

Introducción al cálculo numérico y ciencia de datos

Clase 1

Mg. Víctor Viana

Tacuarembó – abril de 2023

Dr. Diego Passarella

Agenda

- Presentación del curso
- ¿Qué es el aprendizaje automático?
- El entorno Python
- Primer ejemplo



Presentación del curso

```
37     if path:
38         self.file = open(os.path.join(path, 'fingerprint.txt'), 'a')
39         self.fingerprints.add(fp)
40
41
42     @classmethod
43     def from_settings(cls, settings):
44         debug = settings.get('DEBUG')
45         return cls(job_dir=settings.get('JOB_DIR'))
46
47     def request_seen(self, request):
48         fp = self.request_fingerprint(request)
49         if fp in self.fingerprints:
50             return True
51         self.fingerprints.add(fp)
52         if self.file:
53             self.file.write(fp + os.linesep)
54
55     def request_fingerprint(self, request):
56         return self.request_fingerprint(request)
```



Docentes

- Diego Passarella (Profesor Agregado, Udelar)
- diego.passarella@cut.edu.uy
- Víctor Viana (Profesor Adjunto, Udelar)
- victor.viana@cut.edu.uy

Instituciones

- CENUR Noreste, Udelar
- Sede Tacuarembó
- Ingeniería Forestal



Objetivos

- Entender conceptos básicos
- Familiarizarse con la jerga
- Aplicar Metodología
- Fundamentos teóricos
- Desmitificar



Público

- Posgraduados
- egresados,
- investigadores,
- docentes,
- y estudiantes de diferentes disciplinas universitarias
- Profesores de Ciencias Básicas.

Alcance

¿Qué es?

- Aplicado
- Interactivo
- Introdutorio

¿Qué NO es?

- Exhaustivo
- Un curso de programación

Organización y temario

- 4 módulos, 5 semanas
- Módulo:
 - Una clase teórico-práctica (4h)
 - Una clase práctica (2h)
- Temario:
 - Introducción (este módulo).
 - Aprendizaje automático: supervisado y no supervisado.
 - Nociones básicas de cálculo numérico.
 - Aprendizaje profundo: Redes Neuronales.
 - Minería de datos. Reducción de dimensionalidad.
 - Clasificación: SVM y Árboles de Decisión.



Metodología y Evaluación

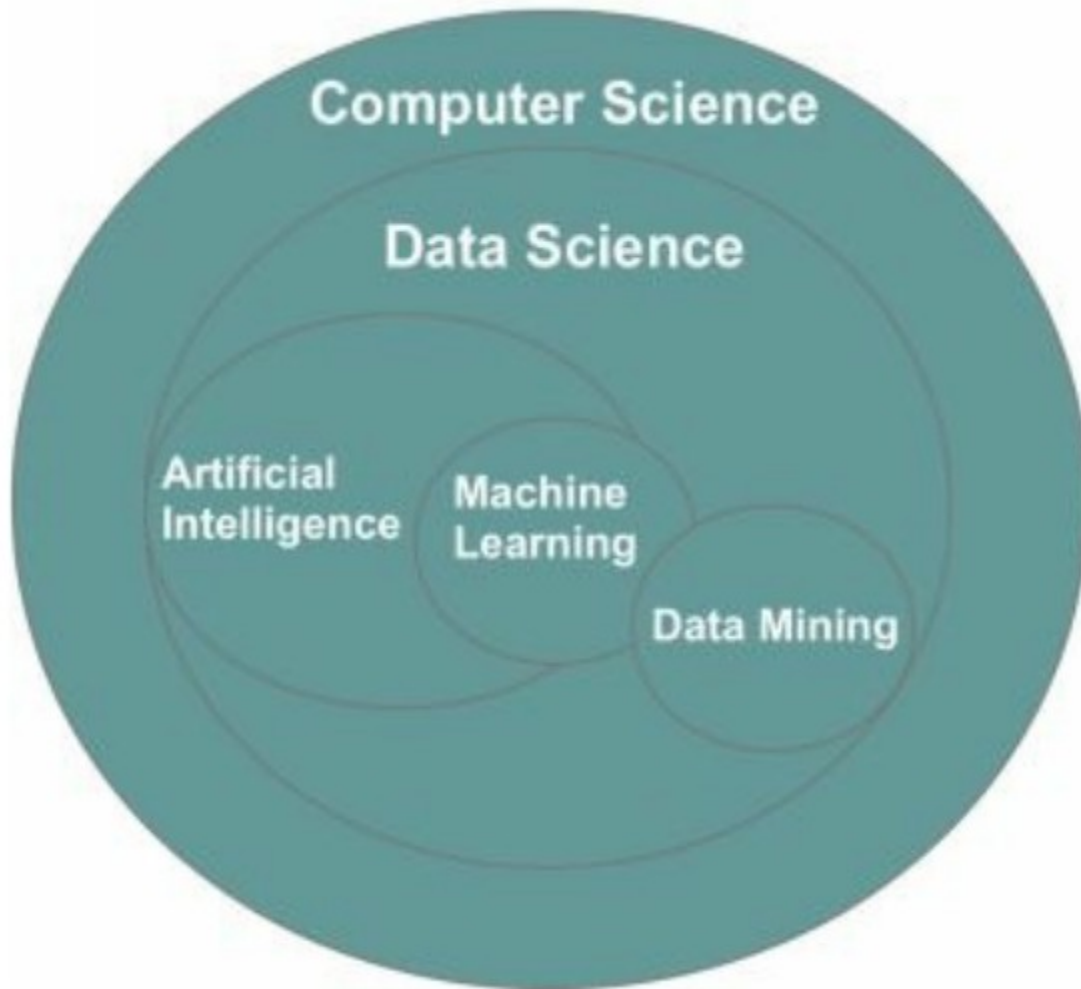
- Teórico + pequeños ejercicios y preguntas
- Asistencia al 75% de las clases
- Proyecto o evaluación final (a definir)
- Grupos de 2 a 3 personas



¿Qué es el aprendizaje automático?



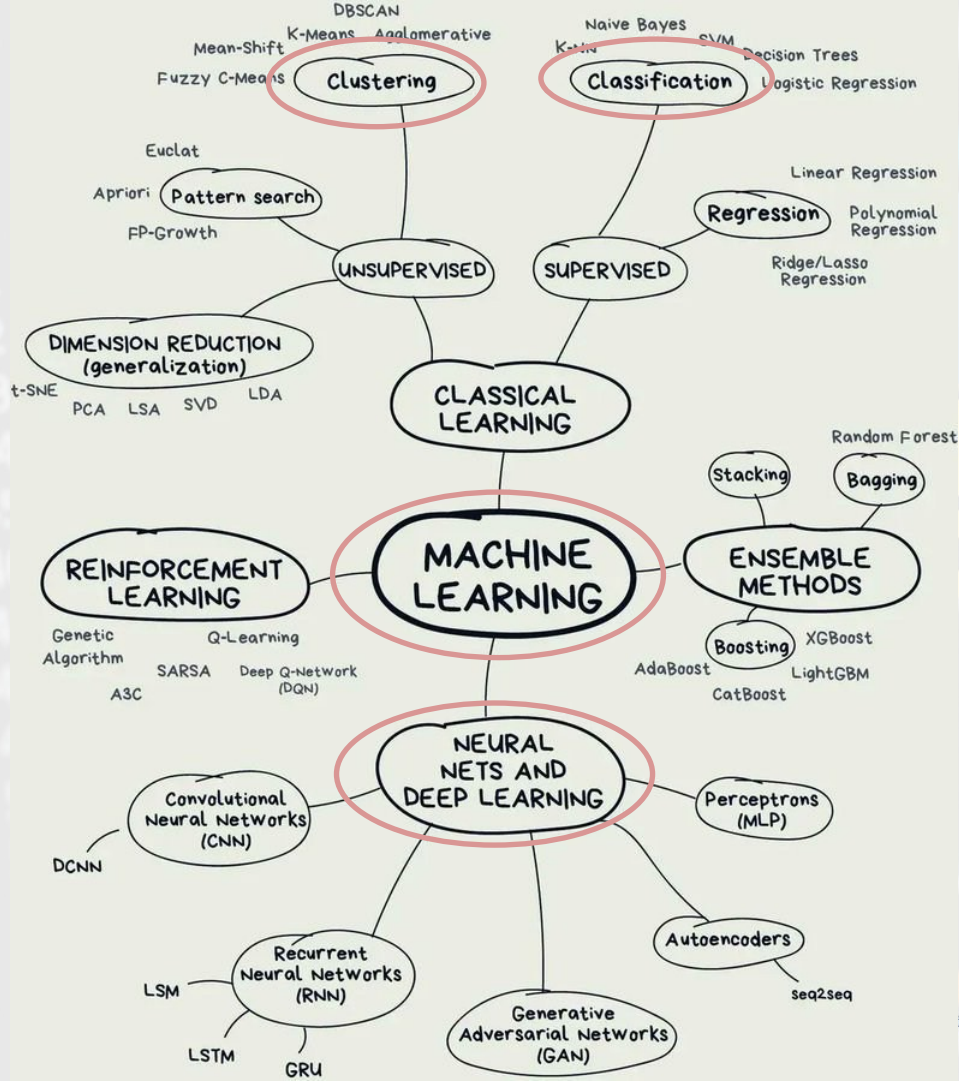
```
37     if path:
38         self.file = open(os.path.join(path, 'request_fingerprint.txt'), 'a')
39         self.file.seek(0)
40         self.fingerprints.update(request_fingerprint(request))
41
42     @classmethod
43     def from_settings(cls, settings):
44         debug = settings.get('DEBUG', False)
45         return cls(job_dir=settings.get('JOB_DIR', None))
46
47     def request_fingerprint(self, request):
48         fp = self.request_fingerprint(request)
49         if fp in self.fingerprints:
50             return True
51         self.fingerprints.add(fp)
52         if self.file:
53             self.file.write(fp + os.linesep)
54
55     def request_fingerprint(self, request):
56         request_fingerprint(request)
```



Extraído de:
Machine Learning for
Absolute Beginners, O.
Theobald (2017) 2nd Ed.

Machine Learning (Aprendizaje Automático)

- Es una rama de la inteligencia artificial (IA) que se enfoca en el desarrollo de algoritmos y modelos que puedan aprender a realizar tareas sin ser explícitamente programados.
- En lugar de seguir instrucciones precisas, los modelos de aprendizaje automático utilizan datos para aprender y mejorar con el tiempo.



Extraído de:
Machine Learning for
Absolute Beginners, O.
Theobald (2017) 2nd Ed.

Nombres, palabras clave

- Machine Learning
- Pattern Recognition
- Artificial Intelligence
- Statistical Learning
- Deep Learning
- Data Mining

¿Qué tienen en común esos términos?



```
37 if path:
38     self.file = open(os.path.join(path, filename), 'w')
39     self.file.write(data)
40     self.fingerprints.update({filename: data})
41
42 @classmethod
43 def from_settings(cls, settings):
44     debug = settings.get('DEBUG', False)
45     return cls(job_dir=settings.get('JOB_DIR', None))
46
47 def request_seen(self, request):
48     fp = self.request_fingerprint(request)
49     if fp in self.fingerprints:
50         return True
51     self.fingerprints.add(fp)
52     if self.file:
53         self.file.write(fp + os.linesep)
54
55 def request_fingerprint(self, request):
56     return request_fingerprint(request)
```

Data

- Datos
- Materia prima
- Fuente de información
- Objeto de estudio
- Recurso (de ahí, *minería*)

```
37 if path:
38     self.file = open(os.path.join(path, 'requirements.txt'), 'w')
39     self.file.write('')
40     self.fingerprints.update({req.name: req.version})
41
42 @classmethod
43 def from_settings(cls, settings):
44     debug = settings.get('DEBUG', False)
45     return cls(job_dir=settings.get('JOB_DIR', 'job_dir'))
46
47 def request_seen(self, request):
48     fp = self.request_fingerprint(request)
49     if fp in self.fingerprints:
50         return True
51     self.fingerprints.add(fp)
52     if self.file:
53         self.file.write(fp + os.linesep)
54
55 def request_fingerprint(self, request):
56     return request_fingerprint(request)
```



Learning

- Aprender
- Entrenar
- Extraer información del entorno
- Transformar en conocimiento
- Adquirir experiencia



Recognition

- Reconocer entorno
- Identificar patrones
- Identificar regularidad, estructura
- Aplicar conocimiento previo
- Organizar



Intelligence

- Aplicar conocimiento
- Extrapolar experiencia
- Adaptarse a nuevos escenarios



Taxonomía del aprendizaje automático

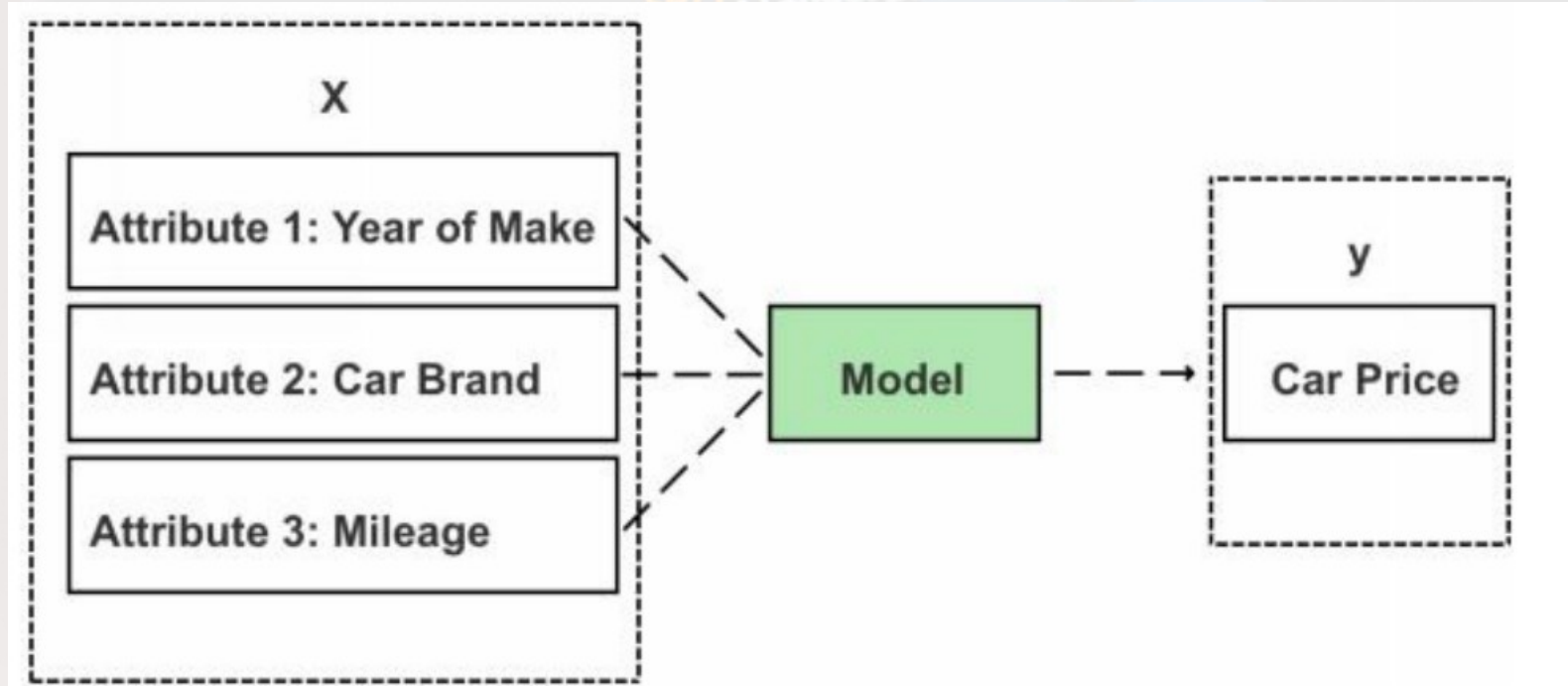
- Supervisado
- No supervisado
- Profundo (Deep Learning)
- Aprendizaje por refuerzos



Aprendizaje supervisado

- Se refiere a un enfoque de aprendizaje en el que el modelo se entrena utilizando un conjunto de datos etiquetados, es decir, cada ejemplo está etiquetado con una respuesta conocida.
- El objetivo del modelo es aprender una función que pueda predecir las etiquetas para nuevos ejemplos que no están en el conjunto de datos de entrenamiento.
- El modelo está "supervisado" durante el entrenamiento para aprender a producir salidas precisas a partir de entradas específicas.

Aprendizaje supervisado



Extraído de:
Machine Learning for Absolute
Beginners, O. Theobald (2017) 2nd Ed.

Aprendizaje supervisado

Ejemplo: clasificación

- Objetivo: determinar clase de cada muestra
- ¿A qué especie pertenece una flor?
- Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. Annals of eugenics, 7(2), 179-188.
- Se aprende con ejemplos previamente clasificados
- Se aplica conocimiento a nuevos ejemplos sin clasificar

Aprendizaje no supervisado

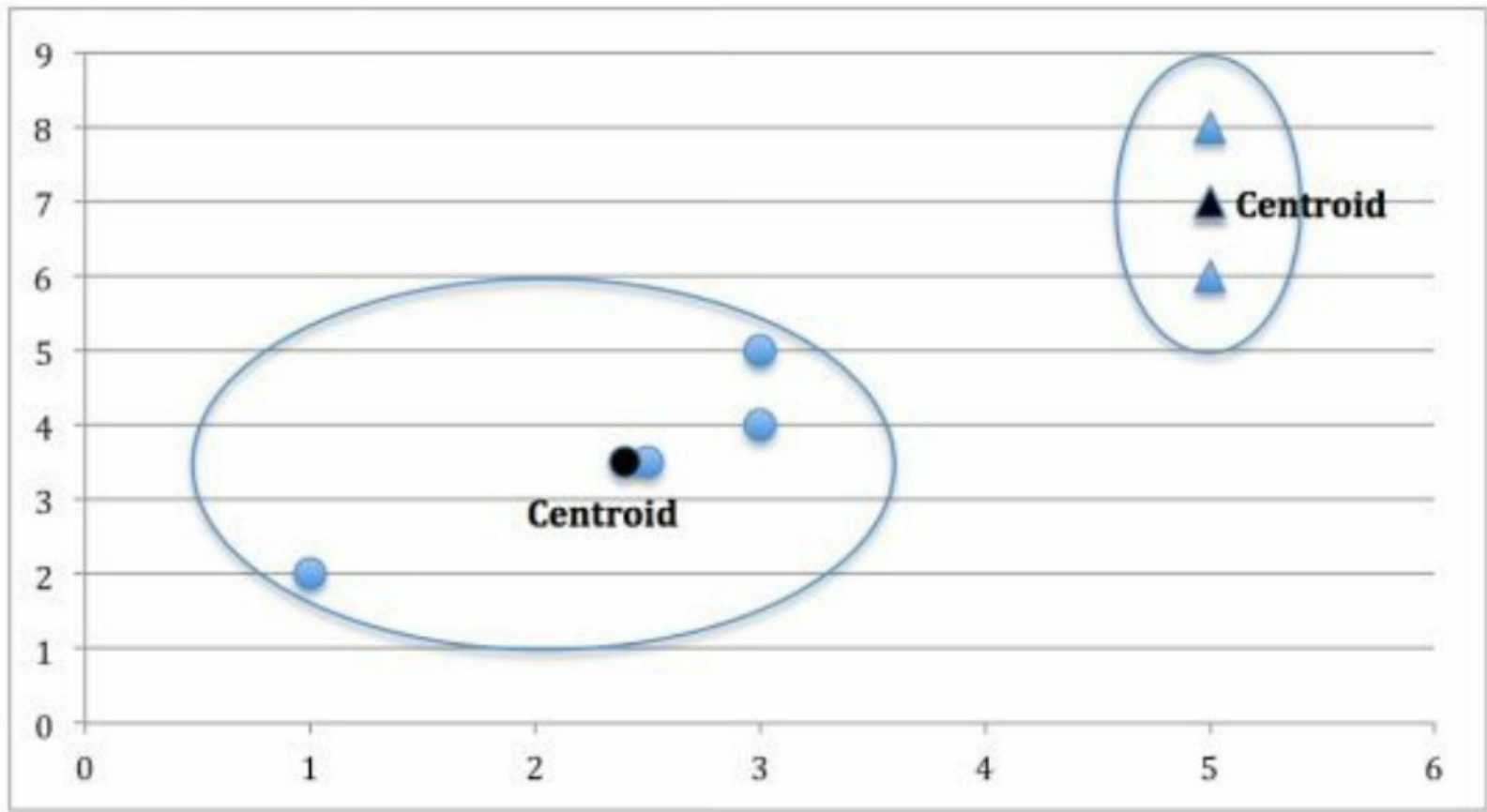
Ejemplo: agrupamiento

- No necesitamos etiquetas
- No necesitamos guía
- Organizamos por atributos
- Reconocemos similitudes
- ...o diferencias



Aprendizaje no supervisado

- Es un enfoque en el que el modelo se entrena utilizando un conjunto de datos sin etiquetar o con etiquetas parciales.
- El objetivo es encontrar patrones y estructuras ocultas dentro de los datos.
- No recibe ninguna retroalimentación explícita durante el entrenamiento y debe encontrar patrones por sí mismo.



Extraído de:
Machine Learning for Absolute
Beginners, O. Theobald (2017) 2nd Ed.

Aprendizaje no supervisado

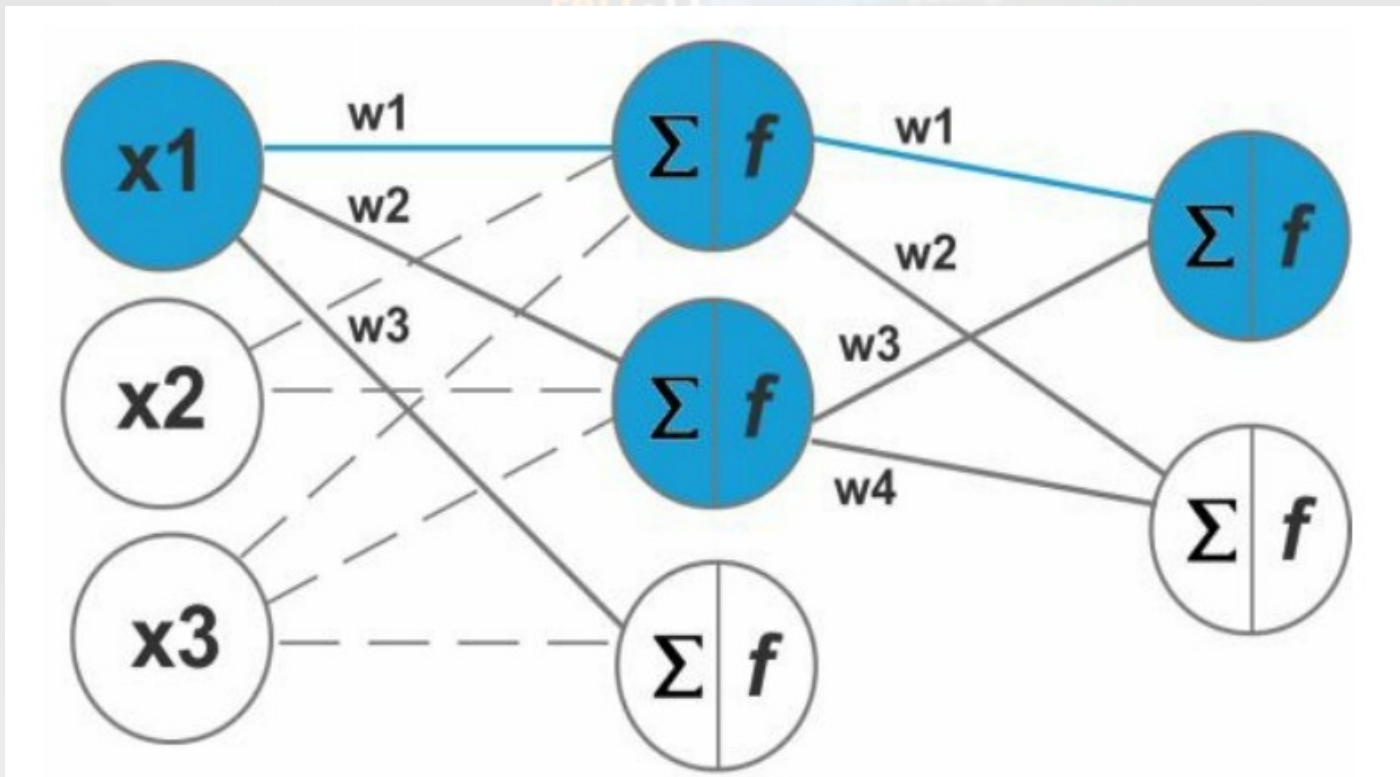
Ejemplo: agrupamiento

- No necesitamos etiquetas
- No necesitamos guía
- Organizamos por atributos
- Reconocemos similitudes
- ...o diferencias



Aprendizaje profundo

- Es una rama especializada del aprendizaje automático que se centra en el desarrollo de redes neuronales profundas (deep neural networks) para el aprendizaje automático.
- El aprendizaje profundo se utiliza para tareas que requieren el procesamiento de grandes cantidades de datos, como la visión por computadora y el procesamiento del lenguaje natural.



Extraído de:
Machine Learning for Absolute
Beginners, O. Theobald (2017) 2nd Ed.

Maquinaria clásica del aprendizaje automático

- **Preprocesamiento** de los datos
- Extracción y selección de características
- Modelado y ajuste de parámetros
- Inferencia y validación



Preprocesamiento

- Limpieza
- Rectificación
- Normalización
- Filtrar datos faltantes
- Filtrar datos anómalos



Extracción y selección de características

- Información relevante para inferencia
- Conocimiento a priori del problema
- Usualmente guiada por expertos
- Ejemplo (absolutamente tosco):
- “Para clasificar una planta según su hoja alcanza con tomar las medidas a, b y c”



Modelado

Sintetiza info sobre el problema

Puede ser:

Probabilístico (densidades)

Geométrico (regiones del espacio)

Analítico (funciones)

Grafos, redes

Árboles de decisión

Otros...

Entrenamiento

ajuste de parámetros del modelo



Inferencia y validación

Inferencia

Decidir sobre nuevo dato

Validación (caso supervisado)

Medir desempeño en datos conocidos

Aplicaciones

- Una vez que se entrena el modelo, se puede utilizar para hacer predicciones o tomar decisiones basadas en nuevas entradas que no se encuentran en el conjunto de datos de entrenamiento.
- El modelo se mejora continuamente a medida que se le proporcionan más datos y retroalimentación.

Entorno de trabajo Python



¿Qué es Python?

- Lenguaje de programación
- De propósito general
- Simple y claro

¿Por qué Python (y no R)?

- Sintaxis sencilla y consistente
- Menos abstracciones
- Muy flexible
- Enorme base de código (paquetes)
- Única alternativa para trabajar con redes neuronales modernas



Entorno para trabajo en ciencias

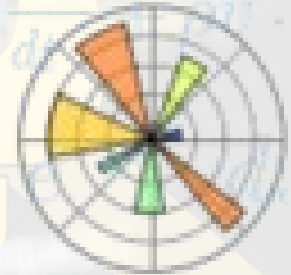
Numpy



Scipy



Matplotlib



SKLearn



Pandas



Jupyter Notebooks & Google Colaboratory

Jupyter Notebooks

- Entorno interactivo
- Interfaz web, en navegador
- Permite intercalar código y documentación
- Totalmente gratuito y libre
- Paquete de Python
- Pueden instalarse en cualquier computadora



Google Colab

- Servicio de Notebooks de Google
- Versión básica gratis
 - En muchos casos alcanza y sobra
 - Suficiente para nuestros objetivos
- Evita tener que instalar y configurar entorno en nuestra máquina!
- Corre en cualquier parte
- Se pueden bajar Notebooks generados



El lenguaje de programación Python

Notebook 1: [Introducción a Python y Colab](#)

- De propósito general
- De computación científica a desarrollo web
- No vamos a programar
- Vamos a hacer *scripts*
- Secuencias de comandos
- Lo básico:
 - Sintaxis
 - Semántica
 - Variables, funciones, módulos



Notebook 2: Matemáticas y Python

- Geometría y Álgebra
- Análisis y cálculo
- Probabilidad y estadística
- NumPy

Notebook 2:

[Matemática en Python](#)



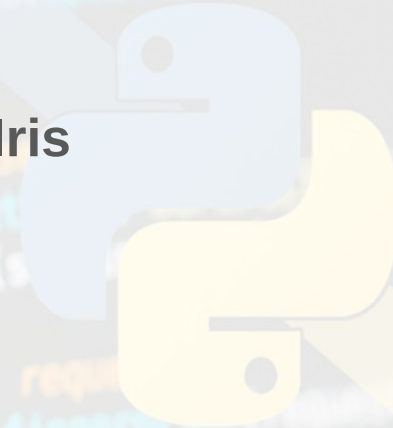
Primer ejemplo



```
37     if path:
38         self.file = open(os.path.join(path, 'request_fingerprint.txt'), 'a')
39         self.file.write(fp)
40         self.fingerprints.append(fp)
41
42     @classmethod
43     def from_settings(cls, settings):
44         debug = settings.get('DEBUG', False)
45         return cls(job_dir=settings.get('JOB_DIR', None))
46
47     def request_fingerprint(self, request):
48         fp = self.request_fingerprint(request)
49         if fp in self.fingerprints:
50             return True
51         self.fingerprints.add(fp)
52         if self.file:
53             self.file.write(fp + os.linesep)
54
55     def request_fingerprint(self, request):
56         request_fingerprint(request)
```

Problema

- Identificar especie en flores de la familia **Iris**
- Hay tres especies en este conjunto
- Setosa
- Versicolor
- Virginica





Iris setosa



Iris versicolor



Iris virginica

Datos

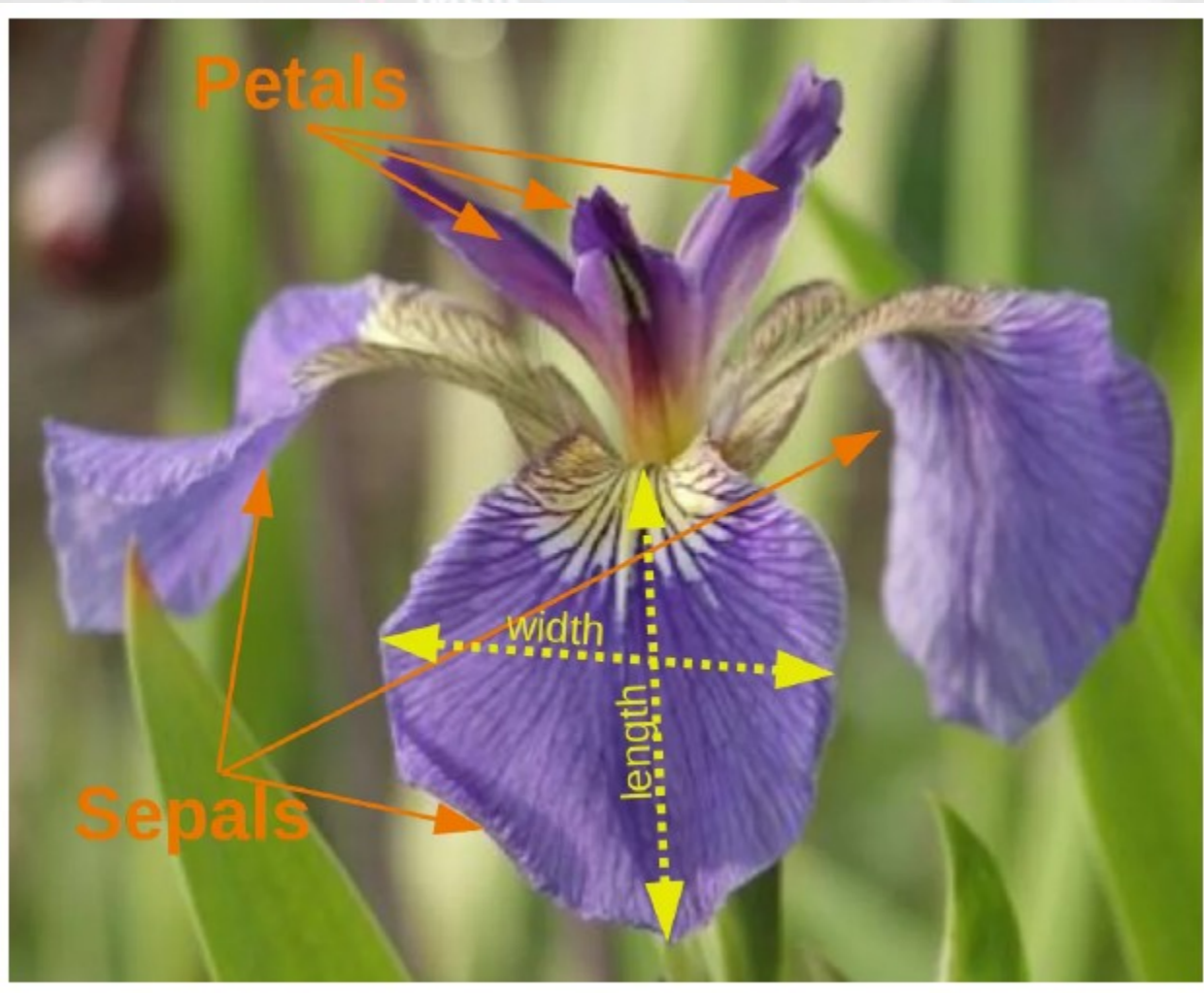
Cuatro medidas

- Largo del pétalo
- Ancho del pétalo
- Largo del sépalo
- Ancho del sépalo

Para entrenamiento

- Especie de la flor





R. A. Fisher (1936). *The use of multiple measurements in taxonomic problems*. *Annals of Eugenics* 7 (2): 179-188. doi:10.1111/j.1469-1809.1936.tb02137.x.

Largo de sépalo ↕	Ancho de sépalo ↕	Largo de pétalo ↕	Ancho de pétalo ↕	Especies ↕
5.1	3.5	1.4	0.2	<i>I. setosa</i>
4.9	3.0	1.4	0.2	<i>I. setosa</i>
4.7	3.2	1.3	0.2	<i>I. setosa</i>
4.6	3.1	1.5	0.2	<i>I. setosa</i>
5.0	3.6	1.4	0.2	<i>I. setosa</i>
5.4	3.9	1.7	0.4	<i>I. setosa</i>
4.6	3.4	1.4	0.3	<i>I. setosa</i>
5.0	3.4	1.5	0.2	<i>I. setosa</i>
4.4	2.9	1.4	0.2	<i>I. setosa</i>

Método: vecino más cercano

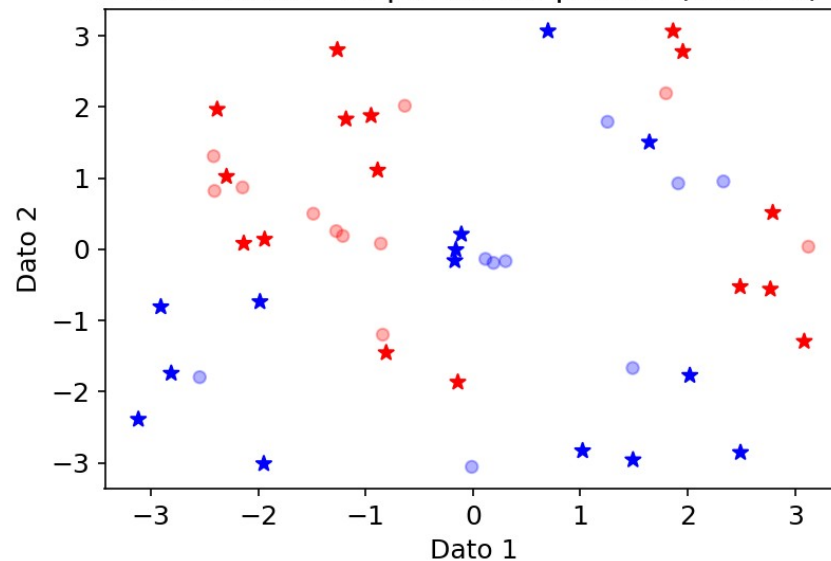
Entrenamiento

- Se almacenan todas las muestras del conjunto de entrenamiento

Predicción (K=1)

- Especie de la nueva muestra se imputa como la de la muestra más cercana en el espacio

Clasificación de puntos de prueba (estrella)



Primer ejemplo de aprendizaje automático

- Problema clásico:
- Iris (Fisher, 1936)
- Objetivo:
 - identificar especies de Iris
- Datos:
 - medidas de pétalos y sépalos
 - Cuatro datos por espécimen

Notebook 3:

[Aprendizaje automático](#)

