



La Escuela Superior de Física y Matemáticas
te invita al **Taller**

Introducción al Aprendizaje Automático.

Fabiola Rodriguez & Daniel Legorreta

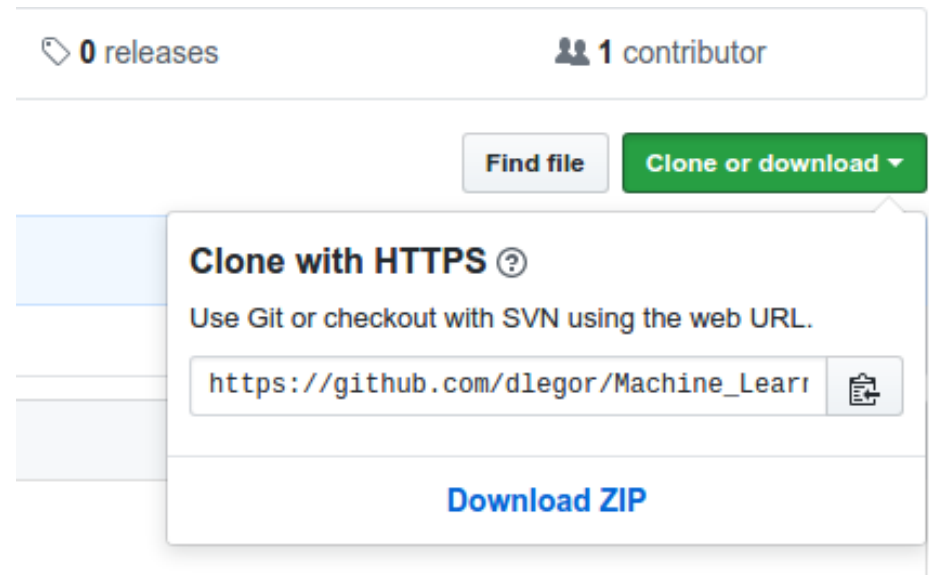
Material del Taller

Todo el contenido del taller se encuentra disponible en:

https://github.com/dlegor/Machine_Learning_con_Python_2017

Material del Taller

Para obtener el material para el taller basta ir al sitio y seleccionar el icono de “Clone or download”.



Temario del Taller

Jueves 22 de Junio:

- Presentación e Introducción
- Breve Introducción a Python
- Entorno de trabajo para el taller
- Introducción al Aprendizaje Automático
- Aprendizaje Supervisado - Clasificación
- Aprendizaje Supervisado – Regresión

Viernes 23 de Junio:

- Aprendizaje No Supervisado - Reducción de Dimensiones
- Aprendizaje No Supervisado – Detección de Cluster
- Métricas para los Modelos Supervisados
- Validación Cruzada y Selección de Parámetros
- Modelos Lineales
- Modelos SVM
- Modelos Ensamble

Entorno necesario



Las bibliotecas requeridas son:

- NumPy
- SciPy
- Pandas
- Scikit-learn
- Matplotlib
- Seaborn
- Bokeh

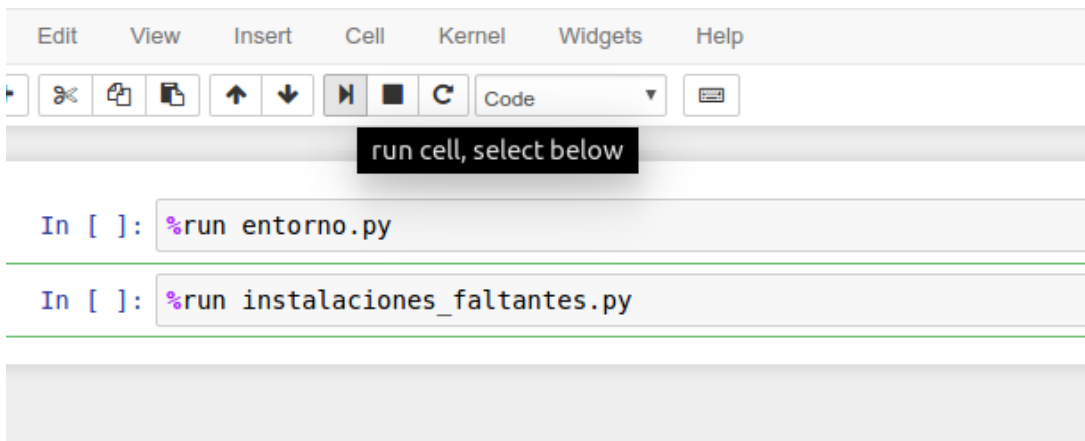
Entorno necesario

```
dlegorreta@dlegorreta-K46CB:~/Documentos/Python_2017/Notas_Curso_ML_2017$ python entorno.py
Ubicación de Python en: /home/dlegorreta/anaconda3

Versión de Python:
3.6.0 [Anaconda custom (64-bit)] (default, Dec 23 2016, 12:22:00)
[GCC 4.4.7 20120313 (Red Hat 4.4.7-1)]

Paquetes y versión que se tienen Instaladas:
[ OK ] numpy version 1.12.1
[ OK ] scipy version 0.19.0
[ OK ] matplotlib version 2.0.2
[ OK ] IPython version 6.0.0
[ OK ] sklearn version 0.18.1
[ OK ] pandas version 0.20.1
[ OK ] seaborn version 0.7.1
[ OK ] statsmodels version 0.8.0
[ OK ] yaml version 3.12
[ OK ] PIL version 1.1.7
```

Para validar el entorno se
puede ejecutar:
entorno.py



Programación en Python

- Lenguaje de multiparadigma: *orientado a objetos, programación imperativa y en menor medida funcional.*
- Se usará para el taller Python 3.6
- Para los ejemplos se usará de manera interactiva desde las celdas de Jupyter.
- Breve revisión del material: **cs228 Tutorial Python**

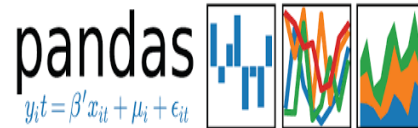
Tres ejercicios en Python

Abrir un Notebook en Jupyter para realizar los ejercicios:

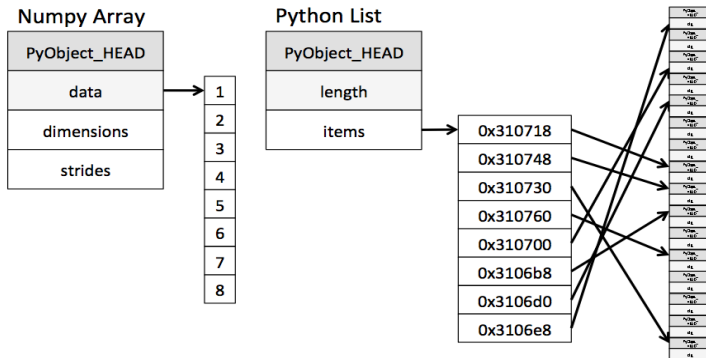
- Escribir una función que reciba una lista y regrese una lista con los mismo elementos pero ordenados al revés. No usar el método `reverse()` de tipo de dato `List`.
- Escribir una función que, dado un número entero `N`, devuelva una lista con todos los números primos hasta `N`. Para solucionar el ejercicio deberás crear una función auxiliar que indique si un determinado número es primo (retornando un valor booleano).
- 3. Escribir una función que reciba una tupla compuesta por caracteres, y devuelva una lista con los caracteres en mayúsculas. Deberás recorrer la tupla carácter a carácter para realizar la conversión. Para convertir un carácter a mayúscula podéis usar el método `upper()`. Por ejemplo `'a'.upper()` nos devuelve `'A'`.

Ejemplos de las Bibliotecas Necesarias

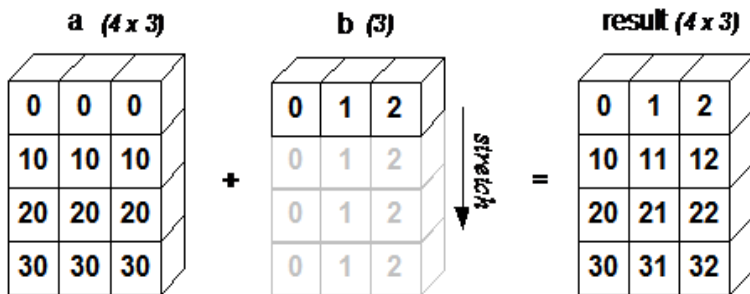
- Se revisa el material del Notebook: **00_Introduccion_ML.ipynb**



NumPy



- Permite la construcción de arreglos y computo vectorizado.
- Cuenta con capacidades de indexación que facilita la manipulación de los arreglos.
- Los arreglos son ordenados en memoria de manera tal que hace un procesamiento sumamente rápido.
- Cuenta con operaciones como Broadcasting que facilitan la programación y mantiene los niveles de eficiencia.



SciPy



```
from scipy.misc import imread, imsave, imresize

# Se lee una imagen JPEG como un arreglo en Numpy
img = imread('Doc/cat.jpg')
print(img.dtype, img.shape) # Se imprime "uint8 (400, 248, 3)"

img_tinted = img * [1, 0.95, 0.9]

# Resize the tinted image to be 300 by 300 pixels.
img_tinted = imresize(img_tinted, (300, 300))

# Write the tinted image back to disk
imsave('Doc/cat_tinted.jpg', img_tinted)
```



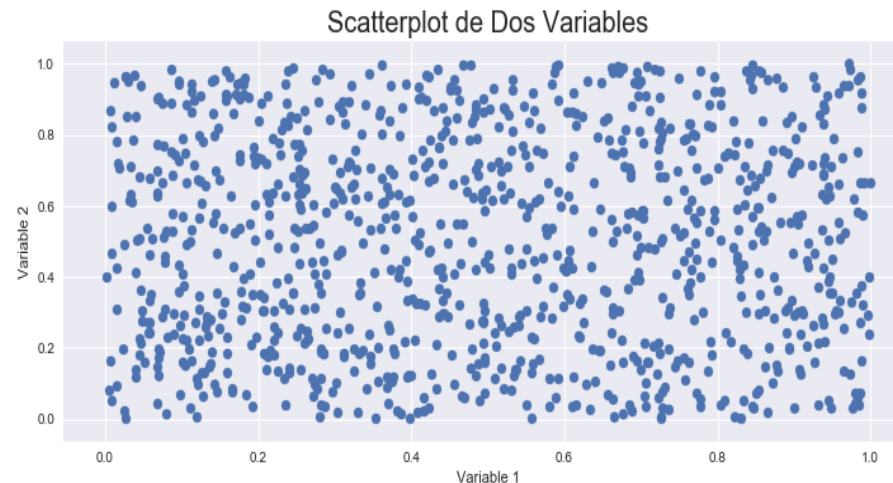
- Funciones Especiales
- Integración
- Optimización
- Interpolación
- Transformada de Fourier
- Procesamiento de Señales
- Álgebra Lineal
- Matrices Dispersas
- Grafos Dispersos
- Funciones Estadísticas
- Procesamiento de Imágenes

Matplotlib

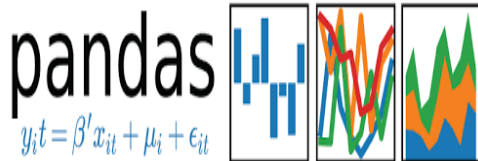


```
#Relación entre variables{'fontsize':  
rcParams['axes.titlesize'],  
plt.title("Scatterplot de Dos Variables",{'fontsize':20})  
plt.ylabel("Variable 2")  
plt.xlabel("Variable 1")  
plt.scatter(Datos.Var_1,Datos.Var_2)
```

Es una biblioteca que facilita la creación de gráficas desde listas o arreglos de NumPy.



Pandas



Pandas = Numpy +
Scipy+Matplotlib+..+..

Los dos objetos principales son Series y DataFrame, las funciones de los DataFrame en R es un subconjunto de todas las funcionalidades de las de Pandas.

Gracias Wes McKinney!!

```
In [10]: #Selección de Personas por condiciones  
DF_Personas[DF_Personas.Edad>28]
```

Out[10]:

	Altura	Ciudad	Edad	Nombres	Peso
2	1.72	México	35	Israel	85
3	1.60	Colombia	32	Alejandra	62

Scikit-learn



En el taller se hará usos principalmente de esta biblioteca!!!

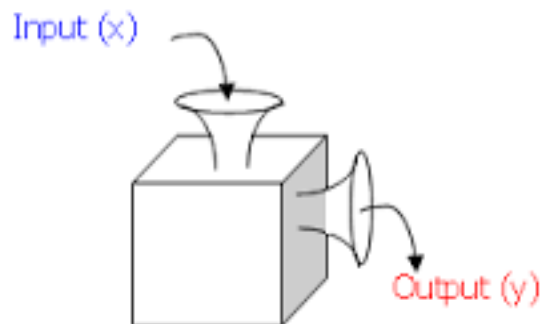
- Modelos Lineales Generalizados
- Modelos Lineales con Penalización
- Máquina de Soporte Vectorial
- Vecinos Cercanos
- Árboles y Ensamblados
- Naive Bayes
- Mezcla de Modelos Gaussianos
- Clustering
- Componentes Principales
- Variadas de Aprendizaje
- Validación Cruzada
- Selección de Parámetros
- Selección de Variables
- Pipeline

Aprendizaje Automático

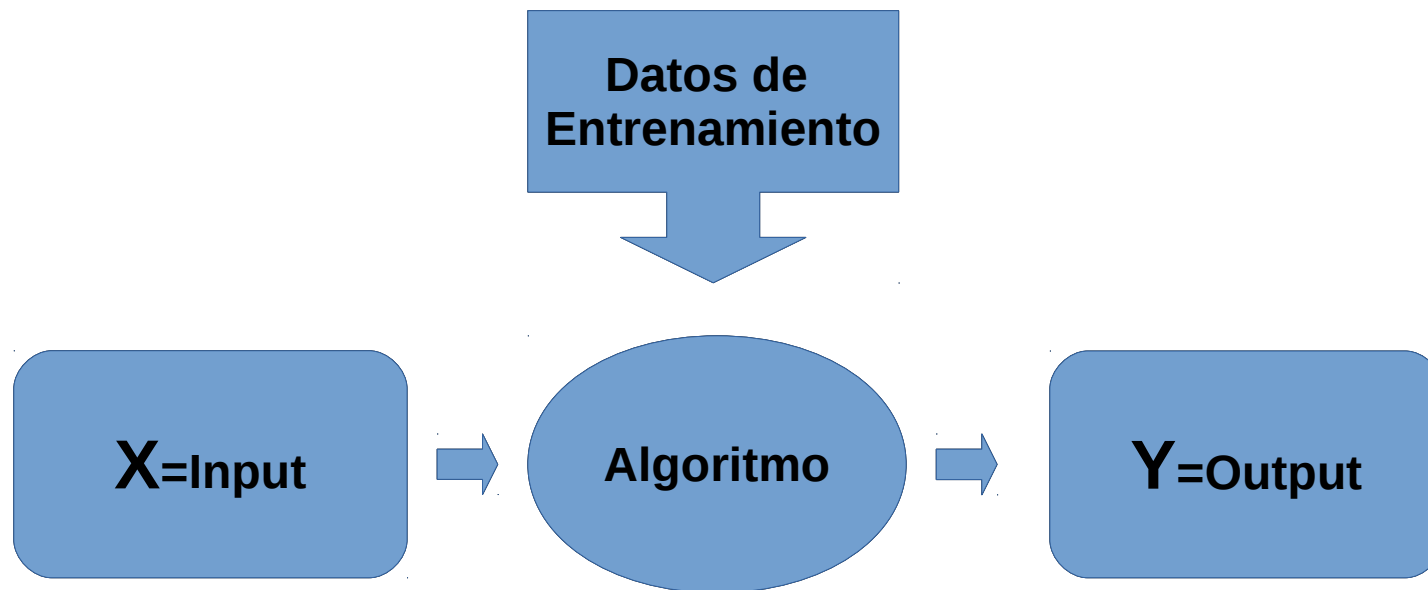
“Definimos Aprendizaje Automático como un conjunto de métodos que pueden detectar automáticamente patrones en los datos y luego utilizarlos para predecir alguna fecha o para realizar otro tipo de toma de decisiones bajo incertidumbre (como plantear como recopilar más datos)” **Kevin Murphy (Research Google)**

“Aprendizaje Automático es el campo de estudio que da a las computadoras la capacidad de aprender sin ser programadas explícitamente ” **Arthur Samuel IBM**

“Aprendizaje Automático es estadística, pero con técnicas de comprobación de modelos y con menos supuestos” **Brian D. Ripley Universidad de Oxford**



Aprendizaje Automático



$$X = h(Y)$$

Aprendizaje Automático

En el Taller solo veremos Aprendizaje Supervisado y No Supervisado

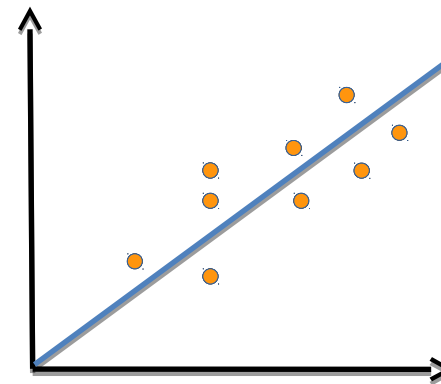
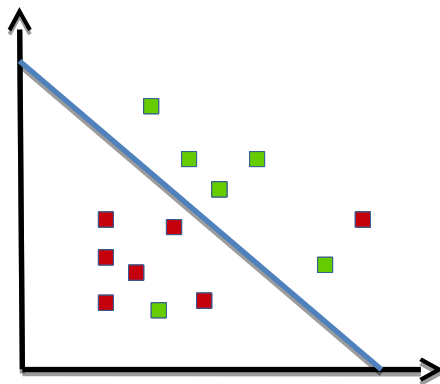
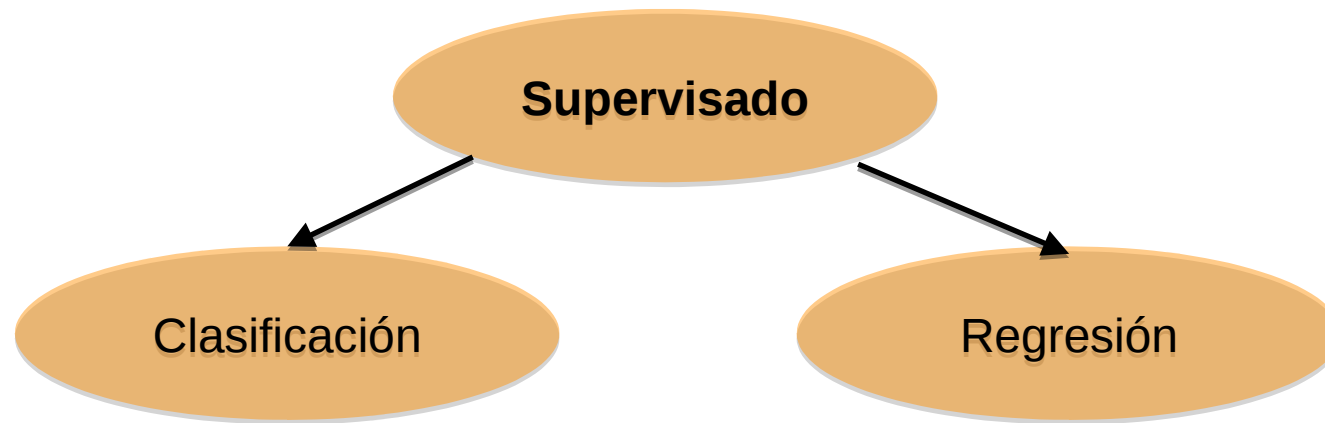


- Predicción de Eventos
- Detección de Fraudes
- Propensión a Impago

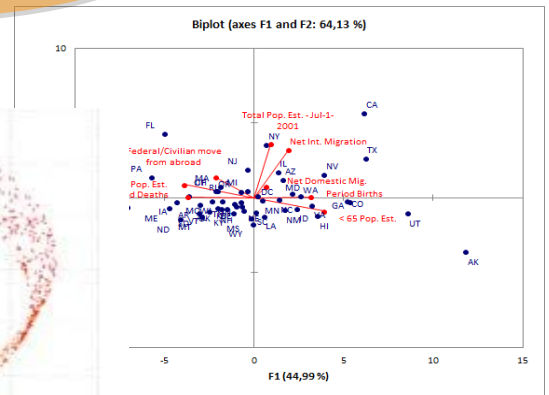
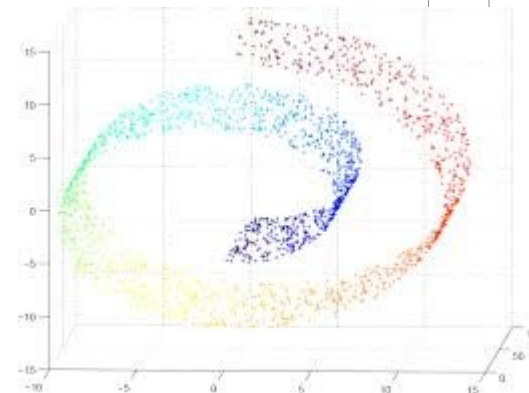
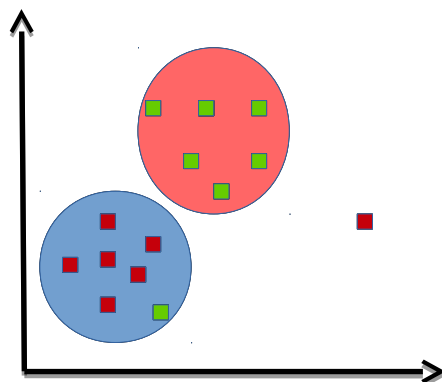
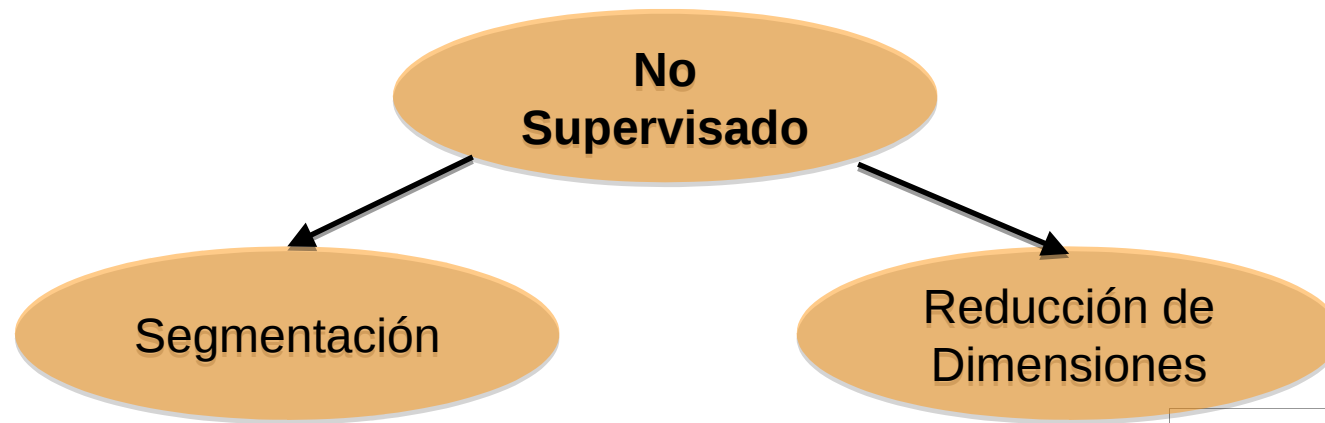
- Segmentación de Mercados
- Extracción de Patrones

- Robótica
- Juegos como: AlfaGo

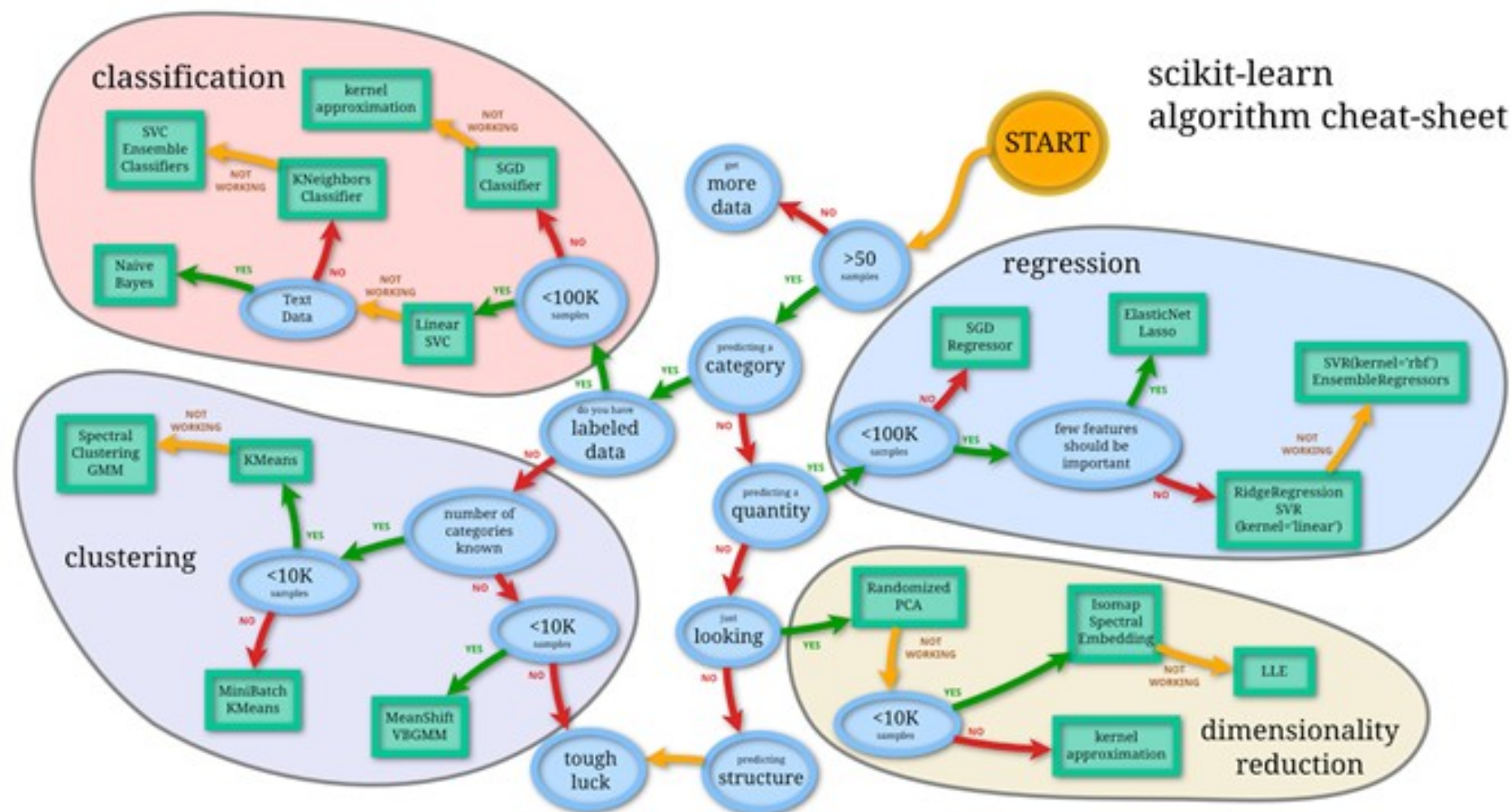
Aprendizaje Supervisado



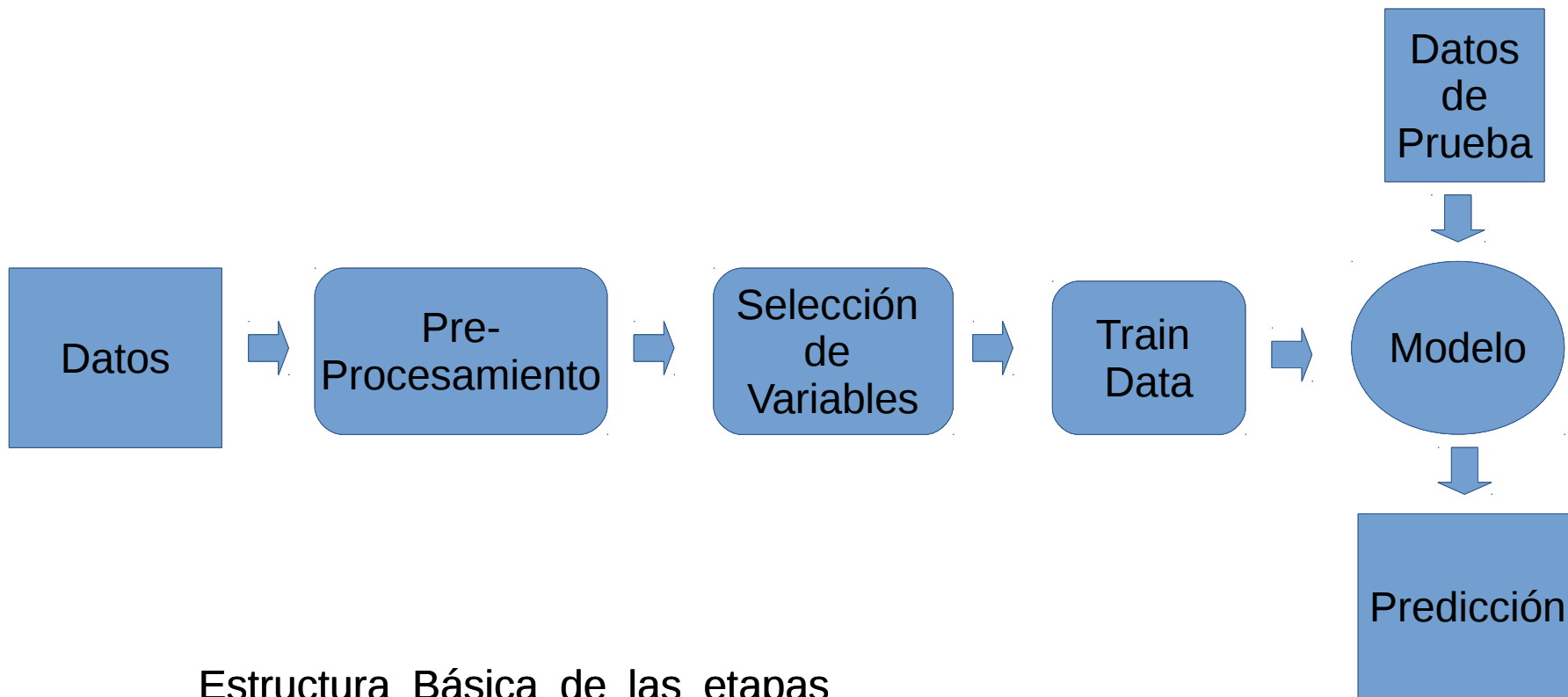
Aprendizaje No Supervisado



Panorama General

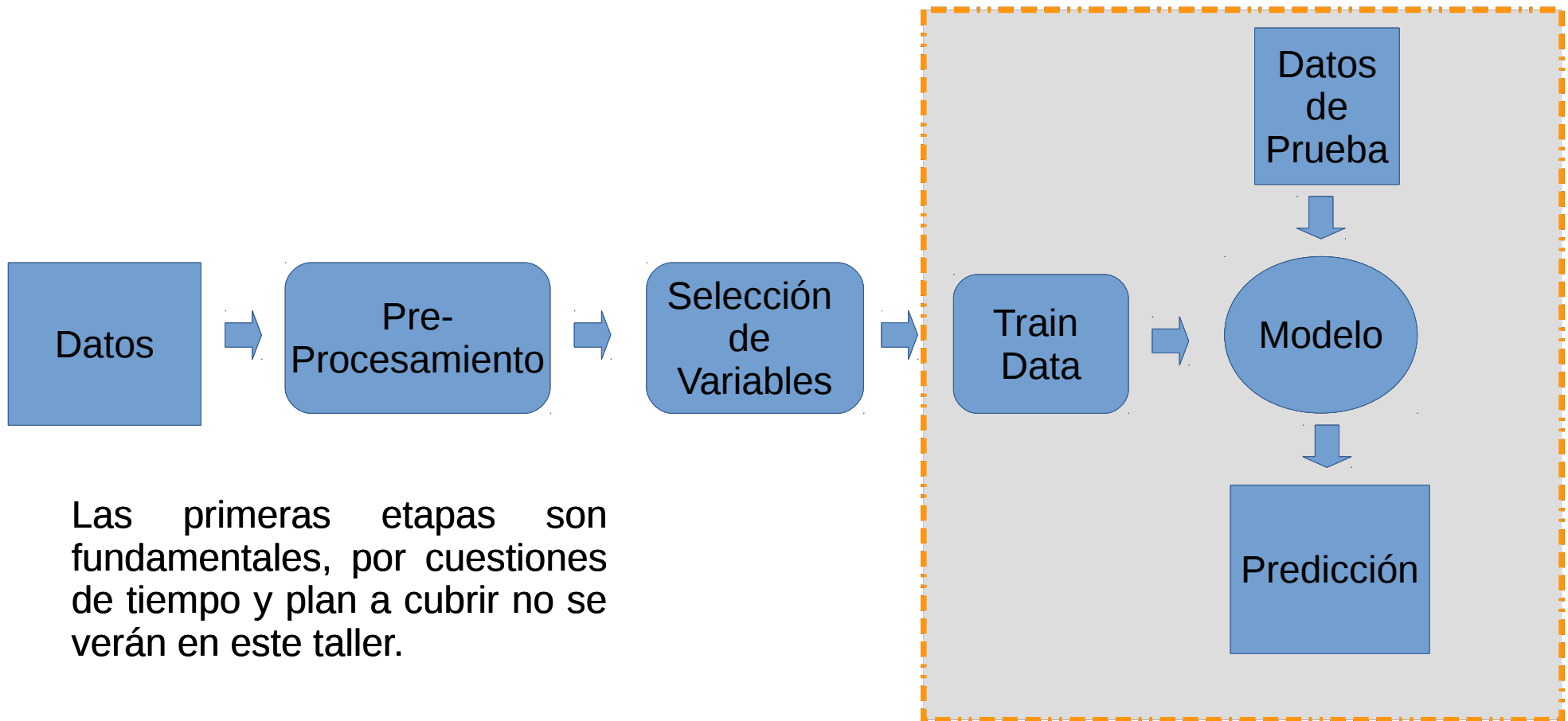


Pipeline Básico



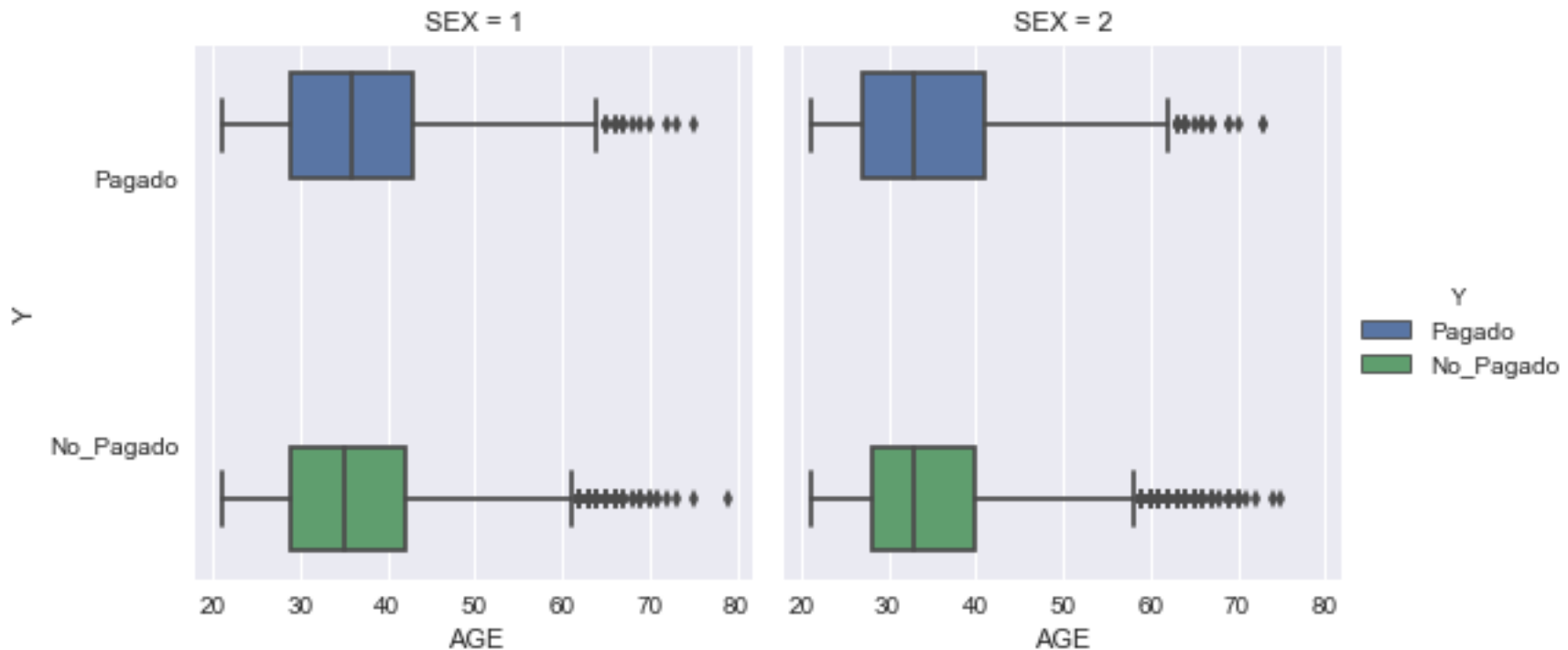
Estructura Básica de las etapas requeridas para un sistema que cuenta con Aprendizaje Automático.

En el taller



Las primeras etapas son fundamentales, por cuestiones de tiempo y plan a cubrir no se verán en este taller.

Aprendizaje Supervisado - Clasificación



Aprendizaje Supervisado - Regresión

