

# Cluster 2018

## Ciencia de Datos en Ingeniería Industrial

clase\_00



**ANALIZABA GRANDES DATOS CON EXCEL**



# agenda\_clase00

- Consignas de la materia, presentación del equipo docente
- Features, samples, the curse of dimensionality
- Tipos de Aprendizaje, Supervisado vs No Supervisado
- Tipos de Datos
- Formato de los datos
- Data Science Workflow
- Primeras prácticas con Python

# Docentes Cluster



## Martin Palazzo

Machine Learning Instituto Max Planck &  
Université de Technologie de Troyes  
Doctorado en curso UTN-UTT  
Docente Inv. Op. UTN BA  
Master OSS (UTN-UTT)  
Ingeniero Industrial UTN BA



## Nicolas Aguirre

Machine Learning UTN FRBA &  
Université de Technologié de Troyes  
Doctorado en curso UTN-UTT  
Master OSS (UTN-UTT)  
Ingeniero Industrial UTN BA



## Agustin Velazquez

Data Analytics Developer en  
AlixPartners  
Master OSS (UTN-UTT)  
Docente Inv. Op. UTN BA  
Ingeniero Industrial UTN BA

# Miembros Cluster



**Matias Callara**

Data Scientist Roche (CH)  
Doctorado en Data Science (Francia)  
Master OSS (UTN-UTT)  
Ingeniero Industrial UTN BA



**Sebastian Pinto**

Docente UTN Cooperativismo  
Consejero depto. Industrial  
Master OSS (UTN-UTT)  
Ingeniero Industrial UTN BA

# Cluster online



**clusterraigroup@gmail.com**



**facebook.com/clusterai/**

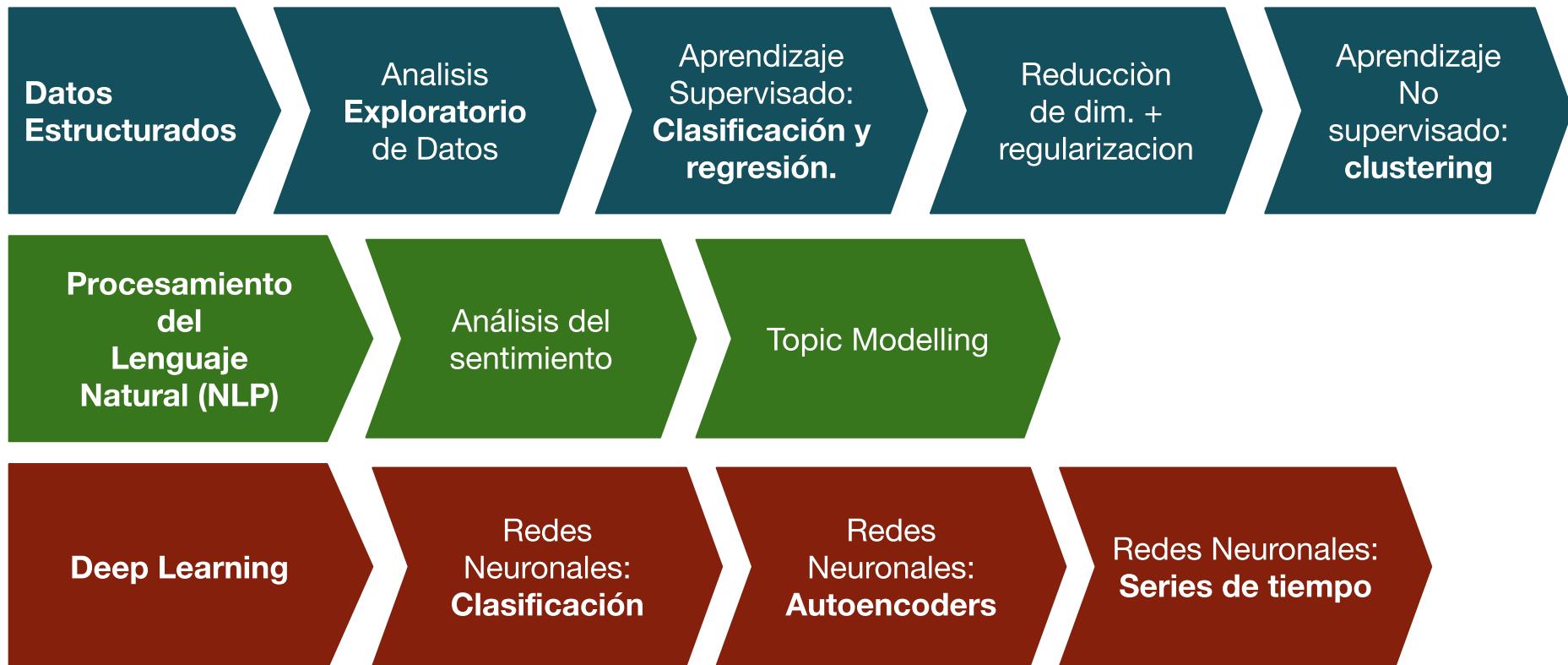


**twitter.com/clusterai**



**github.com/clusterai**

# Estructura del curso



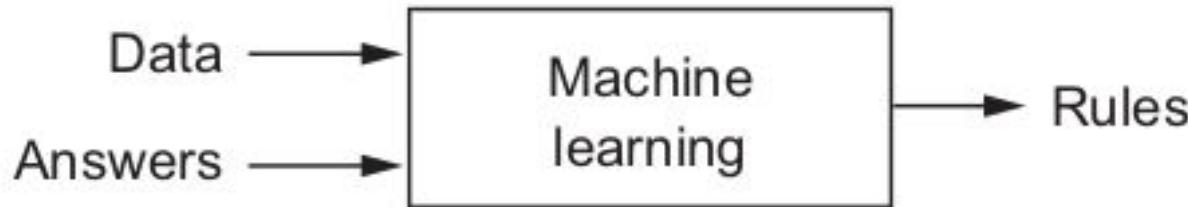
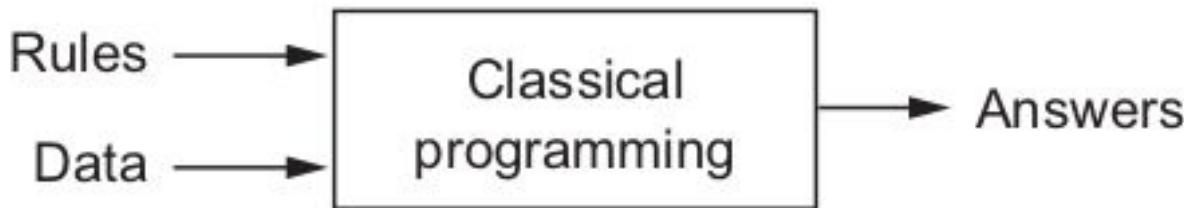
# Requisitos de aprobación

- Asistencia y seguimiento de las clases (4 faltas máximo).
- Entrega de trabajo práctico integral el 16 de Noviembre.
- Poster para ser presentado en el evento de fin de cursada (23 Nov).
- Aprobar parcial teórico (9 Nov).

# machine learning in a nutshell

aprender y construir modelos  
desde los datos.

# machine learning in a nutshell



# Samples (instancia) & Features (atributos)

Los datos estarán caracterizados por dos indicadores:

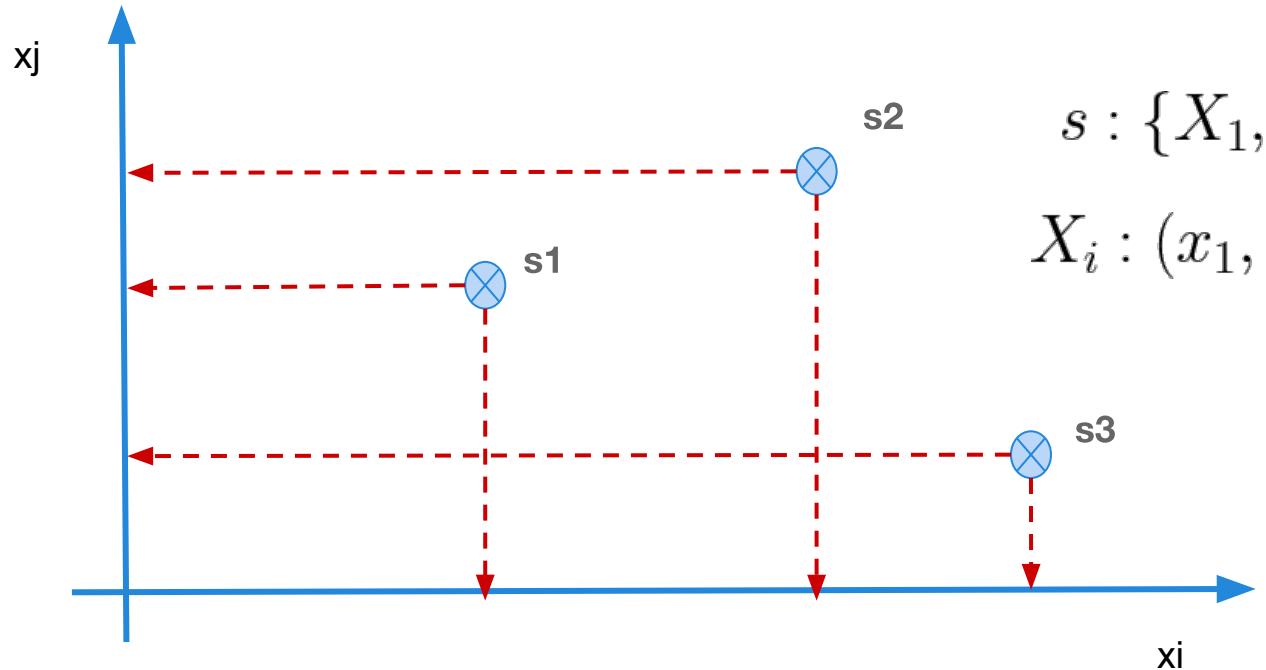
## Samples

Las samples corresponden a las instancias que obtenemos de una **muestra** de datos. Dicha muestra pertenece a una población que generalmente no conocemos por completo. Nuestro set de datos tendrá una cantidad determinada de samples.

## Features

Denominamos features (atributos o mediciones) a las **variables** que definen a cada sample (instancia). La cantidad de features que posea un sample es equivalente a la cantidad de **dimensiones** que describen a esa instancia en un espacio de alta dimensión. Nuestros datos “viven” en un espacio n-dimensional.

# Samples (instancia) & Features (atributos)

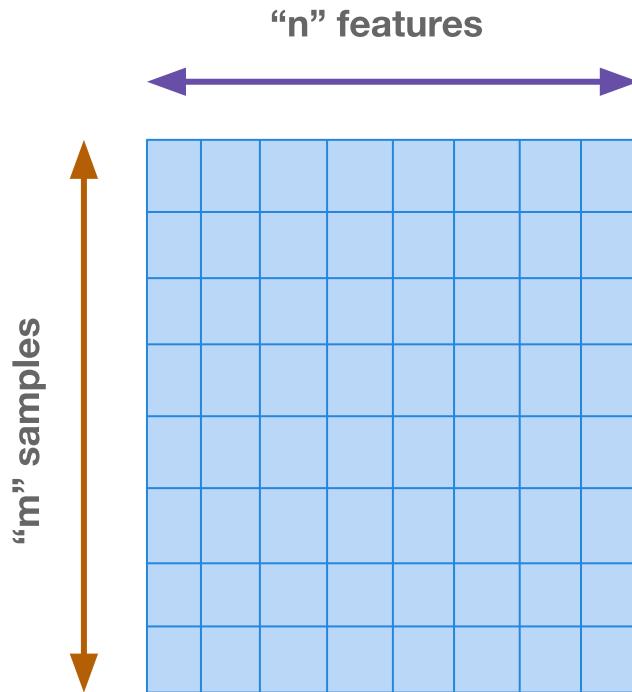


$s : \{X_1, X_2, X_3, X_4, \dots, X_m\}$

$X_i : (x_1, x_2, x_3, x_4, x_5, \dots, x_n)$

¿Cuántas features y cuantas samples hay en este ejemplo?

# Sample-to-feature ratio



$$S2FR = \frac{m}{n}$$

$S2FR \gg 1$  

$S2FR \ll 1$  

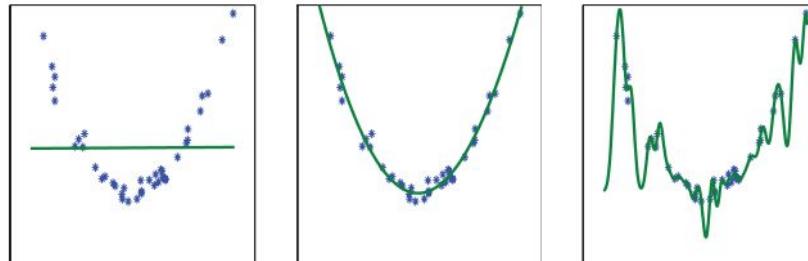
# Aprender de datos

Nuestro problema tendrá una característica particular: no conocemos la distribución de densidad de probabilidad (PDF) de nuestros datos. Nuestros datos difícilmente coincidan con una distribución normal o una exponencial. Lo más probable es que pertenezcan a una distribución compleja.

- Una opción es aproximarlos a distribuciones que ya conozcamos aunque no necesariamente esa es la mejor opción.
- Otra alternativa es aproximar funciones específicas a nuestros datos de manera tal que puedan detectar patrones sin intentar determinar el tipo de distribución que explica los datos. Existen varios enfoques para realizar esta estrategia.

# Aprender de datos

- Partiendo de un set de datos  $S$  aprenderemos una función “ $f(x)$ ” desconocida y será el estimador que utilizaremos.
- Nunca llegaremos a una “ $f(x)$ ” ideal que explique a la perfección nuestros datos, por ende tendremos cierto grado de error. La función  $f(x)$  supone una distribución de probabilidad “ $p(x)$ ” que es incierta.
- Vamos a querer que nuestra función a aprender **generalice** bien para futuros datos nunca vistos (es decir, una vez que encontramos el patrón en los datos disponibles, que siga encontrando los mismos patrones para datos futuros).



# Tipos de Aprendizaje

## Aprendizaje:

- Supervisado
- No Supervisado
- Semi-supervisado
- Por refuerzo

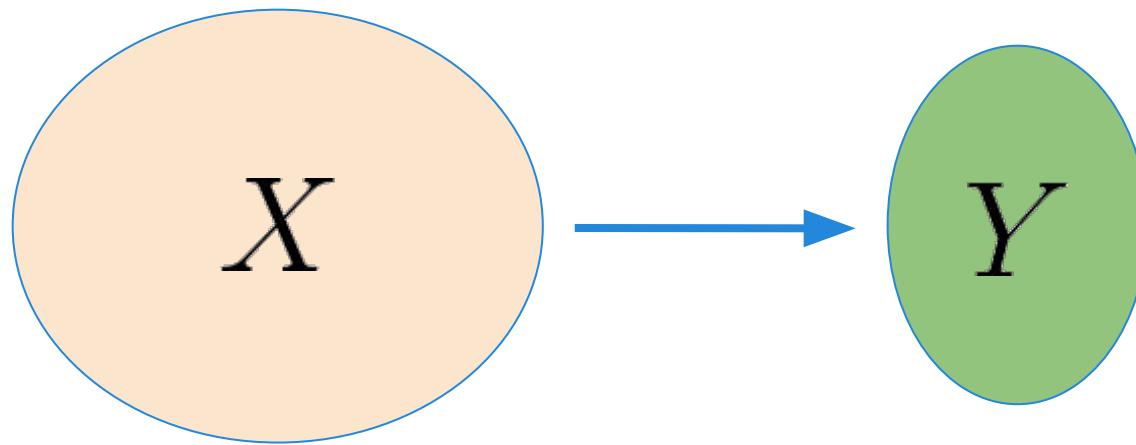
# Aprendizaje Supervisado

$$S = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$$

Suponemos un dataset con observaciones/samples S, donde  $X_i$  es un vector de features e  $Y_i$  es una label (etiqueta) asociada a cada observación  $X_i$ .

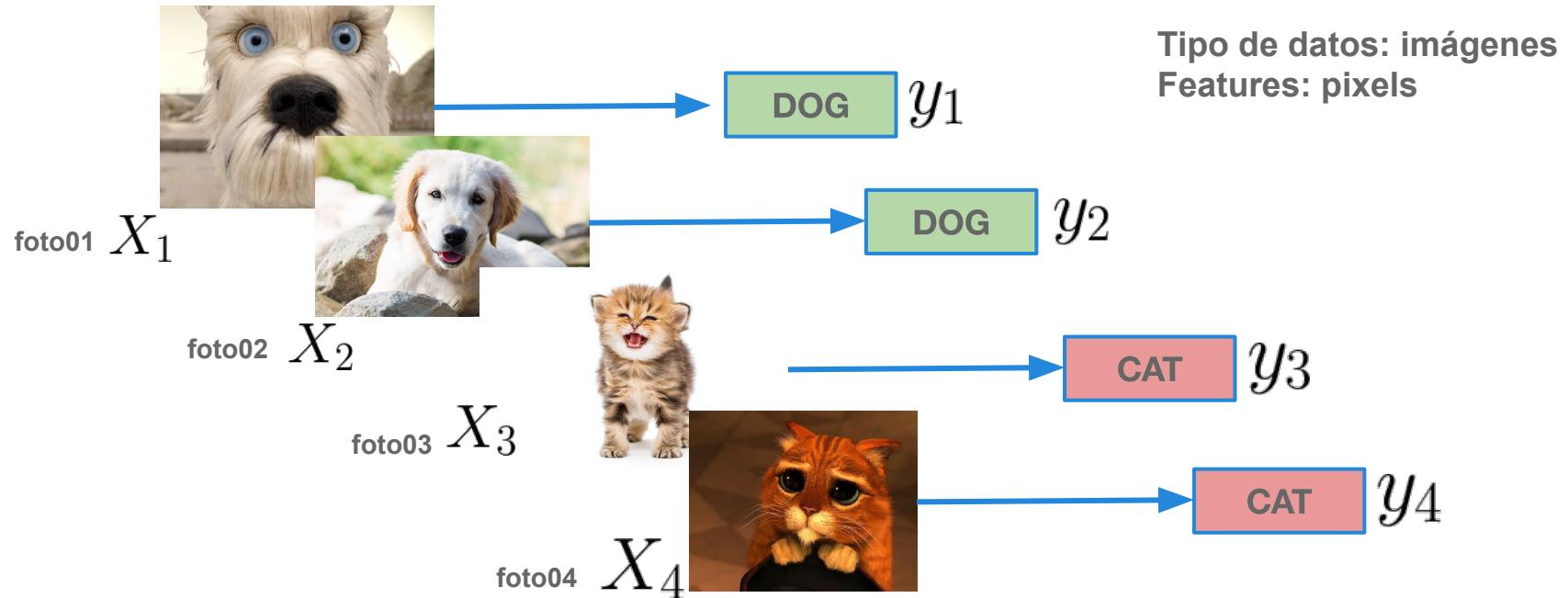
Solemos denominar a cada observación  $X_i$  como “sample” y a cada etiqueta  $Y_i$  como “label”.

# Aprendizaje Supervisado



Suponemos que la variable Y es dependiente de X. Esto quiere decir que Y está condicionada y es consecuencia de X. Lo que no conocemos es la función  $y = f(x)$  y es  $f(x)$  lo que queremos aprender desde los datos.

# Tipos de Aprendizaje: Supervisado



Cada instancia (sample) viene acompañada de una etiqueta (label).

# Tipos de Aprendizaje: Supervisado

Tipo de datos: ADN  
Features: mutaciones

```
##fileformat=VCFv4.1
##fileDate=20200908
##referenceFile:///seq/references/1000GenomesPilot-WCHI36.fasta
##referenceName="WCHI36",assembly="GRCh38",md5="12ec0f8edc737d96181f6eb2da,species="Homo sapiens",taxonomy>
##phasing=partial
##INFO<ID=AF>Number=1,Type=Integer,Description="Number of Samples With Data"
##INFO<ID=DP>Number=1,Type=Integer,Description="Total Depth"
##INFO<ID=RF>Number=1,Type=Float,Description="Allele Frequency"
##INFO<ID=NS>Number=1,Type=Integer,Description="Number of Samples With Data"
##INFO<ID=QD>Number=1,Type=Float,Description="Quality by Depth"
##INFO<ID=MQ>Number=1,Type=Float,Description="Mapping Quality"
##INFO<ID=MQ0>Number=1,Type=Float,Description="Mapping Quality 0"
##INFO<ID=FS>Number=1,Type=String,Description="Fasta Quality"
##FORMAT<ID=DP>Number=1,Type=Integer,Description="Read Depth"
##FORMAT<ID=GT>Number=2,Type=Integer,Description="Readotype Quality">
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA00001 NA00002 NA00003
20 14370 rs6084297 G A 29 PASS NS=3,DP=14,AF=0.5,BQ:82 GT:0/1,DP:82 0/1:48:8:51,81 1/1:43:5,..-
20 11106966 rs600385 A G,T 67 PASS NS=10,DP=10,AF=0.333,0.667,AA=T,DS GT:0/1,DP:10 1/1:6:23,27 211:2:0:16,2 2/2:35:4
20 1230237 . T . 47 PASS NS=4,DP=13,AA=T GT:0/1,DP:13 0/1:48:4,51,81 0/0:61:2
20 1234667 miscreatl GTC G,GTCT 50 PASS NS=3,DP=9,AA=0 GT:0/1,DP:9 0/1:35:4 0/2:17:2 1/1:40:3
```

paciente01  $X_1$

CONTROL SANO  $y_1$

```
##fileformat=VCFv4.1
##fileDate=20200908
##referenceFile:///seq/references/1000GenomesPilot-WCHI36.fasta
##referenceName="WCHI36",assembly="GRCh38",md5="12ec0f8edc737d96181f6eb2da,species="Homo sapiens",taxonomy>
##phasing=partial
##INFO<ID=AF>Number=1,Type=Integer,Description="Number of Samples With Data"
##INFO<ID=DP>Number=1,Type=Integer,Description="Total Depth"
##INFO<ID=RF>Number=1,Type=Float,Description="Allele Frequency"
##INFO<ID=NS>Number=1,Type=Integer,Description="Number of Samples With Data"
##INFO<ID=QD>Number=1,Type=Float,Description="Quality by Depth"
##INFO<ID=MQ>Number=1,Type=Float,Description="Mapping Quality"
##INFO<ID=MQ0>Number=1,Type=Float,Description="Mapping Quality 0"
##INFO<ID=FS>Number=1,Type=String,Description="Fasta Quality"
##FORMAT<ID=DP>Number=1,Type=Integer,Description="Read Depth"
##FORMAT<ID=GT>Number=2,Type=Integer,Description="Readotype Quality">
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA00001 NA00002 NA00003
20 14370 rs6084297 G A 29 PASS NS=3,DP=14,AF=0.5,BQ:82 GT:0/1,DP:82 0/1:48:8:51,81 1/1:43:5,..-
20 11106966 rs600385 A G,T 67 PASS NS=10,DP=10,AF=0.333,0.667,AA=T,DS GT:0/1,DP:10 1/1:6:23,27 211:2:0:16,2 2/2:35:4
20 1230237 . T . 47 PASS NS=4,DP=13,AA=T GT:0/1,DP:13 0/1:48:4,51,81 0/0:61:2
20 1234667 miscreatl GTC G,GTCT 50 PASS NS=3,DP=9,AA=0 GT:0/1,DP:9 0/1:35:4 0/2:17:2 1/1:40:3
```

paciente02  $X_2$

CANCER  $y_2$

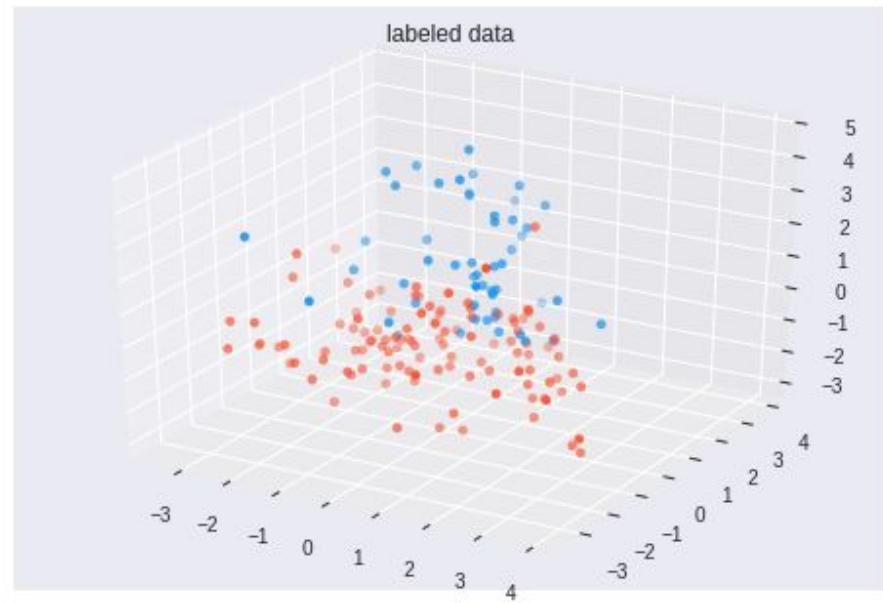
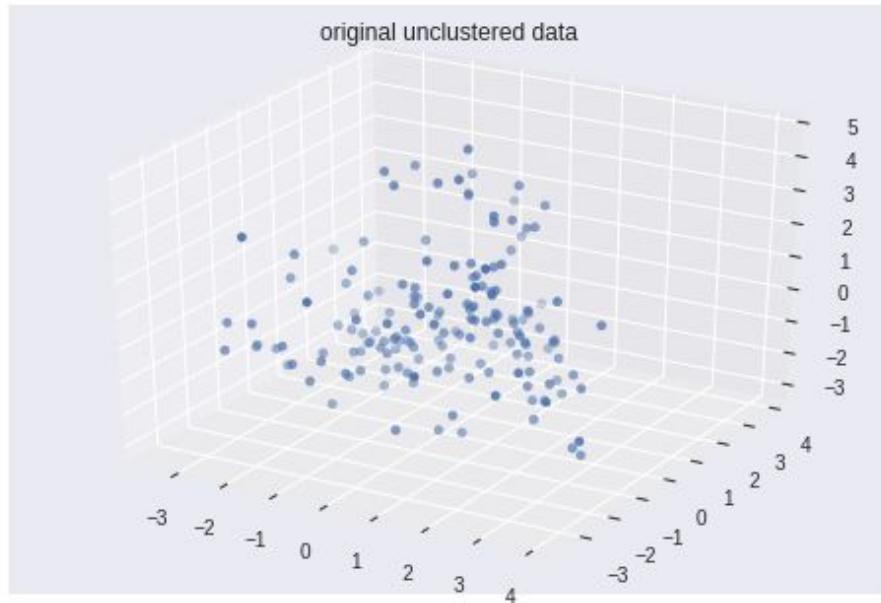
```
##fileformat=VCFv4.1
##fileDate=20200908
##referenceFile:///seq/references/1000GenomesPilot-WCHI36.fasta
##referenceName="WCHI36",assembly="GRCh38",md5="12ec0f8edc737d96181f6eb2da,species="Homo sapiens",taxonomy>
##phasing=partial
##INFO<ID=AF>Number=1,Type=Integer,Description="Number of Samples With Data"
##INFO<ID=DP>Number=1,Type=Integer,Description="Total Depth"
##INFO<ID=RF>Number=1,Type=Float,Description="Allele Frequency"
##INFO<ID=NS>Number=1,Type=Integer,Description="Number of Samples With Data"
##INFO<ID=QD>Number=1,Type=Float,Description="Quality by Depth"
##INFO<ID=MQ>Number=1,Type=Float,Description="Mapping Quality"
##INFO<ID=MQ0>Number=1,Type=Float,Description="Mapping Quality 0"
##INFO<ID=FS>Number=1,Type=String,Description="Fasta Quality"
##FORMAT<ID=DP>Number=1,Type=Integer,Description="Read Depth"
##FORMAT<ID=GT>Number=2,Type=Integer,Description="Readotype Quality">
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA00001 NA00002 NA00003
20 14370 rs6084297 G A 29 PASS NS=3,DP=14,AF=0.5,BQ:82 GT:0/1,DP:82 0/1:48:8:51,81 1/1:43:5,..-
20 11106966 rs600385 A G,T 67 PASS NS=10,DP=10,AF=0.333,0.667,AA=T,DS GT:0/1,DP:10 1/1:6:23,27 211:2:0:16,2 2/2:35:4
20 1230237 . T . 47 PASS NS=4,DP=13,AA=T GT:0/1,DP:13 0/1:48:4,51,81 0/0:61:2
20 1234667 miscreatl GTC G,GTCT 50 PASS NS=3,DP=9,AA=0 GT:0/1,DP:9 0/1:35:4 0/2:17:2 1/1:40:3
```

paciente03  $X_3$

CANCER  $y_3$

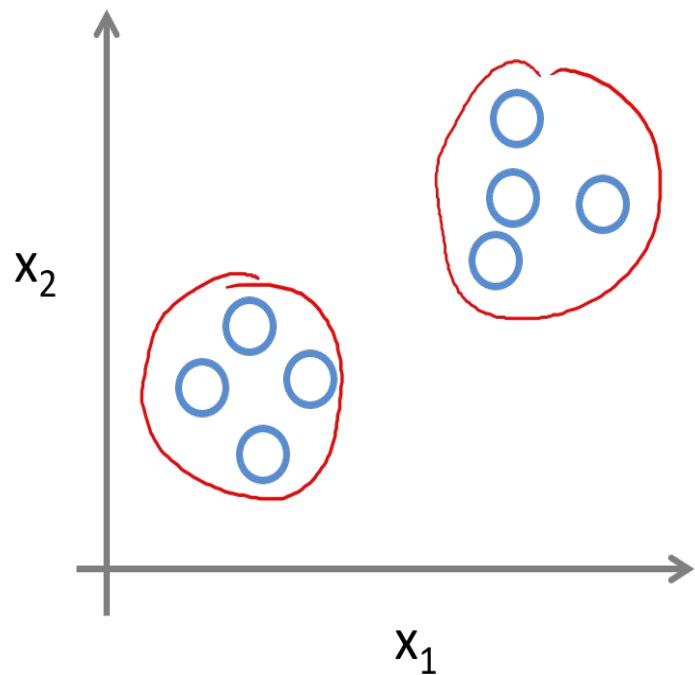
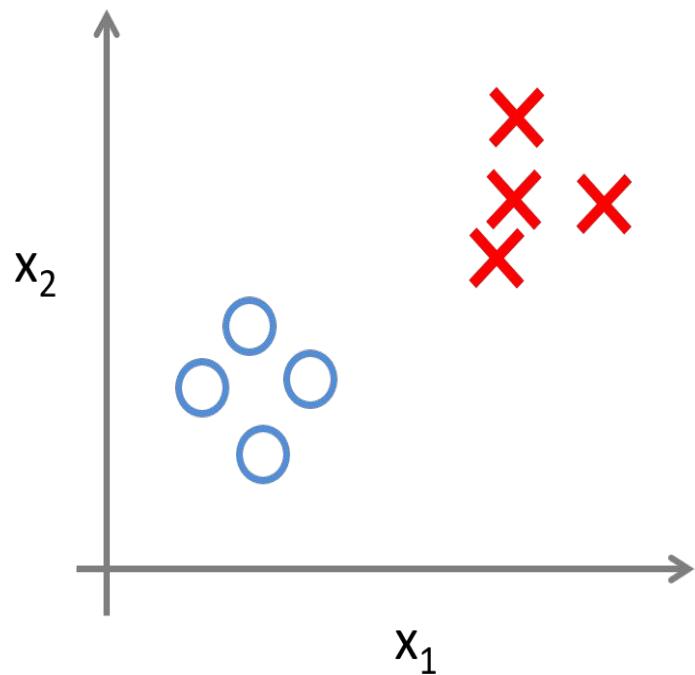
```
##fileformat=VCFv4.1
##fileDate=20200908
##referenceFile:///seq/references/1000GenomesPilot-WCHI36.fasta
##referenceName="WCHI36",assembly="GRCh38",md5="12ec0f8edc737d96181f6eb2da,species="Homo sapiens",taxonomy>
##phasing=partial
##INFO<ID=AF>Number=1,Type=Integer,Description="Number of Samples With Data"
##INFO<ID=DP>Number=1,Type=Integer,Description="Total Depth"
##INFO<ID=RF>Number=1,Type=Float,Description="Allele Frequency"
##INFO<ID=NS>Number=1,Type=Integer,Description="Number of Samples With Data"
##INFO<ID=QD>Number=1,Type=Float,Description="Quality by Depth"
##INFO<ID=MQ>Number=1,Type=Float,Description="Mapping Quality"
##INFO<ID=MQ0>Number=1,Type=Float,Description="Mapping Quality 0"
##INFO<ID=FS>Number=1,Type=String,Description="Fasta Quality"
##FORMAT<ID=DP>Number=1,Type=Integer,Description="Read Depth"
##FORMAT<ID=GT>Number=2,Type=Integer,Description="Readotype Quality">
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA00001 NA00002 NA00003
20 14370 rs6084297 G A 29 PASS NS=3,DP=14,AF=0.5,BQ:82 GT:0/1,DP:82 0/1:48:8:51,81 1/1:43:5,..-
20 17370 . T A 47 PASS NS=3,DP=11,AF=0.333,0.667,AA=T,DS GT:0/1,DP:11 0/1:49:5,56,50 0/1:31:6,3 0/0:41:..-
20 11106966 rs600385 A G,T 67 PASS NS=10,DP=10,AF=0.333,0.667,AA=T,DS GT:0/1,DP:10 1/1:6:23,27 211:2:0:16,2 2/2:35:4
20 1230237 . T . 47 PASS NS=4,DP=13,AA=T GT:0/1,DP:13 0/1:48:4,51,81 0/0:61:2
20 1234667 miscreatl GTC G,GTCT 50 PASS NS=3,DP=9,AA=0 GT:0/1,DP:9 0/1:35:4 0/2:17:2 1/1:40:3
```

# Tipos de Aprendizaje: No supervisado



Cada instancia (sample) no posee etiqueta (izq). Los modelos a aplicar en estos casos buscan encontrar estructuras o grupos implícitas en los datos (ej. clusters)

# Supervisado vs No supervisado



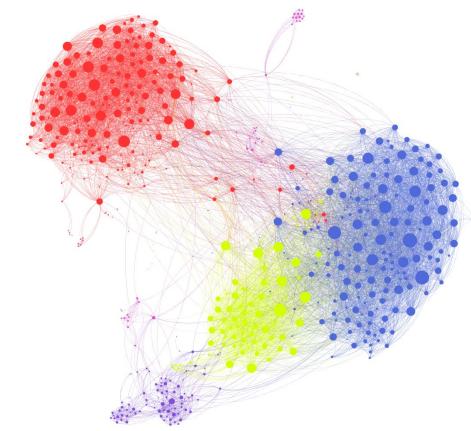
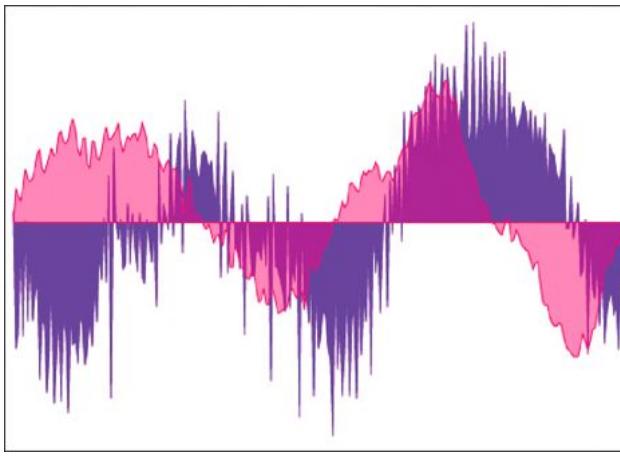
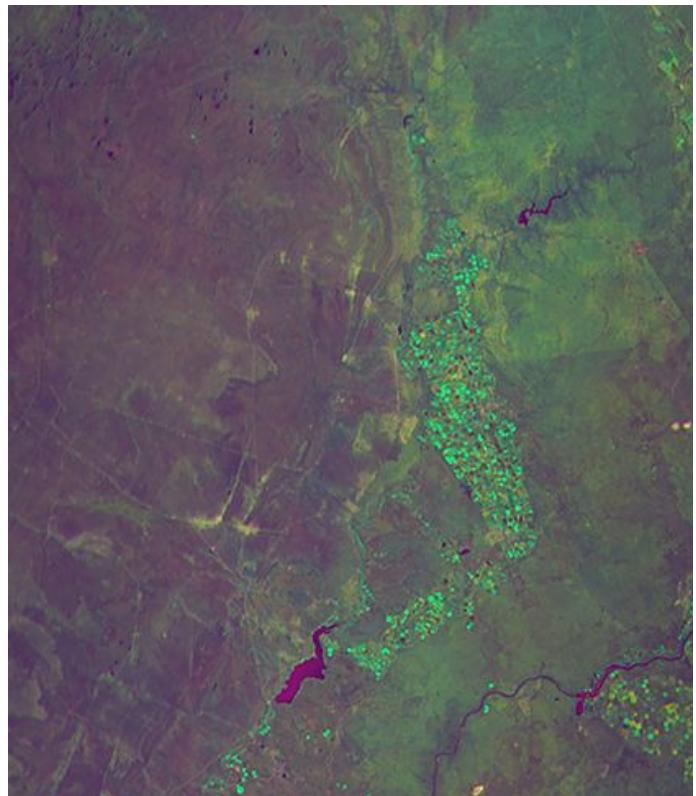
# Tipos de Datos

- Estructurados / tabulares
- Imàgenes
- Grafos-redes
- Lenguaje Natural (texto)
- Audio / Series de tiempo

# Formato de los datos

- .CSV
  - .xlsx
    - .txt
  - .tsv
  - .jpeg
    - SQL query

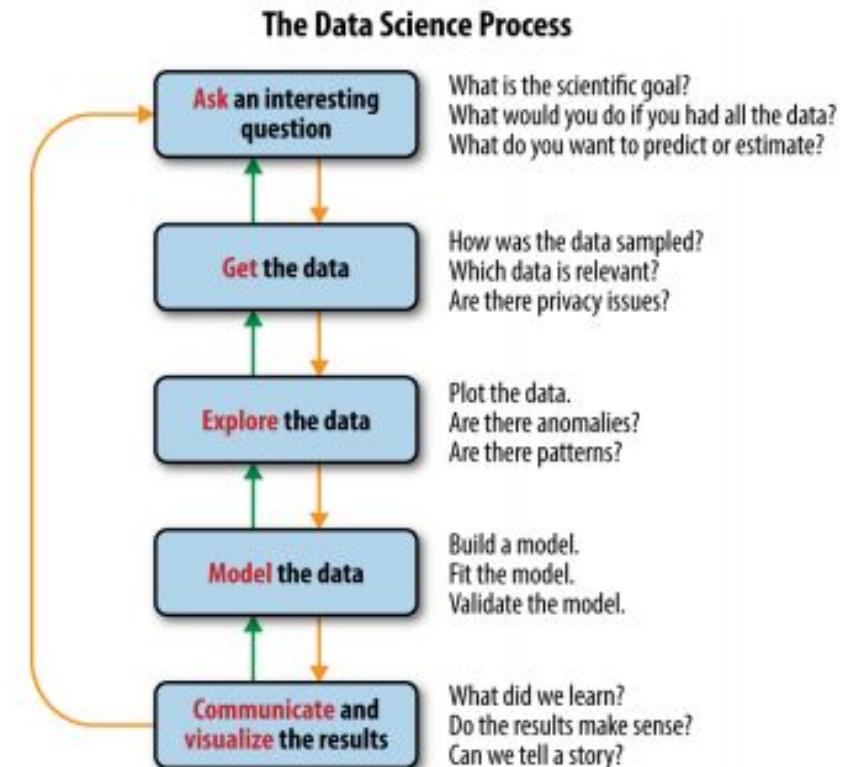
# Tipos de Datos



**text as data**  
quantitative analysis

ability mainly  
already advanced  
knowledge practitioners  
research textual methods  
Software inductive  
London already  
using textual methods  
already advanced  
practitioners  
Finally applicable  
book-length foundations  
courses Economics  
analytical practitioner  
innovative conference  
computer-assisted work powerful  
sciences scaling  
models social assumptions  
data applications  
researchers analyzing  
unlocking nQUANTESS features traditional  
locally developing group development  
whether explicit blog statistical  
approach develop programming traits  
packaging treatment  
means new testing  
possible bring  
field teams freely  
language addition students  
dissemination training  
project particularly  
high single  
sharply implementing  
poorly ubiquitous  
QUANTESS degree  
techniques  
distributes third users  
untested volumes state  
distributing training  
users implementation  
single  
high project  
sharply explaining  
poorly implementing  
ubiquitous  
QUANTESS degree  
techniques  
distributes third users  
untested volumes state  
distributing training  
users implementation  
single  
high project  
sharply explaining  
poorly implementing  
ubiquitous

# Data Science Workflow



\*Development Workflows for Data Scientists

# 1) Data Science Workflow: get the data



## Buenos Aires Data

Iniciativa de Datos Públicos y Transparencia de la Ciudad Autónoma de Buenos Aires.



Durante el curso trataremos de utilizar repositorios de datos abiertos, principalmente aquellos de la Ciudad de Buenos Aires, Provincia de Buenos Aires o Nación.

# 2) Data Science Workflow: Explore

## Pre-processing

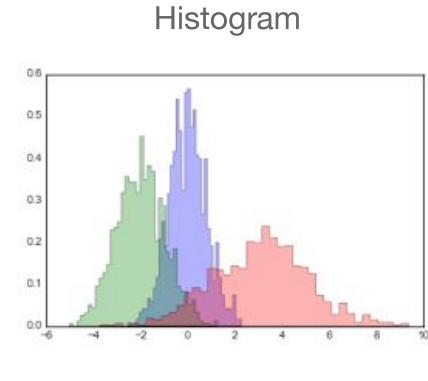
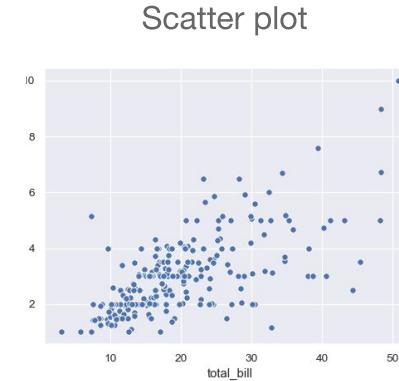
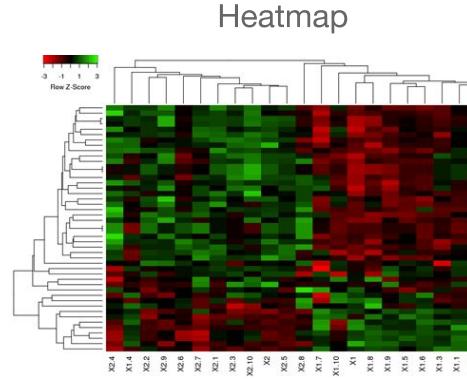
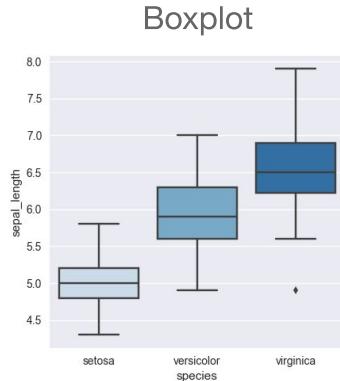
- Clean samples with NaNs
- Transform features
- Normalize data

## Exploratory Data Analysis

- Realizar estadísticas descriptivas
- Quitar outliers estadísticos
- Visualizar con Bar-plots, Box-plots, Scatter-plots, Count-plots

# 2) Exploratory Data Analysis

- Importar datos.
- Revisar si hay NaNs o valores faltantes.
- Filtrar los datos de interés.
- Transformar los datos (ej, tabla pivote)
- Computar estadísticas descriptivas (media, dev. std, percentiles)
- Medir correlación entre variables de interés
- Visualizar:



### 3) Data Science Workflow: Model

clasificación

regresión

Reducción de dimensionalidad

Detección de anomalías

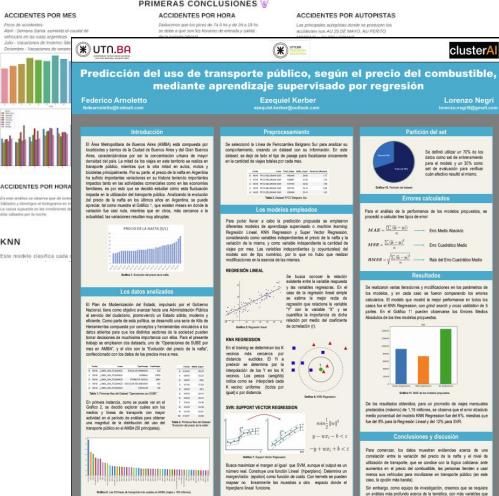
clustering

# 4) Communicate Results

Poster explicando el proyecto



**RESUMEN**  
Este poster pretende dar a conocer un desarrollo en investigación a fin de poder implementar los modelos de aprendizaje supervisado. KNN y SVM, para predecir si un accidente va a haber víctimas o no. Se realizó un análisis exploratorio de datos con los cuales llamarlos e trabajar. Utilizamos Dummies como herramientas para poder predecir si un accidente va a haber víctimas o no. Implementamos los dos modelos de aprendizaje supervisados: KNN y SVM. Se realizó una prueba de hipótesis para ver si el modelo de KNN es mejor que el de SVM. El resultado es que el modelo de KNN es mejor que el de SVM. Accuracy: 0.747, AUC: 0.785. Con estos resultados podemos decir que el modelo de KNN es mejor que el de SVM.



Reporte técnico del proyecto

APLICACIÓN DE MODELOS DE CLUSTERING AL  
ANÁLISIS DE RESULTADOS DE LAS PRUEBAS APRENDER  
EN LA PROVINCIA DE BUENOS AIRES  
Universidad Tecnológica Nacional  
Cátedra de Ciencia de Datos - 2018

Cañabate, Marcos      Delgado, Luciano      Szapsiowicz, Juan  
  
ABSTRACT: a lo largo del presente informe nos proponemos analizar los datos obtenidos a partir de las pruebas APRENDER, con el objetivo de comprender ciertos aspectos referentes la situación académica de la Provincia de Buenos Aires.

clusterAI	GRUPO 5		
	Vencimiento	Cupo	Página
Ciencia de Datos	10/11/2018	1	1 de 7

IMPLEMENTACION DE MODELOS DE APRENDIZAJE SUPERVISADO  
PARA PREDICIR VICTIMAS DE ACCIDENTES DE TRÁNSITO EN  
AUTOPISTAS AUSA

Malena Ruan Barés  
Leg: 150572-5  
mruanbares@gmail.com

Diego Rog  
Leg: 117602-0  
diego.bsa@outlook.com

Eduardo Andino  
Leg: 116647-5  
andino@persat.com.ar

## 1. RESUMEN

Se puede predecir si en un accidente en autopista va a haber víctimas? Es la pregunta que disparó nuestro proyecto.

A partir de data set del Gobierno de la Ciudad implementamos dos modelos de aprendizaje supervisados:

- KNN
- SVM

Bueno mostramos el mejor de los dos, que es KNN. Aunque se realizó un análisis exploratorio de datos, se observó que el mejor resultado se obtuvo con el modelo KNN. Con respecto a la ejecución del proyecto, se realizó un análisis exploratorio de datos, se realizó una descripción de los datos y se realizó una inferencia estadística.

## 3. OBJETIVO

En base al tipo de accidente y todas las condiciones dadas, predecir si hay lesionados y/o fallecidos.

Este objetivo es planteado desde la premisa que solo con los datos de accidentes, no encontramos un dato set que contiene los datos de las condiciones de la autopista o condiciones meteorológicas de días que no haya habido

Exposición de posters



Repository en github del proyecto

Overview    Repositories 8    Projects 0    Stars 2    Followers 9    Follow

Popular repositories

**primex\_pass\_python**  
Este repositorio fue creado con el fin de ser utilizado en cursos donde se dicta en el lenguaje Python desde el comienzo y donde no existe el tiempo suficiente para que los estudiantes lo lean.

• Update repository    2 · 1

**ssh\_localhost\_port\_forwarding\_example**  
Fichero que hace la conexión entre un port local y un port remoto.

• Update repository    2 · 1

**martinpalazzo**  
martinpalazzo

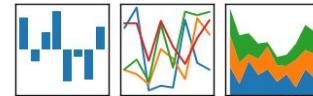
# Tecnologías que utilizaremos



Librerías:



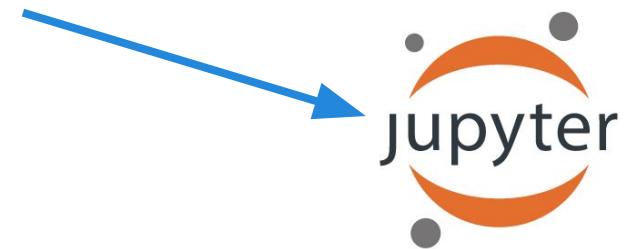
pandas  
 $y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$



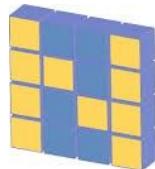
# Python <> Anaconda <> Jupyter



ANACONDA®



# Numpy vs Pandas



NumPy

Calculo con matrices

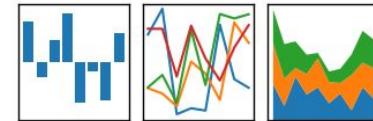
- No admite nombres en cols
- No admite nombres en filas
- Diversidad en aplicaciones de cálculo
- Útil para lidiar con álgebra y operaciones matriciales

Atajos de Numpy acá:

[https://s3.amazonaws.com/assets.datacamp.com/blog\\_assets/Numpy\\_Python\\_Cheat\\_Sheet.pdf](https://s3.amazonaws.com/assets.datacamp.com/blog_assets/Numpy_Python_Cheat_Sheet.pdf)

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



Gestor de datasets en Dataframes (DFs)

- Admite nombre de columnas
- Admite nombre de filas
- Diversas funciones sobre DFs.
- Útil para lidiar con datos, limpiar, pre procesar.

Atajos de Pandas acá:

[https://pandas.pydata.org/Pandas\\_Cheat\\_Sheet.pdf](https://pandas.pydata.org/Pandas_Cheat_Sheet.pdf)

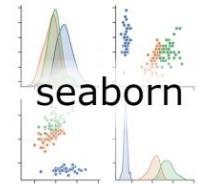
# Pandas Dataframe

pd.columns

pd.index

**Dataframe name = 'pd'**

# Visualización



Librerías de visualización de datos

- Matplotlib es la principal librería de visualización en Python.
- Seaborn corre sobre matplotlib y posee algunas mejoras de estética.
- Tipos de gráficos a realizar:
  - Countplot (graficos de barra)
  - Heatmap (mapas de calor)
  - Boxplot (diagrama de cajas y bigote)
  - Series de tiempo
  - Scatter plot (diagrama de puntos)
  - Distplot (distribuciones y densidades)

Atajos de Matplotlib acá:

[https://s3.amazonaws.com/assets.datacamp.com/blog\\_assets/Python\\_Matplotlib\\_Cheat\\_Sheet.pdf](https://s3.amazonaws.com/assets.datacamp.com/blog_assets/Python_Matplotlib_Cheat_Sheet.pdf)

# Tipos de variables en Python



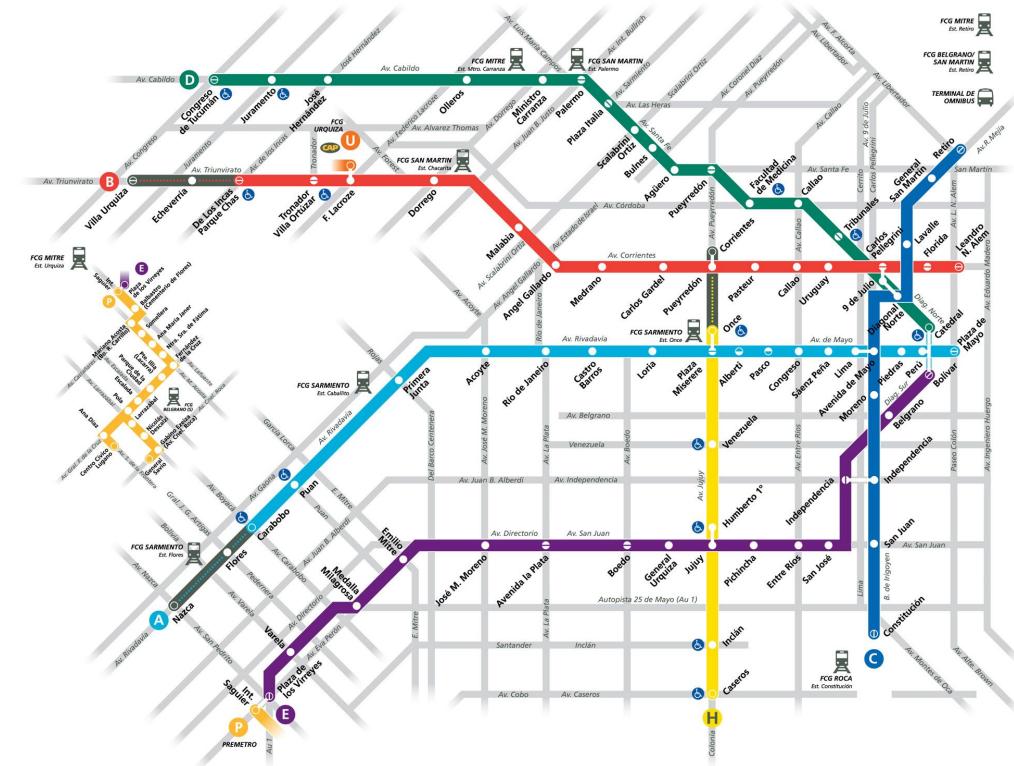
```
a = 3  
b = 0.4  
c = False  
d = "Quiero analizar datos"  
e = [2,3,4,5,6]  
f = [[2,3,4],[1,0,40]]
```

**a = Integer**  
**b = Float**  
**c = Boolean**  
**d = String**  
**e = Numpy Array (1,5)**  
**f = Numpy Array (2,3)**

# A agarrar la PyLA



# Preproc. + EDA con data de subtes



# Import libraries



```
import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns
```

# Import Data



```
molinetes = pd.read_csv( '/home/human/Dropbox/clusterai/molinetes_historico.csv', delimiter=';', index_col=[ 'PERIODO' ] )
```

# Exploratory Data Analysis

