Introducción a la Ciencia de Datos

Maestría en Ciencias de la Computación

Dr. Irvin Hussein López Nava

50 ANIVERSARIO CICESE

CONAHCYT
CONSEJO NACIONAL DE HUMANIDADES CIENCIAS Y TECNOLOGÍAS

Repositorios de datos

kaggle

Microsoft

Lectura 1: Pasado, presente y futuro de la Ciencia de Datos

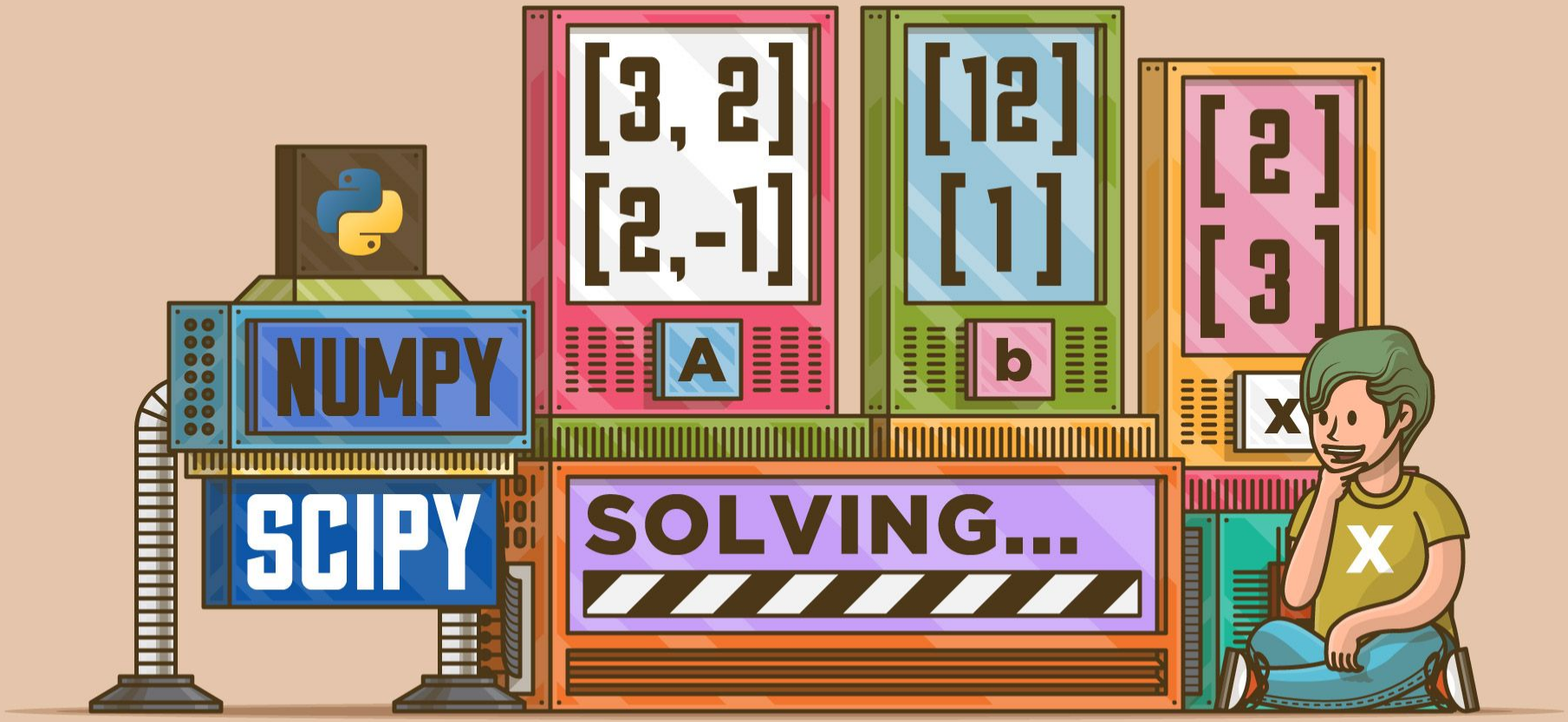
Irvin Hussein Lopez Nava • 28 ago

100 puntos

Preparar un reporte de una cuartilla en el que se analice a detalle el presente del área, partiendo de 3 tópicos abordados en el video adjunto (perspectiva de hace 8 años - pasado). Finalizar el análisis con una proyección a futuro (a 10 años). Complementar con fuentes de información adicionales, y citarlas apropiadamente.

The Future of Data Science -...

Vídeo de YouTube • 25 minutos



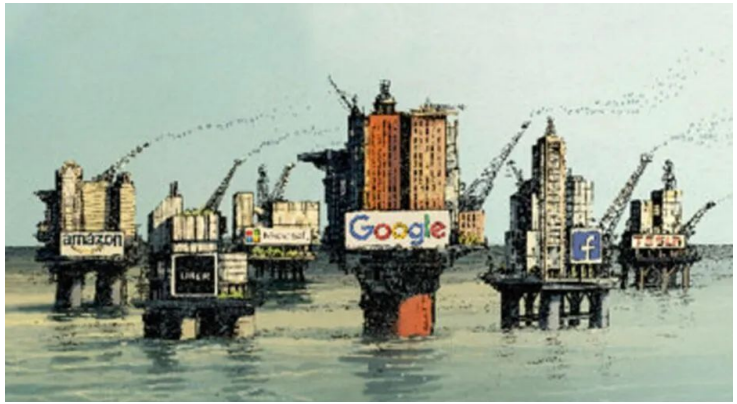
Real Python

1.3 Representación

Let's keep talking (and listening) about data...

Data is the new oil?

Cuando el matemático británico **Clive Humby** lo declaró en 2006, quería decir que los datos, como el petróleo, no son útiles en estado bruto.
Hay que refinarlos, procesarlos y convertirlos en algo útil; su **valor** reside en su **potencial**.

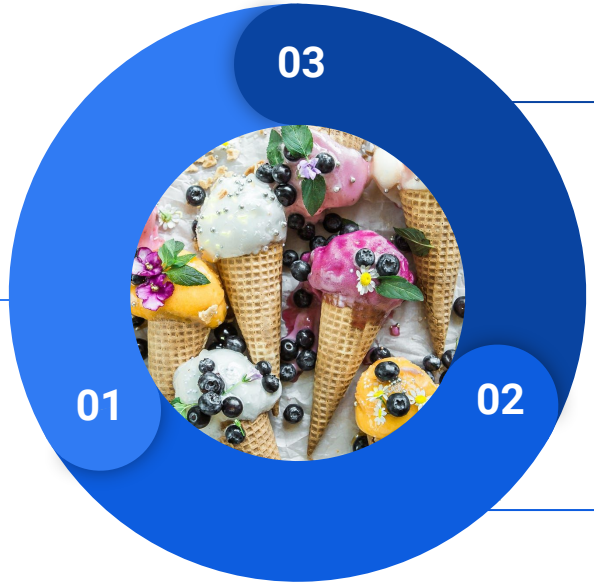


Sin embargo, ha perdido gran parte de su significado: los datos como algo intrínsecamente valioso, algo que simplemente se extrae, se acumula y se usa o se abusa de él.

https://www.thedrum.com/opinion/2022/12/12/data-isn-t-the-new-oil-it-s-way-more-valuable

Data are dessert?

Data are the result of deliberate human intervention



Data are varied within domains

Data are varied across domains

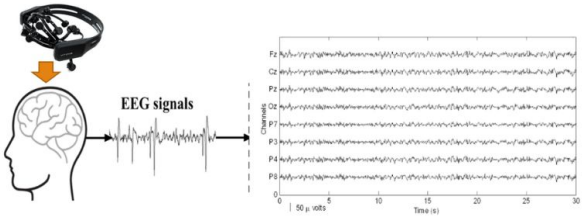
https://ift6758.github.io/lectures.html

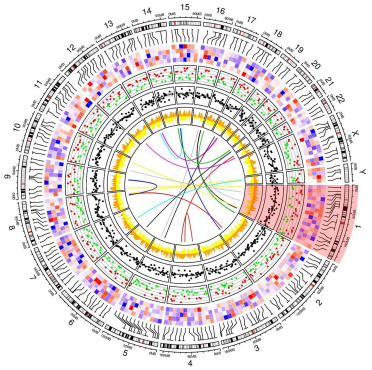
Data are dessert?

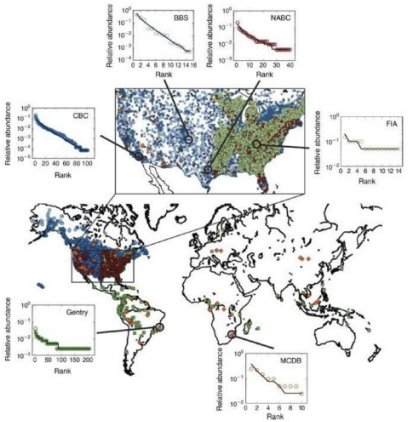
Data are the result of deliberate human intervention

01

Data are dessert?



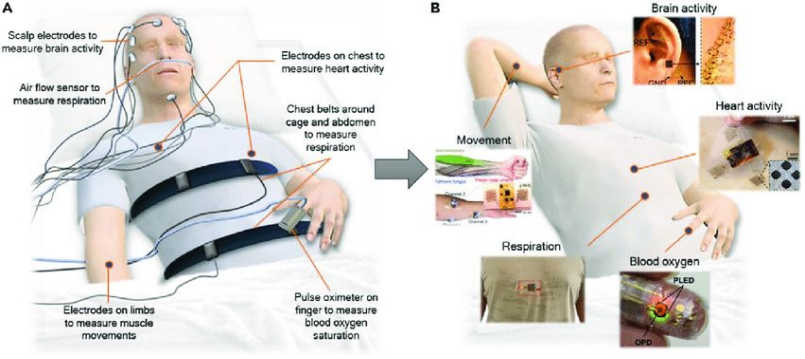




02

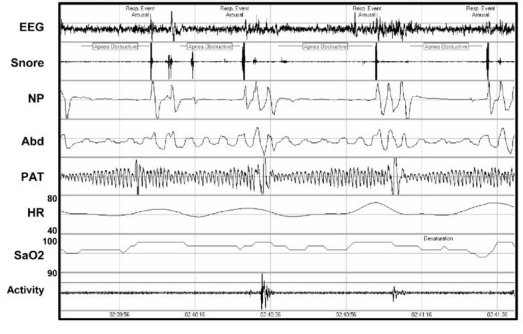
Data are varied across domains

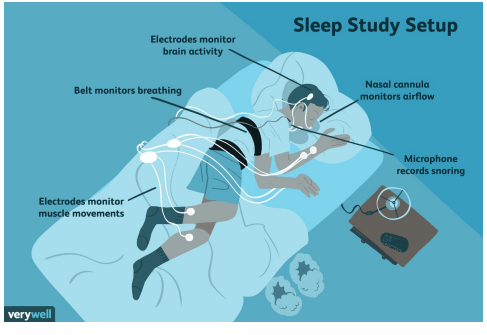
Data are dessert?



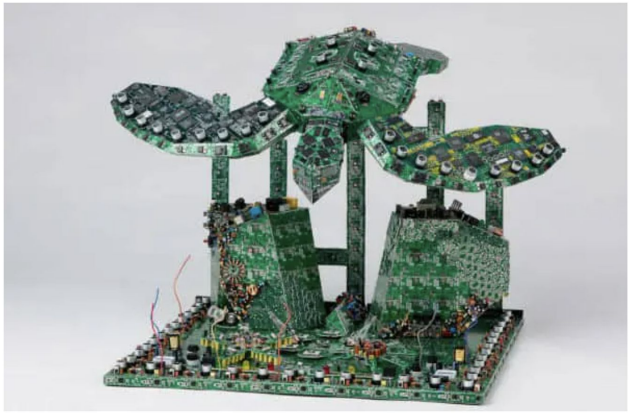
03

Data are varied within domains





What is data?

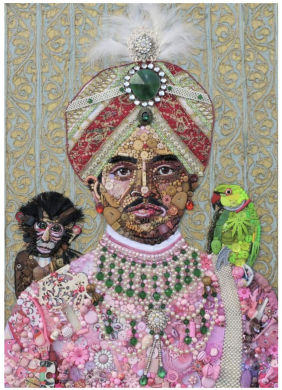


Sea Turtle Searching For Deep Data, 2010 | Credit photo: Copyright © artist Steven Rodrig

Clues, 2019 | Credit photo: Copyright © artist Nick Gentry

Across the universe | Credit photo: Copyright © artist Derek Gores

Credit photo: Copyright © artist Robert Bradford



The Raj's Emerald 2020 | Credit photo: artist Copyright © Jane Perkins

Plastic Ocean, 2016 | Credit photo: Copyright © artist Tan Zi Xi

El objetivo: hacer que valgan!!

Gestión de datos

(era el segundo nombre para esta sesión)

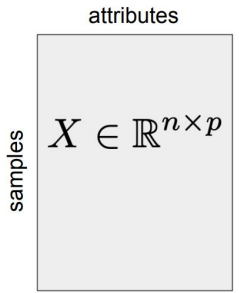
Los datos (+ las personas que los recopilan) son variados.
→ Siempre es necesaria cierta preparación.

Esto es *antes* de ir al análisis de los datos mediante

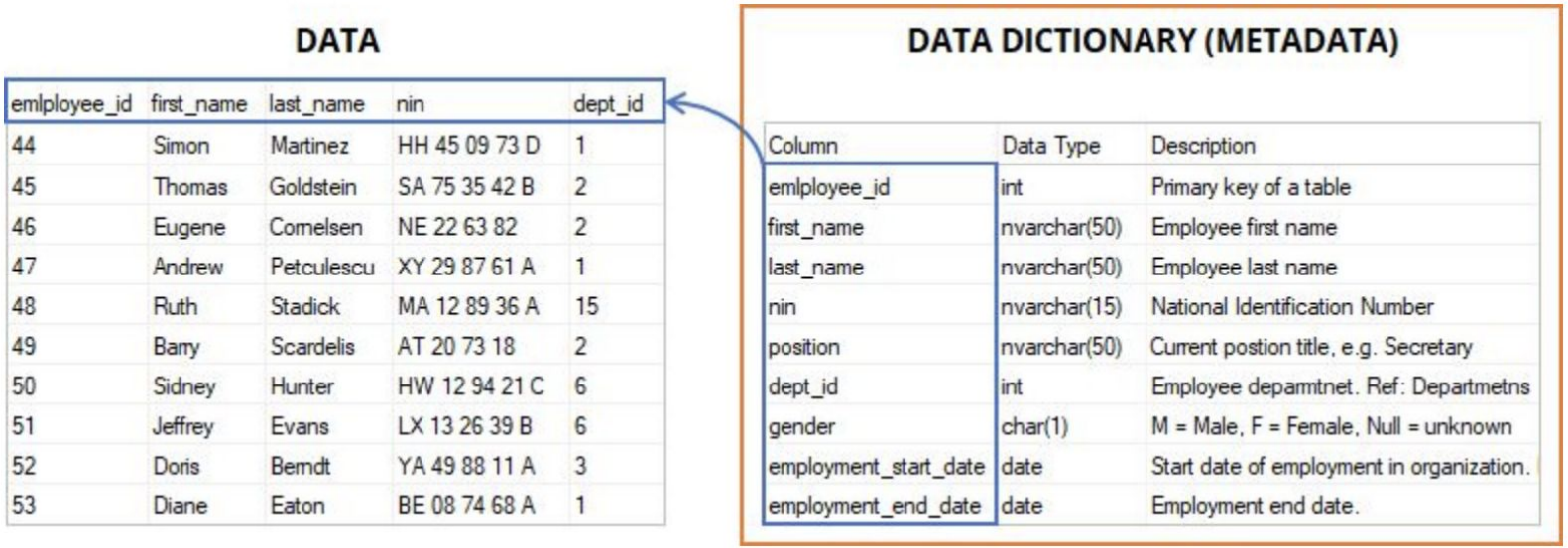
- Entendimiento del contexto
- Estudio del tipo de datos
- Exploración inicial de los datos
- Conversión de los datos
- Limpieza de los datos
- …

Primeros pasos

- Una vez que se tiene un *dataset*, se requiere:
  - Entender qué son las variables.
  - Gestionar los tipos de columnas (atributos).
  - Manejar los datos incompletos.
  - Unir, reorganizar y ordenar.
  - Los datos no siempre llegan como "X".

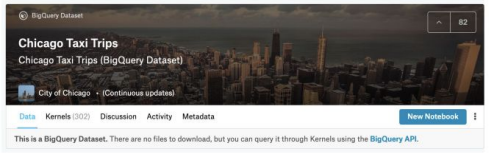


Significado de los datos



Manejando tipos

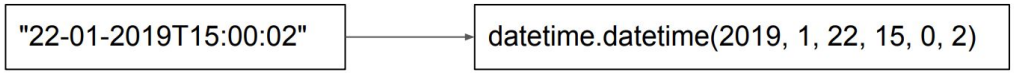
- Los datos vienen con diferentes "tipos.
  - Numéricos, categóricos (ordenados), fechas, enteros (positivos).
- Es valioso asegurar la coherencia respecto a lo que se esperaba. (¿por qué?)




```
In [6]:
taxi.dtypes
Out [6]:
unique_key                 object
taxi_id                    object
trip_start_timestamp       object
trip_end_timestamp         object
trip_seconds              float64
trip_miles                float64
pickup_census_tract       float64
dropoff_census_tract      float64
pickup_community_area     float64
dropoff_community_area    float64
fare                      float64
tips                      float64
tolls                     float64
extras                    float64
trip_total                float64
payment_type               object
company                    object
pickup_latitude           float64
pickup_longitude          float64
pickup_location           float64
dropoff_latitude          float64
dropoff_longitude         float64
dropoff_location          float64
dtype: object
```

Conversión de tipos

Se pueden utilizar funciones para convertir entre tipos de datos, e.g., *to_datatime* de pandas.




```
In [13]:
taxi = taxi.assign(
    day=lambda df: df.trip_start_timestamp.map(lambda x: x.day),
    weekday=lambda df: df.trip_start_timestamp.map(lambda x: x.weekday),
    hour=lambda df: df.trip_start_timestamp.map(lambda x: x.hour)
)
taxi[["day", "weekday"]]
Out [13]:
       day  weekday
0        7        6
1        7        6
2        7        6
3       20        5
4       30        5
...    ...      ...
19995    7        6
19996    9        1
19997    7        6
19998   16        1
19999    5        4

[20000 rows x 2 columns]
```

Tipos categóricos

El número de niveles puede ser abrumador

Una sola categoría puede incorporar diferente información

name
Kris Sankaran

first_name	last_name
Kris	Sankaran

Los niveles podrían no estar consolidados

vegetable
POTATO
carrot
potato

vegetable
potato
carrot
potato

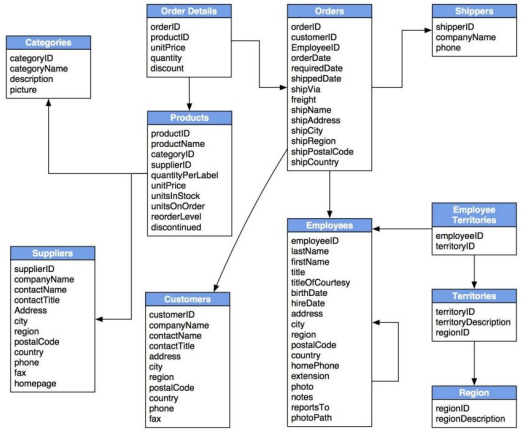
Se podrían convertir en valores numéricos

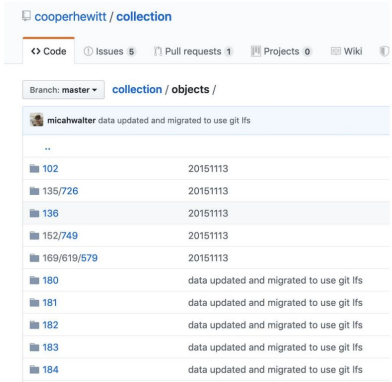
Happy?
yes
yes
no
maybe
no

yes	no	maybe
1	0	0
1	0	0
0	1	0
0	0	1
0	1	0

Agrupar, unir y organizar

Se desea una representación en matriz que facilite el trabajo, pero…






```
,width,height,channels,im_size,ctime,mtime,img_files,img_folders,img_subfolders
0,1300,1300,1,3383838,1515131214.7275624,1511274026.0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img1510.tif
1,1300,1300,1,3383838,1515131210.4355597,1511274012.0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img338.tif
2,1300,1300,1,3383838,1515131211.9995608,1511274005.0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img1468.tif
3,1300,1300,1,3383838,1515131223.5595682,1511274001.0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img857.tif
4,1300,1300,1,3383838,1515131222.8075676,1511274003.0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img86.tif
5,1300,1300,1,3383838,1515131215.503563,1511274006.0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img1276.tif
6,1300,1300,1,3383838,1515131223.479568,1511274002.0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img1582.tif
7,1300,1300,1,3383838,1515131223.0555677,1511274031.0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img127.tif
8,1300,1300,1,3383838,1515131213.975562,1511274031.0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img1550.tif
9,1300,1300,1,3383838,1515131211.9235606,1511274013.0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img1447.tif
10,1300,1300,1,3383838,1515131213.0635614,1511274005,0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img894.tif
11,1300,1300,1,3383838,1515131215.6915631,1511274005,0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img741.tif
12,1300,1300,1,3383838,1515131210.0635595,1511274007,0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img522.tif
13,1300,1300,1,3383838,1515131210.1115596,1511274002,0,AOI_2_Vegas_Roads_Test_Public,PAN,PAN_AOI_2_Vegas_img1151.tif
```

El inconveniente?

La ayuda?

Herramientas

NumPy (**Num**erical **Py**thon) es una **librería** de código abierto de Python que se utiliza en casi todos los campos de la ciencia y la ingeniería.

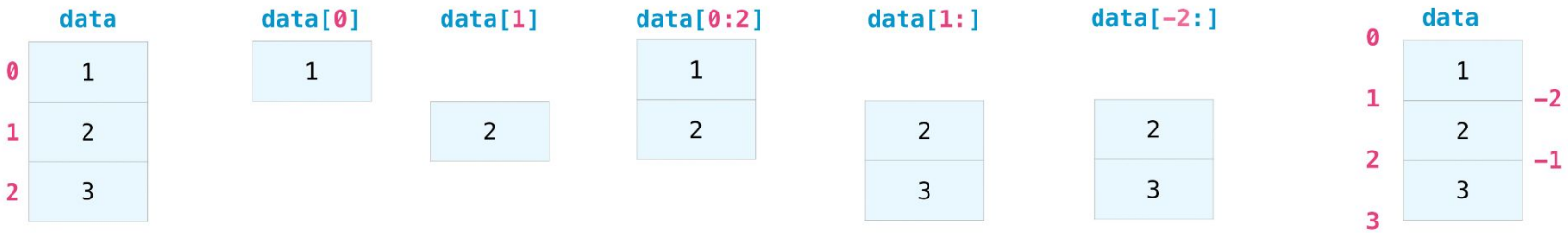
- Es el estándar universal para trabajar con **datos numéricos** en Python.
- La **API** de NumPy se utiliza ampliamente en Pandas, SciPy, Matplotlib, scikit-learn, scikit-image y la mayoría de los paquetes científicos y de ciencia de datos de Python.
- Añade potentes estructuras de datos que garantizan cálculos eficientes con *arrays* y proporciona una enorme biblioteca de funciones matemáticas de alto nivel que operan sobre estos *arrays*.

Operator	Equivalent ufunc	Description
`+`	`np.add`	Addition (e.g., `1 + 1 = 2`)
`-`	`np.subtract`	Subtraction (e.g., `3 - 2 = 1`)
`-`	`np.negative`	Unary negation (e.g., `-2`)
`*`	`np.multiply`	Multiplication (e.g., `2 * 3 = 6`)
`/`	`np.divide`	Division (e.g., `3 / 2 = 1.5`)
`//`	`np.floor_divide`	Floor division (e.g., `3 // 2 = 1`)
`**`	`np.power`	Exponentiation (e.g., `2 ** 3 = 8`)
`%`	`np.mod`	Modulus/remainder (e.g., `9 % 4 = 1`)

Operator	Equivalent ufunc	Operator	Equivalent ufunc
`==`	`np.equal`	`!=`	`np.not_equal`
`<`	`np.less`	`<=`	`np.less_equal`
`>`	`np.greater`	`>=`	`np.greater_equal`

Function name	NaN-safe version	Description
`np.sum`	`np.nansum`	Compute sum of elements
`np.prod`	`np.nanprod`	Compute product of elements
`np.mean`	`np.nanmean`	Compute mean of elements
`np.std`	`np.nanstd`	Compute standard deviation
`np.var`	`np.nanvar`	Compute variance
`np.min`	`np.nanmin`	Find minimum value
`np.max`	`np.nanmax`	Find maximum value
`np.argmin`	`np.nanargmin`	Find index of minimum value
`np.argmax`	`np.nanargmax`	Find index of maximum value
`np.median`	`np.nanmedian`	Compute median of elements
`np.percentile`	`np.nanpercentile`	Compute rank-based statistics of elements
`np.any`	N/A	Evaluate whether any elements are true
`np.all`	N/A	Evaluate whether all elements are true

Lo más básico



https://numpy.org/doc/stable/user/absolute_beginners.html

```
data = np.array([1,2])
```

data

1
2

```
ones = np.ones(2)
```

ones

1
1

```
data + ones =
```

data

1
2

+

ones

1
1

=

2
3

data

1
2

-

ones

1
1

=

0
1

data

1
2

*

data

1
2

=

1
4

data

1
2

/

data

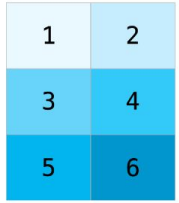
1
2

=

1
1

https://numpy.org/doc/stable/user/absolute_beginners.html

```
np.array([[1,2],[3,4],[5,6]])
```



data

	0	1
0	1	2
1	3	4
2	5	6

data[0,1]

	0	1
0	1	2
1	3	4
2	5	6

data[1:3]

	0	1
0	1	2
1	3	4
2	5	6

data[0:2,0]

	0	1
0	1	2
1	3	4
2	5	6

data

1	2
3	4
5	6

`.max()` = 6

data

1	2
3	4
5	6

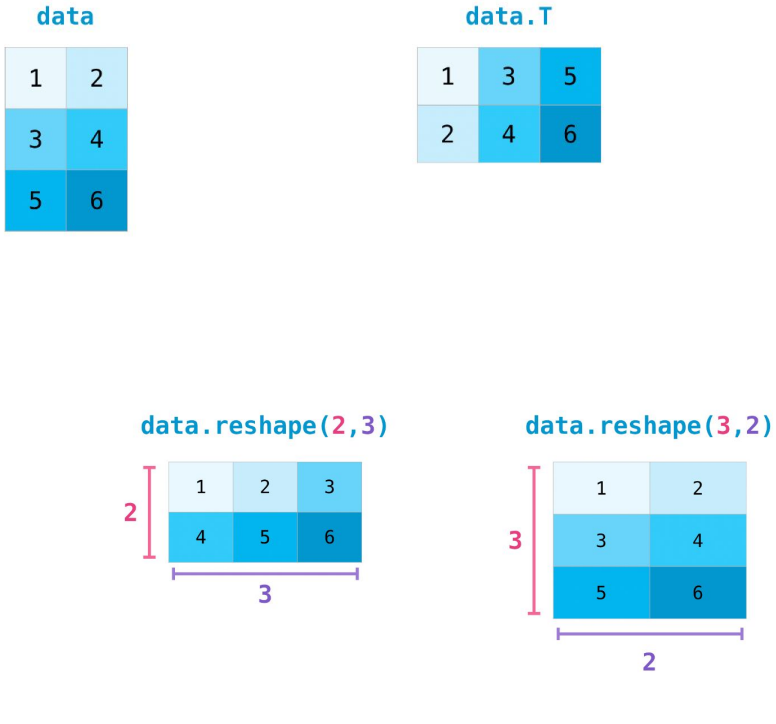
`.min()` = 1

data

1	2
3	4
5	6

`.sum()` = 21

https://numpy.org/doc/stable/user/absolute_beginners.html



https://numpy.org/doc/stable/user/absolute_beginners.html

pandas es una **librería** de código abierto de Python que proporciona estructuras de datos rápidas, flexibles y expresivas.

Es adecuada para muchos tipos de **datos**:

- Datos tabulares con columnas de tipos heterogéneos, como en una tabla SQL o una hoja de cálculo Excel.
- Datos de series temporales ordenados y desordenados.
- Datos matriciales arbitrarios (de tipificación homogénea o heterogénea) con etiquetas de filas y columnas.
- Cualquier otra forma de conjunto de datos observacionales/estadísticos.

Python operator	Pandas method(s)
+	add
-	sub, subtract
*	mul, multiply
/	truediv, div, divide
//	floordiv
%	mod
**	pow

Aggregation	Returns
count	Total number of items
first, last	First and last item
mean, median	Mean and median
min, max	Minimum and maximum
std, var	Standard deviation and variance
mad	Mean absolute deviation
prod	Product of all items
sum	Sum of all items

Method	Description
get	Indexes each element
slice	Slices each element
slice_replace	Replaces slice in each element with the passed value
cat	Concatenates strings
repeat	Repeats values
normalize	Returns Unicode form of strings
pad	Adds whitespace to left, right, or both sides of strings
wrap	Splits long strings into lines with length less than a given width
join	Joins strings in each element of the Series with the passed separator
get_dummies	Extracts dummy variables as a DataFrame

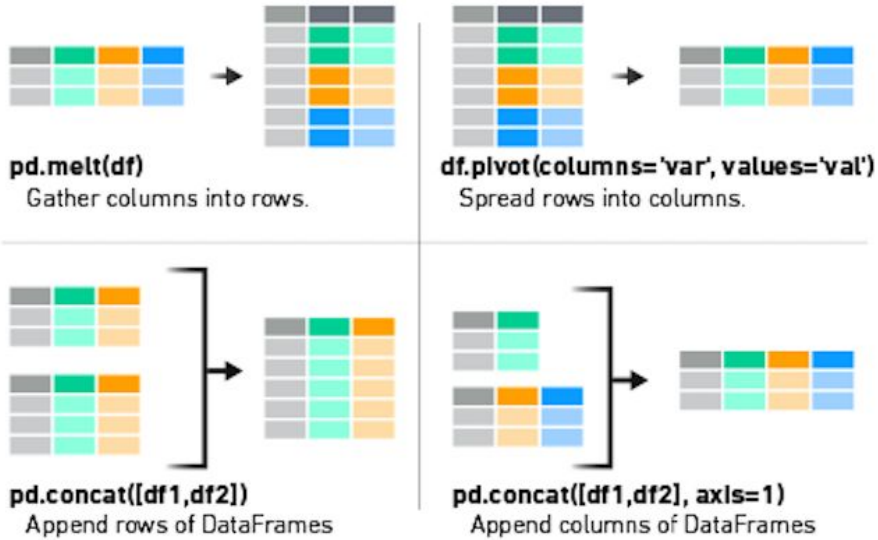
Lo más básico

	a	b	c
1	4	7	10
2	5	8	11
3	6	9	12

```
df = pd.DataFrame(
                {"a" : [4 ,5, 6],
                 "b" : [7, 8, 9],
                 "c" : [10, 11, 12]},
Index = [1, 2, 3])
```

		a	b	c
n	v			
d	1	4	7	10
	2	5	8	11
e	2	6	9	12

```
df = pd.DataFrame(
                {"a" : [4 ,5, 6],
                 "b" : [7, 8, 9],
                 "c" : [10, 11, 12]},
Index = pd.MultiIndex.from_tuples(
                [('d',1),('d',2),('e',2)],
                  names=['n','v']))
```


pd.melt(df)
Gather columns into rows.
df.pivot(columns='var', values='val')
Spread rows into columns.
pd.concat([df1,df2])
Append rows of DataFrames
pd.concat([df1,df2], axis=1)
Append columns of DataFrames

Set Operations

x1	x2
B	2
C	3

pd.merge(ydf, zdf)
Rows that appear in both ydf and zdf (Intersection).

x1	x2
A	1
B	2
C	3
D	4

pd.merge(ydf, zdf, how='outer')
Rows that appear in either or both ydf and zdf (Union).

x1	x2
A	1

```
pd.merge(ydf, zdf, how='outer',
         indicator=True)
        .query('_merge == "left_only"')
        .drop(columns=['_merge'])
```

Rows that appear in ydf but not zdf (Setdiff)

Standard Joins

x1	x2	x3
A	1	T
B	2	F
C	3	NaN

dpd.merge(adf, bdf, how='left', on='x1')
Join matching rows from bdf to adf.

x1	x2	x3
A	1.0	T
B	2.0	F
D	NaN	T

pd.merge(adf, bdf, how='right', on='x1')
Join matching rows from adf to bdf.

x1	x2	x3
A	1	T
B	2	F

pd.merge(adf, bdf, how='inner', on='x1')
Join data. Retain only rows in both sets.

x1	x2	x3
A	1	T
B	2	F
C	3	NaN
D	NaN	T

pd.merge(adf, bdf, how='outer', on='x1')
Join data. Retain all values, all rows.

Filtering Joins

x1	x2
A	1
B	2

adf[adf.x1.isin(bdf.x1)]
All rows in adf that have a match in bdf.

x1	x2
C	3

adf[~adf.x1.isin(bdf.x1)]
All rows in adf that do not have a match in bdf

Notebook Numpy & Pandas

Fecha de entrega: 4 sept, 12:10

Publicado: 28 ago

Convertir (cambiar de representación) los cheatsheets adjuntos en Notebooks de Jupyter.

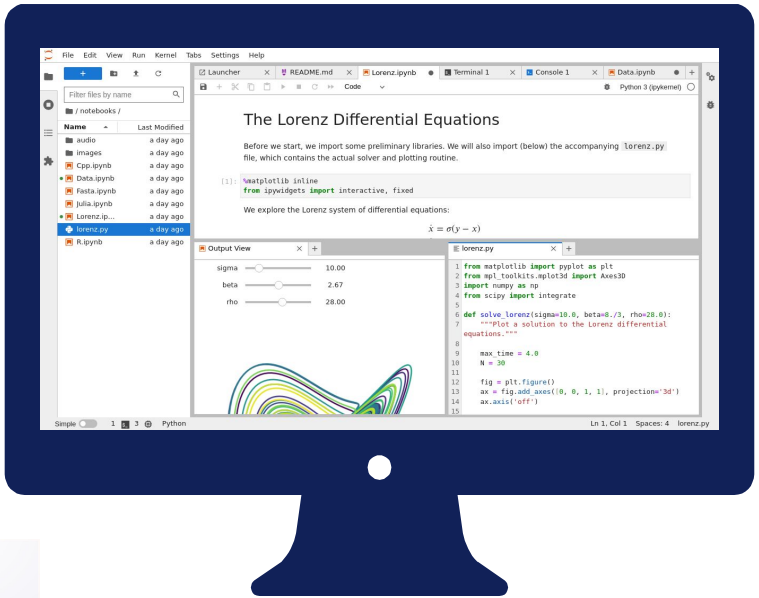
0	17
Entregadas	Asignadas

Numpy_Cheat_Sheet.pdf
PDF

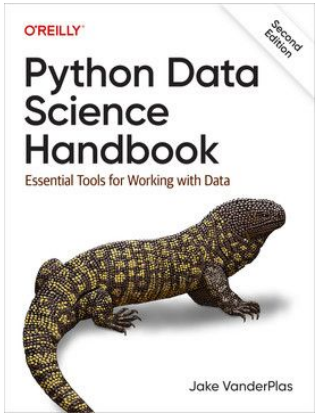
Pandas_Cheat_Sheet.pdf
PDF

Ver instrucciones

(Go to live notebook)



Extra Libro



GitHub

- 03.00-Introduction-to-Pandas.ipynb
- 03.01-Introducing-Pandas-Objects.ipynb
- 03.02-Data-Indexing-and-Selection.ipynb
- 03.03-Operations-in-Pandas.ipynb
- 03.04-Missing-Values.ipynb
- 03.05-Hierarchical-Indexing.ipynb
- 03.06-Concat-And-Append.ipynb
- 03.07-Merge-and-Join.ipynb
- 03.08-Aggregation-and-Grouping.ipynb
- 03.09-Pivot-Tables.ipynb
- 03.10-Working-With-Strings.ipynb
- 03.11-Working-with-Time-Series.ipynb
- 03.12-Performance-Eval-and-Query.ipynb
- 03.13-Further-Resources.ipynb

Notebook 1: Análisis exploratorio de dataset

Irvin Hussein Lopez Nava • 28 ago (Última modificación: 1:50)

100 puntos **Fecha de entrega: 11 sept**

- Crear un repositorio para cargar las prácticas realizadas en formato Notebook de Python, e.g., Github.
- La primera práctica consiste en seleccionar un Dataset de interés, y realizar un análisis exploratorio, detallando en cada bloque de código las funciones aplicadas, y considerando las buenas prácticas de programación.

Best Practices for Coding, O...
https://mitcommlab.mit.edu/broad/

Gracias!

¿Alguna pregunta?

hussein@cicese.mx
https://sites.google.com/view/husseinlopeznava

SCAN ME

CREDITS: This presentation was based on a template by **Slidesgo**, and includes icons by **Flaticon**.