



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
SECCIONAL TUNJA

VIGILADA MINEDUCACIÓN - SNIES 1732

TEORÍA DE LA PROBABILIDAD TEORÍA DE LA DECISIÓN PYTHON



Teoría de la probabilidad - conceptos

1. ¿Qué es un evento determinista? ¿Qué es un evento aleatorio?
2. ¿Cómo se calcula la probabilidad de fracaso o no ocurrencia de un evento?
3. ¿Qué es la población o universo? ¿Qué es una muestra?
4. ¿Cómo se define la probabilidad de un evento?

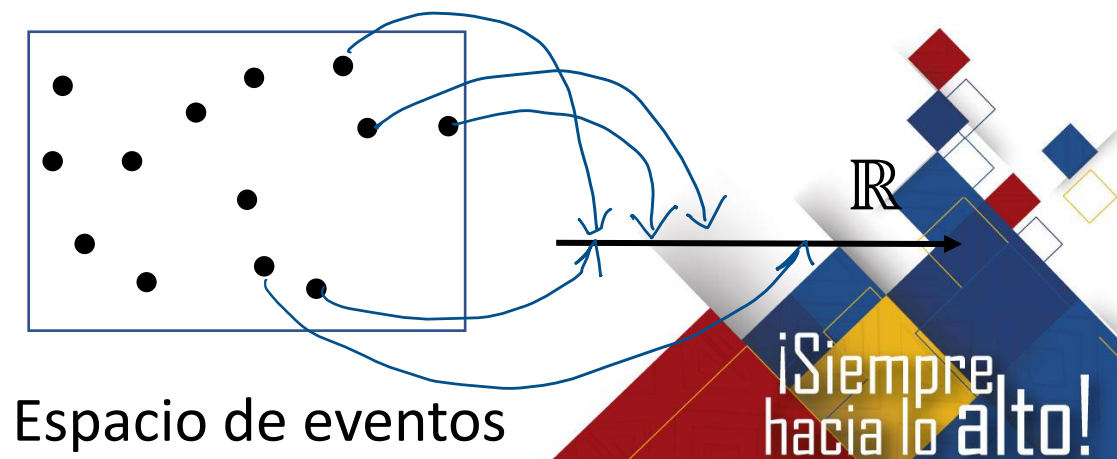
¡Siempre
hacia lo alto!



Teoría de la probabilidad - conceptos

- Elemento aleatorio: es una función determinista definida en un espacio de eventos.
- Asignar el resultado de un experimento aleatorio (experimento que tiene diferentes resultados bajo condiciones experimentales idénticas) a:
 - Un valor real (llamada variable aleatoria).

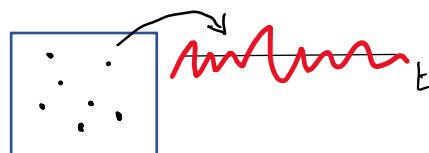
Ejemplo: lanzar dos dados y observar el valor de la suma





Teoría de la probabilidad - conceptos

- Elemento aleatorio: es una función determinista definida en un espacio de eventos.
- Asignar el resultado de un experimento aleatorio (experimento que tiene diferentes resultados bajo condiciones experimentales idénticas) a:
 - Un valor real (llamada variable aleatoria).
 - Un vector de números (vector aleatorio).
 - Un conjunto de números indexado por un espacio continuo o tiempo (proceso estocástico).





Teoría de la probabilidad - conceptos

- Elemento aleatorio: es una función determinista definida en un espacio de eventos.
- Asignar el resultado de un experimento aleatorio (experimento que tiene diferentes resultados bajo condiciones experimentales idénticas) a:
 - Un valor real (llamada variable aleatoria).
 - Un vector de números (vector aleatorio).
 - Un conjunto de números indexado por un espacio continuo o tiempo (proceso estocástico).
 - Un conjunto de números indexados en dos o más dimensiones espaciales discretas (campo aleatorio).



¡Siempre
hacia lo alto!



Teoría de la probabilidad - conceptos

- Los resultados de los experimentos también se pueden representar mediante árboles, gráficas, etc.

¡Siempre
hacia lo alto!



Teoría de la probabilidad - conceptos

- Una variable aleatoria se denota con letra mayúscula, ej: X .
- Los valores concretos se denotan con letra minúscula.



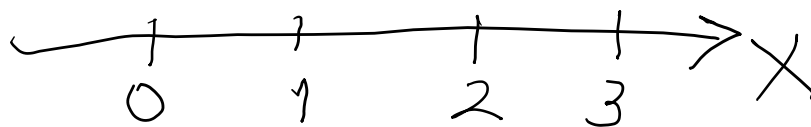
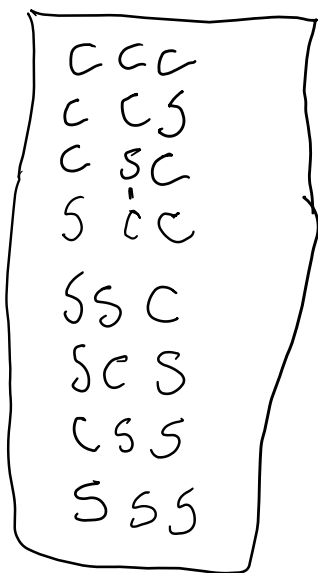
Teoría de la probabilidad - conceptos

- Variables aleatorias (RVs) inducen a una distribución de probabilidad.
 - Distribuciones discretas asignan probabilidad positiva a valores diferentes dentro de un rango.
Ejemplo: número de caras al lanzar tres monedas.



Teoría de la probabilidad - conceptos

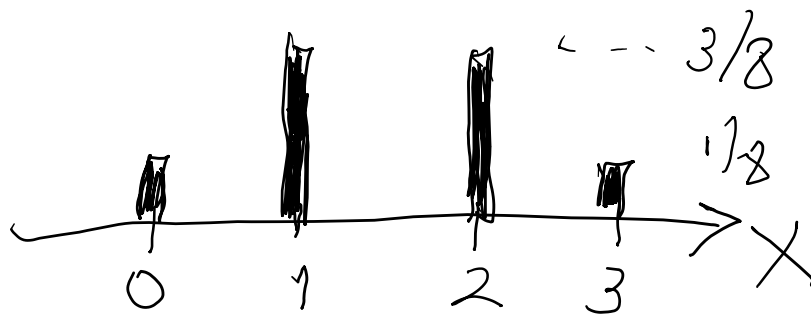
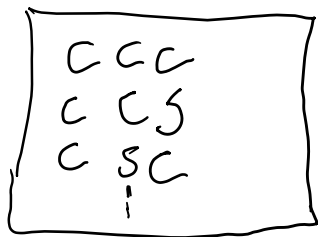
- Variables aleatorias (RVs) inducen a una distribución de probabilidad.
 - Distribuciones discretas asignan probabilidad positiva a valores diferentes dentro de un rango.
Ejemplo: número de caras al lanzar tres monedas.





Teoría de la probabilidad - conceptos

- Variables aleatorias (RVs) inducen a una distribución de probabilidad.
 - Distribuciones discretas asignan probabilidad positiva a valores diferentes dentro de un rango.
Ejemplo: número de caras al lanzar tres monedas.

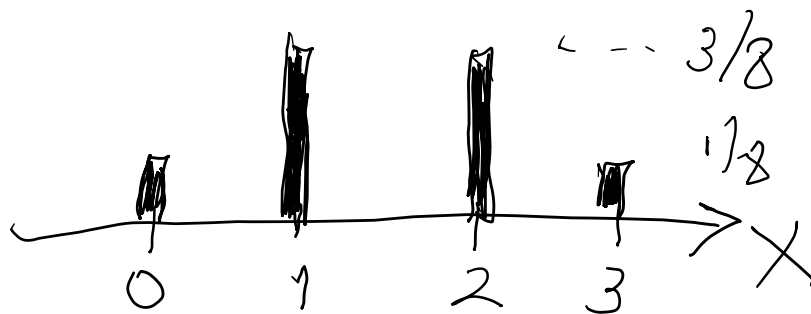




Teoría de la probabilidad - conceptos

- Variables aleatorias (RVs) inducen a una distribución de probabilidad.
 - Distribuciones discretas asignan probabilidad positiva a valores diferentes dentro de un rango.
Ejemplo: número de caras al lanzar tres monedas.

CCC
CCS
CSC
SCC

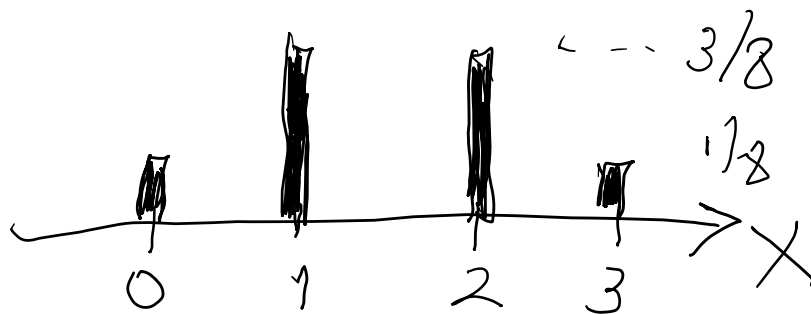
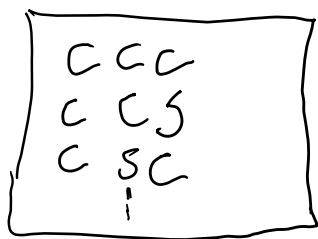


$$P(X=2) = P(\{CCS, CSC, SCC\}) = 3/8$$



Teoría de la probabilidad - conceptos

- Variables aleatorias (RVs) inducen a una distribución de probabilidad.
 - Distribuciones discretas asignan probabilidad positiva a valores diferentes dentro de un rango.
Ejemplo: número de caras al lanzar tres monedas.



$$P(X=2) = P(\{CCS, CSC, SCC\}) = 3/8$$

$$X \sim \text{Binomial}(3, 1/2)$$



Teoría de la probabilidad - conceptos

- Distribuciones continuas son caracterizadas por una densidad de probabilidad

$$P(X \in A) = \int_A p(x) dx$$



Teoría de la probabilidad - conceptos

- Distribuciones continuas son caracterizadas por una densidad de probabilidad

$$P(X \in A) = \int_A p(x) dx$$

$$P(a \leq X \leq b) = \int_a^b p(x) dx$$

$$\int_{-\infty}^{\infty} p(x) dx = 1$$



Teoría de la probabilidad - conceptos

- Esperanza: se usa para caracterizar la ubicación de una distribución.

Discreta:
$$E_x(x) = \sum_{x_i \in \text{Rango}} x_i P(\{X = x\}) = \sum_{x_i} x_i p(x_i)$$

Continua:
$$E(x) = \int x p(x) dx$$



Teoría de la probabilidad - conceptos

- Varianza: caracteriza la extensión de una distribución.

$$\text{Var}(X) = E((X - E(X))^2)$$

$p(x_i)$

Desviación estándar:

$$\sigma(X) = \text{Var}(X)^{1/2}$$



Teoría de la decisión

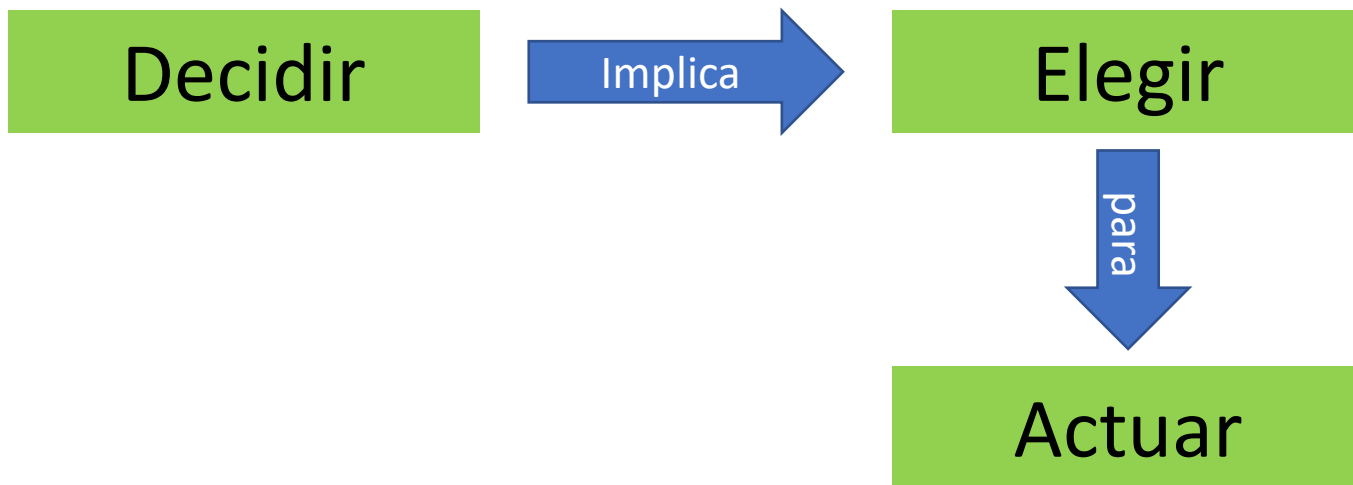
- Se combina con la teoría de la probabilidad para la toma de decisiones óptimas en situaciones donde se involucra la incertidumbre.
- ¿Qué es decidir?

¡Siempre
hacia lo alto!



Teoría de la decisión

- Se combina con la teoría de la probabilidad para la toma de decisiones óptimas en situaciones donde se involucra la incertidumbre.
- ¿Qué es decidir?



¡Siempre
hacia lo alto!



Teoría de la decisión

Elementos de la decisión:

- Estados: situaciones en las que se encuentra un elemento. Puede ser incierto.
- Acciones u opciones: alternativas ante la elección.
- Resultado o rendimiento: evaluación de las consecuencias al elegir (ganancias o pérdidas).

	Estado 1	Estado 2	Estado n
Acción 1	Resultado_11	Resultado_12	Resultado_1n
Acción 2	Resultado_21	Resultado_22	Resultado_2n
Acción n	Resultado_n1	Resultado_n2	Resultado_nn

¡Siempre
hacia lo alto!



Teoría de la decisión

Elementos de la decisión - Ejemplo:

Un fabricante planea introducir un nuevo teléfono móvil. Puede elegir entre cuatro procesos de producción, A, B, C Y D, que van desde una modificación relativamente pequeña de las instalaciones existentes, hasta una gran ampliación de la planta. La decisión sobre el curso de acción debe tomarse en un momento en el que no se conoce la demanda posible del producto. Por comodidad, decimos que esta demanda potencial puede ser «baja», «moderada» o «alta». También se supone que el fabricante puede calcular para cada proceso de producción el beneficio durante la vida de la inversión correspondiente a cada uno de los tres niveles de demanda. La Tabla muestra estos niveles de beneficios (en dólares) para cada combinación “proceso de producción -nivel de demanda. Averigüe si hay alguna acción inadmisibles.

Tabla 21.2. Beneficios estimados de un fabricante de teléfonos móviles correspondientes a diferentes combinaciones de proceso-demanda.

Acción	Estado de la naturaleza		
Proceso de producción	Demanda baja	Demanda moderada	Demanda alta
A	70.000	120.000	200.000
B	80.000	120.000	180.000
C	100.000	125.000	160.000
D	100.000	120.000	150.000

Tomado de:

<http://biblio3.url.edu.gt/Libros/2012/esta-AE/21.pdf>



Teoría de la decisión

Elementos de la decisión - Ejemplo:

Un fabricante planea introducir un nuevo teléfono móvil. Puede elegir entre cuatro procesos de producción, A, B, C Y D, que van desde una modificación relativamente pequeña de las instalaciones existentes, hasta una gran ampliación de la planta. La decisión sobre el curso de acción debe tomarse en un momento en el que no se conoce la demanda posible del producto. Por comodidad, decimos que esta demanda potencial puede ser «baja», «moderada» o «alta». También se supone que el fabricante puede calcular para cada proceso de producción el beneficio durante la vida de la inversión correspondiente a cada uno de los tres niveles de demanda. La Tabla muestra estos niveles de beneficios (en dólares) para cada combinación “proceso de producción -nivel de demanda. Averigüe si hay alguna acción inadmisibles.

Tabla 21.2. Beneficios estimados de un fabricante de teléfonos móviles correspondientes a diferentes combinaciones de proceso-demanda.

Acción	Estado de la naturaleza		
Proceso de producción	Demanda baja	Demanda moderada	Demanda alta
A	70.000	120.000	200.000
B	80.000	120.000	180.000
C	100.000	125.000	160.000
D	100.000	120.000	150.000

Tomado de:

<http://biblio3.url.edu.gt/Libros/2012/esta-AE/21.pdf>



Teoría de la decisión

Elementos de la decisión - Ejemplo:

Un fabricante planea introducir un nuevo teléfono móvil. Puede elegir entre cuatro procesos de producción, A, B, C Y D, que van desde una modificación relativamente pequeña de las instalaciones existentes, hasta una gran ampliación de la planta. La decisión sobre el curso de acción debe tomarse en un momento en el que no se conoce la demanda posible del producto. Por comodidad, decimos que esta demanda potencial puede ser «baja», «moderada» o «alta». También se supone que el fabricante puede calcular para cada proceso de producción el beneficio durante la vida de la inversión correspondiente a cada uno de los tres niveles de demanda. La Tabla muestra estos niveles de beneficios (en dólares) para cada combinación “proceso de producción -nivel de demanda. Averigüe si hay alguna acción inadmisibles.

C domina a D → D es inadmisibles

Tomado de:

<http://biblio3.url.edu.gt/Libros/2012/esta-AE/21.pdf>

Tabla 21.2. Beneficios estimados de un fabricante de teléfonos móviles correspondientes a diferentes combinaciones de proceso-demanda.

Acción	Estado de la naturaleza		
Proceso de producción	Demanda baja	Demanda moderada	Demanda alta
A	70.000	120.000	200.000
B	80.000	120.000	180.000
C	100.000	125.000	160.000
D	100.000	120.000	150.000



Teoría de la decisión

Un fabricante de desodorantes está a punto de ampliar la capacidad de producción para fabricar un nuevo producto. Tiene cuatro procesos de producción alternativos. La tabla adjunta muestra los beneficios estimados, en dólares, de estos procesos correspondientes a tres niveles de demanda del producto posibles.

Acción	Estado de la naturaleza		
Proceso de producción	Demanda baja	Demanda moderada	Demanda alta
A	100.000	350.000	900.000
B	150.000	400.000	700.000
C	250.000	400.000	600.000
D	250.000	400.000	550.000

Tomado de:

<http://biblio3.url.edu.gt/Libros/2012/esta-AE/21.pdf>

¿Es inadmisibile alguna de estas acciones?

¡Siempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones

- **Criterio Maximin:** selecciona la acción que tiene el mayor dentro de los rendimientos mínimos (se maximiza el rendimiento mínimo)

¡Siempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones

- **Criterio Maximin:** selecciona la acción que tiene el mayor dentro de los rendimientos mínimos (se maximiza el rendimiento mínimo)

Tabla 21.2. Beneficios estimados de un fabricante de teléfonos móviles correspondientes a diferentes combinaciones de proceso-demanda.

Acción		Estado de la naturaleza		
Proceso de producción		Demanda baja	Demanda moderada	Demanda alta
A		70.000	120.000	200.000
B		80.000	120.000	180.000
C		100.000	125.000	160.000

¡Siempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones

- **Criterio Maximin:** selecciona la acción que tiene el mayor dentro de los rendimientos mínimos (se maximiza el rendimiento mínimo)

Tabla 21.2. Beneficios estimados de un fabricante de teléfonos móviles correspondientes a diferentes combinaciones de proceso-demanda.

Acción		Estado de la naturaleza		
Proceso de producción	Demanda baja	Demanda moderada	Demanda alta	
A	70.000	120.000	200.000	
B	80.000	120.000	180.000	
C	100.000	125.000	160.000	

¡Siempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones

- **Criterio Maximin:** selecciona la acción que tiene el mayor dentro de los rendimientos mínimos (se maximiza el rendimiento mínimo)

Tabla 21.2. Beneficios estimados de un fabricante de teléfonos móviles correspondientes a diferentes combinaciones de proceso-demanda.

Acción	Estado de la naturaleza		
Proceso de producción	Demanda baja	Demanda moderada	Demanda alta
A	70.000	120.000	200.000
B	80.000	120.000	180.000
C	100.000	125.000	160.000

El criterio maximin selecciona el proceso de producción C.

¡Siempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones: Ejemplo 2

Un inversor quiere elegir entre invertir 10.000 \$ durante un año a un tipo de interés garantizado del 12% e invertir la misma cantidad durante ese periodo en una cartera de acciones ordinarias. Si elige el tipo de interés fijo, tendrá con seguridad un rendimiento de 1.200 \$. Si elige la cartera de acciones, el rendimiento dependerá del comportamiento del mercado durante el año. Si el mercado está boyante, se espera un beneficio de 2.500 \$; si el mercado se mantiene estable, el beneficio esperado es de 500 \$; Y si está deprimido, se espera una pérdida de 1.000 \$. Elabore la tabla de rendimientos de este inversor y halle la elección de la acción mediante el criterio maximin

¡Siempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones: Ejemplo 2

Un inversor quiere elegir entre invertir 10.000 \$ durante un año a un tipo de interés garantizado del 12% e invertir la misma cantidad durante ese periodo en una cartera de acciones ordinarias. Si elige el tipo de interés fijo, tendrá con seguridad un rendimiento de 1.200 \$. Si elige la cartera de acciones, el rendimiento dependerá del comportamiento del mercado durante el año. Si el mercado está boyante, se espera un beneficio de 2.500 \$; si el mercado se mantiene estable, el beneficio esperado es de 500 \$; Y si está deprimido, se espera una pérdida de 1.000 \$. Elabore la tabla de rendimientos de este inversor y halle la elección de la acción mediante el criterio maximin

Acción / Mercado	Boyante	Estable	Deprimido
Interés fijo del 12%	\$1200	\$1200	\$1200
Acciones ordinarias	\$2500	\$500	-\$1000

iSiempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones: Ejemplo 2

Un inversor quiere elegir entre invertir 10.000 \$ durante un año a un tipo de interés garantizado del 12% e invertir la misma cantidad durante ese periodo en una cartera de acciones ordinarias. Si elige el tipo de interés fijo, tendrá con seguridad un rendimiento de 1.200 \$. Si elige la cartera de acciones, el rendimiento dependerá del comportamiento del mercado durante el año. Si el mercado está boyante, se espera un beneficio de 2.500 \$; si el mercado se mantiene estable, el beneficio esperado es de 500 \$; Y si está deprimido, se espera una pérdida de 1.000 \$. Elabore la tabla de rendimientos de este inversor y halle la elección de la acción mediante el criterio maximin

Acción / Mercado	Boyante	Estable	Deprimido
Interés fijo del 12%	\$1200	\$1200	\$1200
Acciones ordinarias	\$2500	\$500	-\$1000

iSiempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones

- **Criterio Minimax:** también llamado criterio de la pérdida de oportunidades minimax. Los pasos a seguir son:
 1. Se resta cada rendimiento de la tabla, del rendimiento mayor de su columna (esto produce la tabla de pérdida de oportunidades).
 2. Se halla en cada fila la máxima pérdida.
 3. Se elige la acción que tiene el mínimo de esas pérdidas máximas.

¡Siempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones

- **Criterio Minimax:**

1. Se resta cada rendimiento de la tabla, del rendimiento mayor de su columna (esto produce la tabla de pérdida de oportunidades).

Tabla 21.2. Beneficios estimados de un fabricante de teléfonos móviles correspondientes a diferentes combinaciones de proceso-demanda.

Acción	Estado de la naturaleza		
Proceso de producción	Demanda baja	Demanda moderada	Demanda alta
A	70.000	120.000	200.000
B	80.000	120.000	180.000
C	100.000	125.000	160.000

¡Siempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones

- **Criterio Minimax:**

1. Se resta cada rendimiento de la tabla, del rendimiento mayor de su columna (esto produce la tabla de pérdida de oportunidades).

Tabla 21.2. Beneficios estimados de un fabricante de teléfonos móviles correspondientes a diferentes combinaciones de proceso-demanda.

Acción		Estado de la naturaleza		
Proceso de producción		Demanda baja	Demanda moderada	Demanda alta
A		70.000	120.000	200.000
B		80.000	120.000	180.000
C		100.000	125.000	160.000

Acción / Demanda	Baja	Moderada	Alta
A	30000	5000	0
B	20000	5000	20000
C	0	0	40000

iSiempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones

- **Criterio Minimax:**
 2. Se halla en cada fila la máxima pérdida.

Tabla 21.2. Beneficios estimados de un fabricante de teléfonos móviles correspondientes a diferentes combinaciones de proceso-demanda.

Acción		Estado de la naturaleza		
Proceso de producción		Demanda baja	Demanda moderada	Demanda alta
A		70.000	120.000	200.000
B		80.000	120.000	180.000
C		100.000	125.000	160.000

Acción / Demanda	Baja	Moderada	Alta
A	30000	5000	0
B	20000	5000	20000
C	0	0	40000

¡Siempre
hacia lo alto!



Teoría de la decisión

Estrategias o métodos para la toma de decisiones

- **Criterio Minimax:**

3. Se elige la acción que tiene el mínimo de esas pérdidas máximas.

Tabla 21.2. Beneficios estimados de un fabricante de teléfonos móviles correspondientes a diferentes combinaciones de proceso-demanda.

Acción	Estado de la naturaleza		
Proceso de producción	Demanda baja	Demanda moderada	Demanda alta
A	70.000	120.000	200.000
B	80.000	120.000	180.000
C	100.000	125.000	160.000

Acción / Demanda	Baja	Moderada	Alta
A	30000	5000	0
B	20000	5000	20000
C	0	0	40000

¡Siempre
hacia lo alto!



Teoría de la decisión

21.3. Considere el ejercicio 21.1, en el que un inversor está considerando tres alternativas —un certificado de depósito, un fondo de acciones de bajo riesgo y un fondo de acciones de alto riesgo— para hacer una inversión de 20.000 \$. Considere tres estados de la naturaleza posibles:

s_1 : mercado de valores fuerte

s_2 : mercado de valores moderado

s_3 : mercado de valores débil

La tabla de rendimientos (en dólares) es la siguiente:

Acción	Estado de la naturaleza		
	s_1	s_2	s_3
Alternativas de inversión posibles			
Certificado de depósito	1.200	1.200	1.200
Fondo de acciones de bajo riesgo	4.300	1.200	-600
Fondo de acciones de alto riesgo	6.600	800	-1.500

- ¿Qué acción se selecciona mediante el criterio maximin?
- ¿Qué acción se selecciona mediante el criterio de la pérdida de oportunidades minimax?

¡Siempre
hacia lo alto!

PYTHON



Uso de Python en ML

Python se utiliza en muchas aplicaciones de ciencia de datos. Otros lenguajes comunes son Matlab y R. Python tiene bibliotecas para carga de datos, visualización, estadísticas, procesamiento de lenguaje natural, procesamiento de imágenes y más.

En este curso usaremos Jupyter Notebook como entorno de programación.

Python también permite la creación de interfaces gráficas de usuario (GUI) y servicios web, y la integración en sistemas existentes.

¡Siempre
hacia lo alto!



Uso de Python en ML – Scikit-learn

- Scikit-learn es un proyecto open source.
- scikit-learn es una herramienta muy popular y la biblioteca de Python más destacada para el aprendizaje automático (Machine Learning).
- scikit-learn depende de otros dos paquetes de Python, NumPy y SciPy. Para el trazado y el desarrollo interactivo, también debe instalar matplotlib, IPython y Jupyter Notebook.

¡Siempre
hacia lo alto!



Instalación de librerías en Jupyter

```
!pip install numpy scipy matplotlib ipython scikit-learn pandas pillow
```

```
Requirement already satisfied: numpy in c:\users\susanac\anaconda3\lib\site-packages (1.16.5)
Requirement already satisfied: scipy in c:\users\susanac\anaconda3\lib\site-packages (1.3.1)
Requirement already satisfied: matplotlib in c:\users\susanac\anaconda3\lib\site-packages (3.1.1)
Requirement already satisfied: ipython in c:\users\susanac\anaconda3\lib\site-packages (7.8.0)
Requirement already satisfied: scikit-learn in c:\users\susanac\anaconda3\lib\site-packages (0.21.3)
Requirement already satisfied: pandas in c:\users\susanac\anaconda3\lib\site-packages (0.25.1)
Requirement already satisfied: pillow in c:\users\susanac\anaconda3\lib\site-packages (6.2.0)
Requirement already satisfied: cyclor>=0.10 in c:\users\susanac\anaconda3\lib\site-packages (from matplotlib) (0.10.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\susanac\anaconda3\lib\site-packages (from matplotlib) (1.1.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in c:\users\susanac\anaconda3\lib\site-packages (from matplotlib) (2.4.2)
```

```
Requirement already satisfied: ipython-genutils in c:\users\susanac\anaconda3\lib\site-packages (from traitlets>=4.2->ipython) (0.2.0)
Requirement already satisfied: wcwidth in c:\users\susanac\anaconda3\lib\site-packages (from prompt-toolkit<2.1.0,>=2.0.0->ipython) (0.1.7)
Requirement already satisfied: parso>=0.5.0 in c:\users\susanac\anaconda3\lib\site-packages (from jedi>=0.10->ipython) (0.5.1)
```

Note: you may need to restart the kernel to use updated packages.

Siempre
hacia lo alto!



Uso de Python en ML – NumPy

- NumPy contiene funcionalidad para matrices multidimensionales, funciones matemáticas de alto nivel como operaciones de álgebra lineal y la transformada de Fourier, y generadores de números pseudoaleatorios.
- Cualquier dato que esté utilizando scikit-learn deberá convertirse en una matriz NumPy. La funcionalidad principal de NumPy es la clase ndarray, una matriz multidimensional (n-dimensional). Todos los elementos de la matriz deben ser del mismo tipo. Una matriz NumPy se ve así:

```
► import numpy as np  
x = np.array([[1, 2, 3], [4, 5, 6]])  
print("x:\n{}".format(x))
```

```
x:  
[[1 2 3]  
 [4 5 6]]
```

¡Siempre
hacia lo alto!



Uso de Python en ML – SciPy

- Proporciona rutinas avanzadas de álgebra lineal, optimización de funciones matemáticas, procesamiento de señales, funciones matemáticas especiales y distribuciones estadísticas.
- scikit-learn se basa en la colección de funciones de SciPy para implementar sus algoritmos. La parte más importante de SciPy para nosotros es `scipy.sparse`: esto proporciona matrices dispersas, que son otra representación que se usa para los datos en scikitlearn.
- Las matrices dispersas se utilizan siempre que queremos almacenar una matriz 2D que contiene principalmente ceros:



Uso de Python en ML – SciPy

```
from scipy import sparse  
eye = np.eye(4)  
print("NumPy array:\n{}".format(eye))
```

NumPy array:
[[1. 0. 0. 0.]
 [0. 1. 0. 0.]
 [0. 0. 1. 0.]
 [0. 0. 0. 1.]]

```
sparse_matrix = sparse.csr_matrix(eye)  
print("\nSciPy sparse CSR matrix:\n{}".format(sparse_matrix))
```

SciPy sparse CSR matrix:
(0, 0) 1.0
(1, 1) 1.0
(2, 2) 1.0
(3, 3) 1.0



Uso de Python en ML – Matplotlib

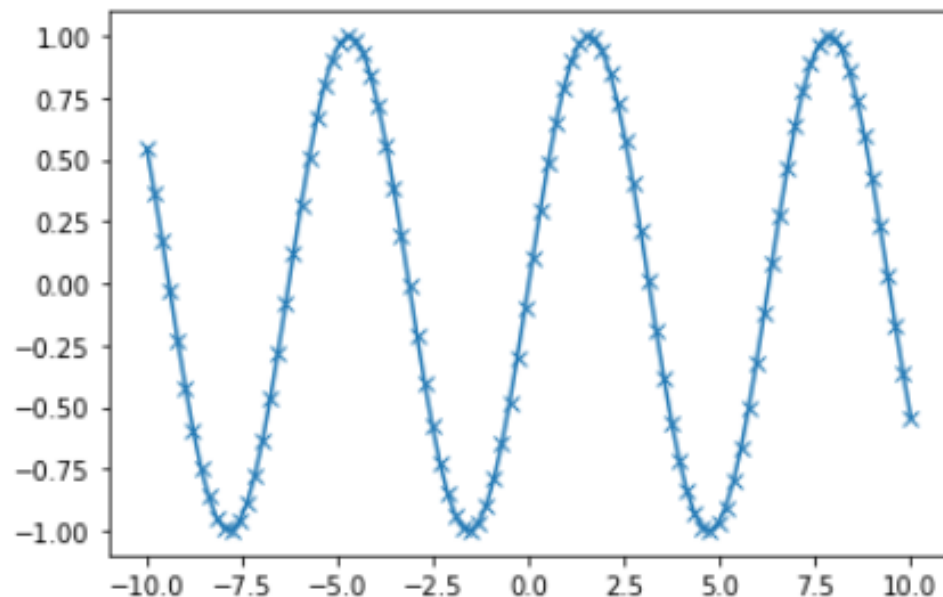
- matplotlib proporciona funciones para realizar visualizaciones con calidad de publicación, como gráficos de líneas, histogramas, diagramas de dispersión, etc.
- Visualizar sus datos y diferentes aspectos de su análisis puede brindarle información importante, y usaremos matplotlib para todas nuestras visualizaciones.



Uso de Python en ML – Matplotlib

```
In [24]: ▶ %matplotlib inline
import matplotlib.pyplot as plt
# Generate a sequence of numbers from -10 to 10 with 100 steps in between
x = np.linspace(-10, 10, 100)
# Create a second array using sin
y = np.sin(x)
# The plot function makes a line chart of one array against another
plt.plot(x, y, marker="x")
```

Out[24]: [<matplotlib.lines.Line2D at 0x24f94b46e08>]



¡Siempre
hacia lo alto!



Uso de Python en ML – Pandas

- pandas es una biblioteca de Python para la gestión y el análisis de datos. Está construido alrededor de una estructura de datos llamada DataFrame. En pocas palabras, un DataFrame de pandas es una tabla, similar a una hoja de cálculo de Excel.
- pandas proporciona una gran variedad de métodos para modificar y operar en esta tabla; en particular, permite consultas de tipo SQL y uniones de tablas. A diferencia de NumPy, que requiere que todas las entradas en una matriz sean del mismo tipo, pandas permite que cada columna tenga un tipo separado (por ejemplo, números enteros, fechas, números de punto flotante y cadenas).



Uso de Python en ML – Pandas

- Otra herramienta valiosa proporcionada por pandas es su capacidad para incluir datos desde una gran variedad de formatos de archivo y bases de datos, como SQL, archivos de Excel y archivos de valores separados por comas (CSV).

```
import pandas as pd
from IPython.display import display
# create a simple dataset of people
data = {'Name': ["John", "Anna", "Peter", "Linda"],
        'Location': ["New York", "Paris", "Berlin", "London"],
        'Age': [24, 13, 53, 33]}
data_pandas = pd.DataFrame(data)
# IPython.display allows "pretty printing" of dataframes
# in the Jupyter notebook
display(data_pandas)
```

	Name	Location	Age
0	John	New York	24
1	Anna	Paris	13
2	Peter	Berlin	53
3	Linda	London	33



Uso de Python en ML – Pandas

```
import pandas as pd
from IPython.display import display
# create a simple dataset of people
data = {'Name': ["John", "Anna", "Peter", "Linda"],
        'Location': ["New York", "Paris", "Berlin", "London"],
        'Age': [24, 13, 53, 33]}
data_pandas = pd.DataFrame(data)
# IPython.display allows "pretty printing" of dataframes
# in the Jupyter notebook
display(data_pandas)
```

	Name	Location	Age
0	John	New York	24
1	Anna	Paris	13
2	Peter	Berlin	53
3	Linda	London	33

```
# Select all rows that have an age column greater than 30
display(data_pandas[data_pandas.Age > 30])
```

	Name	Location	Age
2	Peter	Berlin	53
3	Linda	London	33



Una primera aplicación: clasificación de especies de iris

Supongamos que un botánico está interesado en distinguir las especies de algunas flores de iris que ha encontrado. Ha recopilado algunas medidas asociadas con cada iris: el largo y ancho de los pétalos y el largo y ancho de los sépalos, todos medidos en centímetros.

También tiene las medidas de algunos lirios que han sido previamente identificados por un botánico experto como pertenecientes a las especies setosa, versicolor o virginica. Para estas medidas, puede estar segura de a qué especie pertenece cada iris. Supongamos que estas son las únicas especies que nuestro botánico encontrará en la naturaleza. Nuestro objetivo es construir un modelo de aprendizaje automático que pueda aprender de las medidas de estos lirios cuya especie se conoce, de modo que podamos predecir la especie para un nuevo iris.



Imagen de la flor tomada de: https://estag.fimágenes.com/img/1/X/m/h/Xmh_900.jpg

¡Siempre
hacia lo alto!



Una primera aplicación: clasificación de especies de iris

Este es un ejemplo de un problema de clasificación. Las posibles salidas (diferentes especies de iris) se denominan clases. Cada iris del conjunto de datos pertenece a una de tres clases, por lo que este problema es un problema de clasificación de tres clases. La salida deseada para un solo punto de datos (un iris) es la especie de esta flor. Para un punto de datos en particular, la especie a la que pertenece se llama etiqueta.

Los datos que usaremos para este ejemplo son el conjunto de datos Iris, un conjunto de datos clásico en aprendizaje automático y estadísticas. Está incluido en scikit-learn en el módulo de conjuntos de datos. Podemos cargarlo llamando a la función `load_iris`:



Una primera aplicación: clasificación de especies de iris

```
#Ejemplo de flores de iris
```

```
from sklearn.datasets import load_iris
iris_dataset = load_iris()
print("Keys of iris_dataset: \n{}".format(iris_dataset.keys()))
```

```
Keys of iris_dataset:
dict_keys(['data', 'target', 'target_names', 'DESCR', 'feature_names', 'filename'])
```

```
| print(iris_dataset['DESCR'][:230] + "\n...")
```

```
.. _iris_dataset:
```

```
Iris plants dataset
```

```
-----
```

```
**Data Set Characteristics:**
```

```
 :Number of Instances: 150 (50 in each of three classes)
```

```
 :Number of Attributes: 4 numeric, predictive attributes and the class
```

```
 ...
```

¡Siempre
hacia lo alto!



Una primera aplicación: clasificación de especies de iris

```
▶ print("Target names: {}".format(iris_dataset['target_names']))
```

```
Target names: ['setosa' 'versicolor' 'virginica']
```

```
print("Feature names: \n{}".format(iris_dataset['feature_names']))  
print("Type of data: {}".format(type(iris_dataset['data'])))  
print("Shape of data: {}".format(iris_dataset['data'].shape))
```

```
Feature names:
```

```
['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)', 'petal width (cm)']
```

```
Type of data: <class 'numpy.ndarray'>
```

```
Shape of data: (150, 4)
```



Una primera aplicación: clasificación de especies de iris

```
print("First five rows of data:\n{}".format(iris_dataset['data'][:5]))
```

First five rows of data:

```
[[5.1 3.5 1.4 0.2]
 [4.9 3.  1.4 0.2]
 [4.7 3.2 1.3 0.2]
 [4.6 3.1 1.5 0.2]
 [5.  3.6 1.4 0.2]]
```

¡Siempre
hacia lo alto!



Una primera aplicación: clasificación de especies de iris

Para evaluar el rendimiento del modelo, le mostramos nuevos datos (datos que no ha visto antes) para los que tenemos etiquetas. Esto generalmente se hace dividiendo los datos etiquetados que hemos recopilado (aquí, nuestras 150 medidas de flores) en dos partes. Una parte de los datos se utiliza para construir nuestro modelo de aprendizaje automático y se denomina datos de entrenamiento o conjunto de entrenamiento. El resto de los datos se utilizará para evaluar qué tan bien funciona el modelo; esto se denomina datos de prueba, conjunto de prueba o conjunto de reserva.

scikit-learn contiene una función que mezcla el conjunto de datos y lo divide por usted: la función `train_test_split`. Esta función extrae el 75% de las filas de los datos como conjunto de entrenamiento, junto con las etiquetas correspondientes para estos datos. El 25% restante de los datos, junto con las etiquetas restantes, se declara como conjunto de prueba. Decidir cuántos datos desea poner en el entrenamiento y el conjunto de prueba respectivamente es algo arbitrario, pero usar un conjunto de prueba que contenga el 25% de los datos es una buena regla general.

¡Siempre
hacia lo alto!



Una primera aplicación: clasificación de especies de iris

En scikit-learn, los datos generalmente se indican con una X mayúscula, mientras que las etiquetas se indican con una y minúscula. Esto está inspirado en la formulación estándar $f(x) = y$ en matemáticas, donde x es la entrada a una función e y es la salida. Siguiendo más convenciones de las matemáticas, usamos una X mayúscula porque los datos son una matriz bidimensional (una matriz) y una y minúscula porque el objetivo es una matriz unidimensional (un vector). Llamemos a `train_test_split` en nuestros datos y asignemos las salidas usando esta nomenclatura:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    iris_dataset['data'], iris_dataset['target'], random_state=0)
```



Una primera aplicación: clasificación de especies de iris

Antes de realizar la división, la función `train_test_split` baraja el conjunto de datos utilizando un generador de números pseudoaleatorios. Si tomáramos el último 25% de los datos como un conjunto de prueba, todos los puntos de datos tendrían la etiqueta 2, ya que los puntos de datos están ordenados por la etiqueta (consulte la salida para iris ['objetivo'] que se muestra anteriormente). Debemos mezclar nuestros datos para asegurarnos que los datos de prueba contengan datos de todas las clases. Para asegurarnos de que obtendremos el mismo resultado si ejecutamos la misma función varias veces, proporcionamos al generador de números pseudoaleatorios una semilla fija usando el parámetro `random_state`. Esto hará que el resultado sea determinista, por lo que esta línea siempre tendrá el mismo resultado.



Una primera aplicación: clasificación de especies de iris

Una de las mejores formas de inspeccionar datos es visualizarlos. Una forma de hacer esto es usando un gráfico de dispersión. Un diagrama de dispersión de los datos coloca una característica a lo largo del eje x y otra a lo largo del eje y, y dibuja un punto para cada punto de datos.

Es difícil trazar conjuntos de datos con más de tres características de esta manera. Una forma de solucionar este problema es hacer un diagrama de pares, que observe todos los posibles pares de características. Si tiene una pequeña cantidad de funciones, como las cuatro que tenemos aquí, esto es bastante razonable.

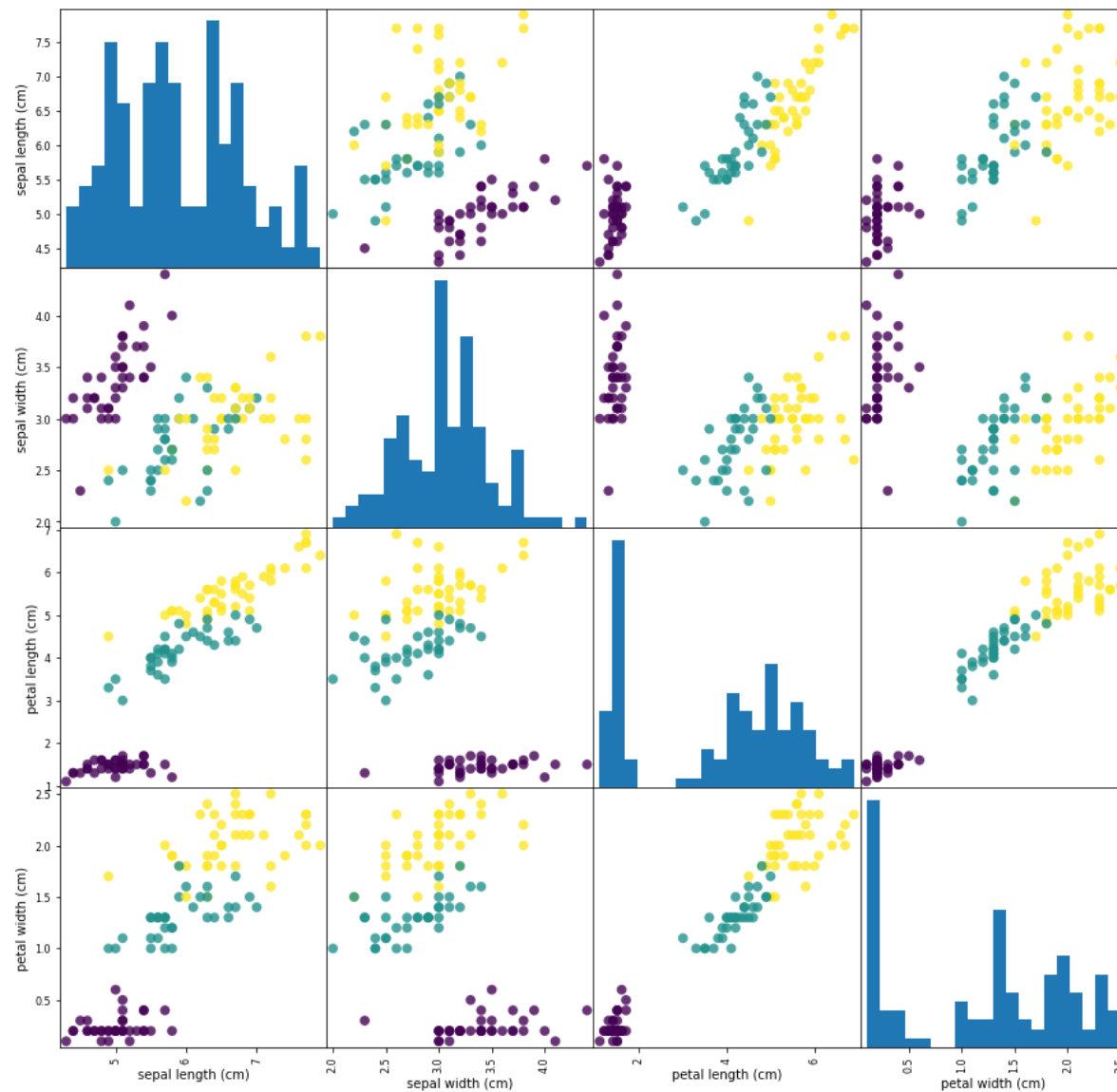


Una primera aplicación: clasificación de especies de iris

```
| # create dataframe from data in X_train  
# label the columns using the strings in iris_dataset.feature_names  
iris_dataframe = pd.DataFrame(X_train, columns=iris_dataset.feature_names)  
# create a scatter matrix from the dataframe, color by y_train  
pd.plotting.scatter_matrix(iris_dataframe, c=y_train, figsize=(15, 15),  
marker='o', hist_kws={'bins': 20}, s=60,  
alpha=0.8)|
```



Una primera aplicación: clasificación de especies de iris



¡Siempre
hacia lo alto!



REFERENCIAS BIBLIOGRÁFICAS

Bishop, C. (2006). Pattern recognition and Machine Learning

<http://biblio3.url.edu.gt/Libros/2012/esta-AE/21.pdf>

Muller, A. and Guido, S. (2017). Introduction to Machine Learning with Python

¡Siempre
hacia lo alto!