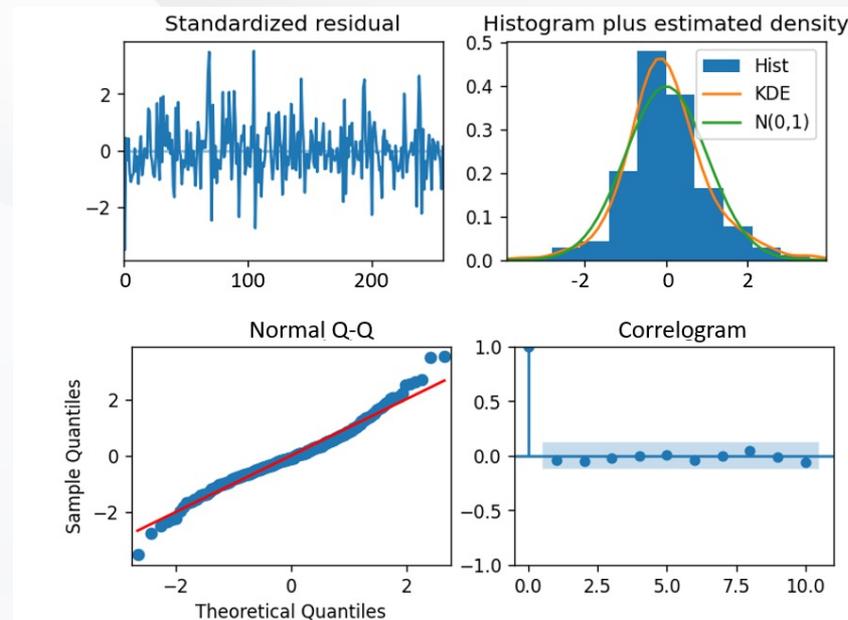
=====
SARIMAX Results
=====
Dep. Variable:          y          No. Observations:      259
Model:                SARIMAX(1, 1, 1)  Log Likelihood       166.374
Date:                 Wed, 08 Mar 2023  AIC                  -324.747
Time:                 06:28:23         BIC                  -310.535
Sample:               0               HQIC                 -319.032
                    - 259
Covariance Type:      opg
=====
              coef    std err          z      P>|z|      [0.025    0.975]
-----
intercept    -0.0217     0.015    -1.439     0.150     -0.051     0.008
ar.L1        -0.7681     0.170    -4.523     0.000     -1.101    -0.435
ma.L1         0.8520     0.153     5.582     0.000     0.553     1.151
sigma2        0.0161     0.001    14.656     0.000     0.014     0.018
=====
Ljung-Box (L1) (Q):                0.47  Jarque-Bera (JB):                27.11
Prob(Q):                            0.49  Prob(JB):                       0.00
Heteroskedasticity (H):              0.78  Skew:                            0.33
Prob(H) (two-sided):                 0.25  Kurtosis:                        4.45
=====
```

Esta pantalla se obtuvo directamente del software que se está explicando en la computadora, para fines educativos.

- En seguida se comprueba la validez del modelo, verificando que el comportamiento de los residuales tiene una distribución normal. El siguiente conjunto de gráficas confirman el modelo óptimo identificado en base a los residuales, dado que la diferencia entre los residuales tiende a cero, el histograma muestra que tienen una distribución Normal, la prueba de normalidad QQ y la regresión de nivel 1.

Comprobación de los resultados del modelo óptimo:

- **model.plot_diagnostics(figsize=(7,5))**
- **plt.show()**



Esta pantalla se obtuvo directamente del software que se está explicando en la computadora, para fines educativos.

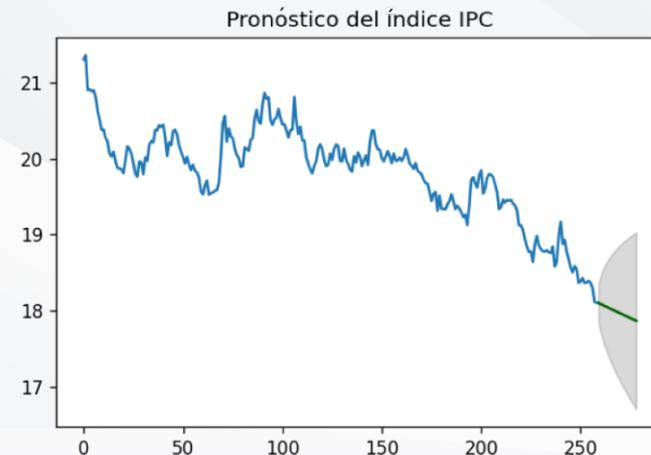
Por último, se utiliza el modelo ARIMA(1,1,1) obtenido para pronosticar los valores del tipo de cambio peso/dólar para los próximos 20 días.

- **# pronostico de 20 días**
- **n_periods = 20**
- **fc, confint = model.predict(n_periods=n_periods, return_conf_int=True)**
- **index_of_fc = np.arange(len(val), len(val)+n_periods)**

- **# ordenar las series para la grafica**
- **fc_series = pd.Series(fc, index=index_of_fc)**
- **lower_series = pd.Series(confint[:, 0], index=index_of_fc)**
- **upper_series = pd.Series(confint[:, 1], index=index_of_fc)**

- **# genera la grafica**
- **plt.figure(figsize=(6, 4))**
- **plt.plot(val)**
- **plt.plot(fc_series, color='darkgreen')**
- **plt.fill_between(lower_series.index,**
- **lower_series,**
- **upper_series,**
- **color='k', alpha=.15)**
- **plt.title("Pronostico del indice IPC")**
- **plt.show()**

Una práctica recomendable es validar el comportamiento del modelo antes de utilizarlo en cualquier proceso de toma de decisiones.



Esta pantalla se obtuvo directamente del software que se está explicando en la computadora, para fines educativos.



Para reforzar tu comprensión del tema, realiza lo siguiente:

1. Menciona la forma en que las series de tiempo aportan valiosa información para el análisis técnico en el ambiente financiero de las inversiones.
2. Desarrolla el código Python para resolver los siguientes incisos:
 - a) Descarga los precios históricos de la divisa Euro frente al dólar (EURUSD=X), con frecuencia diaria en el periodo de 3 años a la fecha actual.
 - b) Elabora la gráfica con la serie de tiempo original y dos nuevas series con el promedio móvil de 30 y 50 días respectivamente.
 - c) Elabora la gráfica con la serie original y una nueva serie con el promedio móvil ponderado exponencialmente, considerando $\alpha=0.2$ y ancho de ventana de 30 días.
 - d) Desarrolla el modelo óptimo ARIMA, que incluya la verificación del modelo y el pronóstico de los siguientes 10 días.



El análisis de las series de tiempo ofrece una valiosa metodología para entender mejor el comportamiento de procesos a través del tiempo. Mediante su correcta aplicación es posible hacer que las series de tiempo expongan los secretos de sus fluctuaciones. En el ámbito de las finanzas, forman un complemento del análisis técnico de precios y cotizaciones de la bolsa de valores para obtener importante información que apoya en la toma de decisiones.

Ahora la Sra. Domínguez tiene la herramienta que estaba buscando para hacer sus pronósticos de ventas que le ayudarán a elaborar sus presupuestos y planes de operación de su exitosa fábrica de zapatos, tipo guantes para los pies.





- Hyndman, R., y Athanasopoulos, G. (2021). *Forecasting: principles and practice* (2a ed.). OTexts.

Tecmilenio no guarda relación alguna con las marcas mencionadas como ejemplo. Las marcas son propiedad de sus titulares conforme a la legislación aplicable, estas se utilizan con fines académicos y didácticos, por lo que no existen fines de lucro, relación publicitaria o de patrocinio.

Todos los derechos reservados @ Universidad Tecmilenio

La obra presentada es propiedad de ENSEÑANZA E INVESTIGACIÓN SUPERIOR A.C. (UNIVERSIDAD TECNILENIO), protegida por la Ley Federal de Derecho de Autor; la alteración o deformación de una obra, así como su reproducción, exhibición o ejecución pública sin el consentimiento de su autor y titular de los derechos correspondientes es constitutivo de un delito tipificado en la Ley Federal de Derechos de Autor, así como en las Leyes Internacionales de Derecho de Autor. El uso de imágenes, fragmentos de videos, fragmentos de eventos culturales, programas y demás material que sea objeto de protección de los derechos de autor, es exclusivamente para fines educativos e informativos, y cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por UNIVERSIDAD TECNILENIO. Queda prohibido copiar, reproducir, distribuir, publicar, transmitir, difundir, o en cualquier modo explotar cualquier parte de esta obra sin la autorización previa por escrito de UNIVERSIDAD TECNILENIO. Sin embargo, usted podrá bajar material a su computadora personal para uso exclusivamente personal o educacional y no comercial limitado a una copia por página. No se podrá remover o alterar de la copia ninguna leyenda de Derechos de Autor o la que manifieste la autoría del material.