

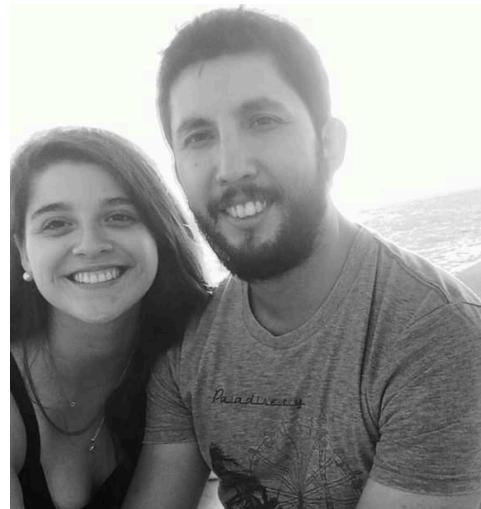
# TICS-411 Minería de Datos

Clase 0: Presentación del Curso

Alfonso Tobar-Arancibia

[alfonso.tobar.a@edu.uai.cl](mailto:alfonso.tobar.a@edu.uai.cl)

# ¿Quién soy?



- Alfonso Tobar-Arancibia, estudié **Ingeniería Civil** pero llevo 9 años trabajando como:
  - Data Analyst.
  - Data Scientist.
  - ML Engineer.
  - Data Engineer.
- Terminando mi Msc. y empezando mi PhD en la UAI.
- Me gusta mucho programar (en vivo).
- Contribuyo a **HuggingFace** y **Feature Engine**.
- He ganado 2 competencias de Machine Learning.
- Publiqué mi primer [paper](#) el año pasado sobre Hate Speech en Español.
- Juego **Tenis de Mesa**, hago **Agility** con mi perrita Kira y escribo en mi Blog.

**Cuéntenme de ustedes!!**

# Sobre el Curso

# Objetivos del Curso



- Identificar Elementos Claves del Machine Learning (Terminología, Nomenclatura, Intuición).
- Entender como interactúan los algoritmos más importantes.
- Aprender a seleccionar el mejor Algoritmo para el Problema.
- Ejecutar y aplicar algoritmos clásicos de Machine Learning.
- Evaluar el desempeño esperado del Modelo.

# Tópicos

- Introducción a la Minería de Datos
  - Análisis Exploratorio de Datos (EDA)
  - Modelos No Supervisados/Descriptivos
  - Modelos Supervisados/Predictivos

## Modelos no Supervisados

- K-Means
- Hierarchical Clustering
- DBScan
- Apriori

## Modelos Supervisados

- KNN
- Árboles de Decisión
- Naive Bayes
- Regresión Logística

# Sobre las clases

- Clases presenciales, con participación activa de los estudiantes.
- Es un curso coordinado.
- Canal oficial será **Webcursos**.
- Mucha terminología y material de estudio será en **Inglés**.
- Horario: Jueves.
  - 15:30 a 16:40 (Cátedra)
  - 17:00 a 18:10 (Práctico)
  - Idealmente!!
- Asistencia es voluntaria, pero **altamente recomendada**.

# Materiales de Clases

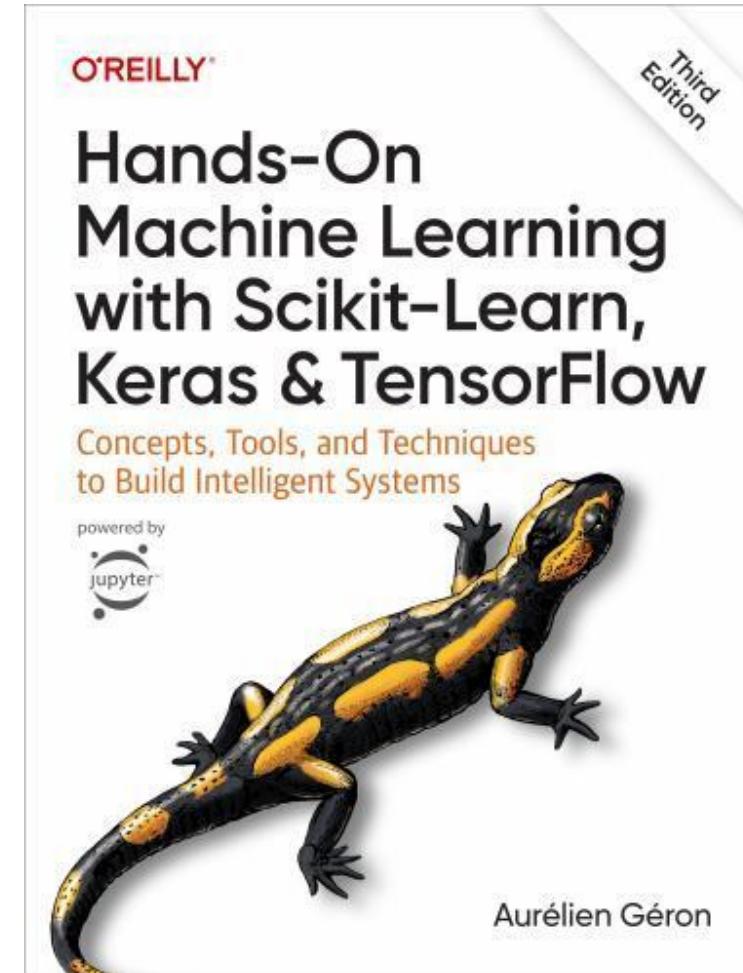
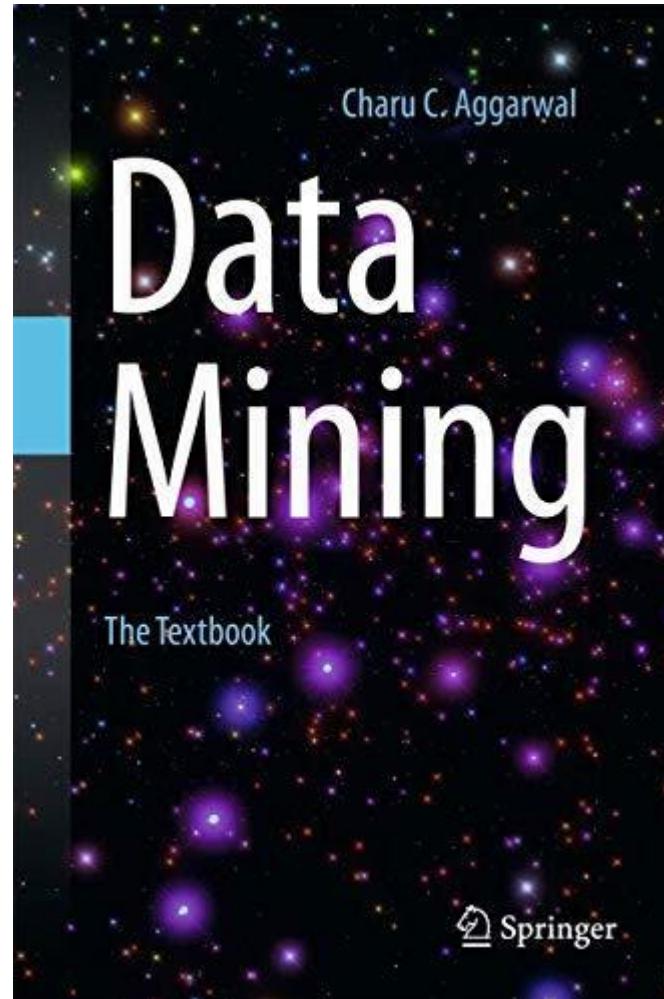
- Diapositivas
- Prácticos

-  • Slides interactivas (Código se puede copiar e imágenes se pueden ver en grande).  
• Se puede buscar contenido en las diapositivas mediante un buscador.  
• Se dejarán copias en PDF en Webcursos (levemente distintas).

 Se espera que los estudiantes dominen las siguientes tecnologías:

- Python
- Google Colab
- Pandas/Numpy
- Scikit-Learn (Se enseñará a lo largo del curso).

# Material Complementario



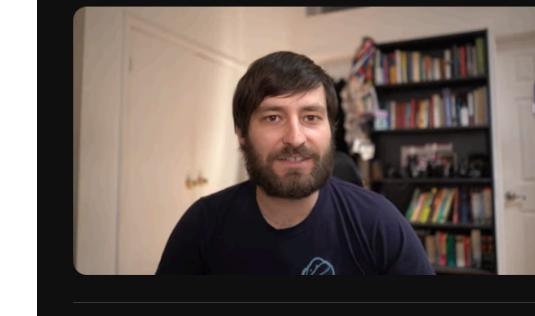
## Curso de Scikit-Learn

Channel Intro - Applied Machine Learning  
43,041 views • 3 years ago

Welcome!  
Check out the Applied Machine Learning 2020 playlist.  
The class website is here:  
<https://www.cs.columbia.edu/~amuller...>

Jake's Data Science Handbook is here:  
<https://jakevdp.github.io/PythonDataS.....>

[READ MORE](#)



- Tutorial Colab
- Agregar Datos Externos a Colab

# Evaluación

- Dos Evaluaciones Escritas (P1, P2) coordinadas y cuatro tareas prácticas en parejas (T1, T2, T3, T4)

$$NP = 0.35 \cdot P1 + 0.35 \cdot P2 + 0.3 \cdot \bar{T}$$

$$\bar{T} = (T1 + T2 + T3 + T4)/4$$

 Si  $NP > 5$

$$NF = NP$$

 En caso contrario:

$$NF = 0.7 \cdot NP + 0.3 \cdot E$$

# Ayudantías

Ayudante: TBD

email: TBD



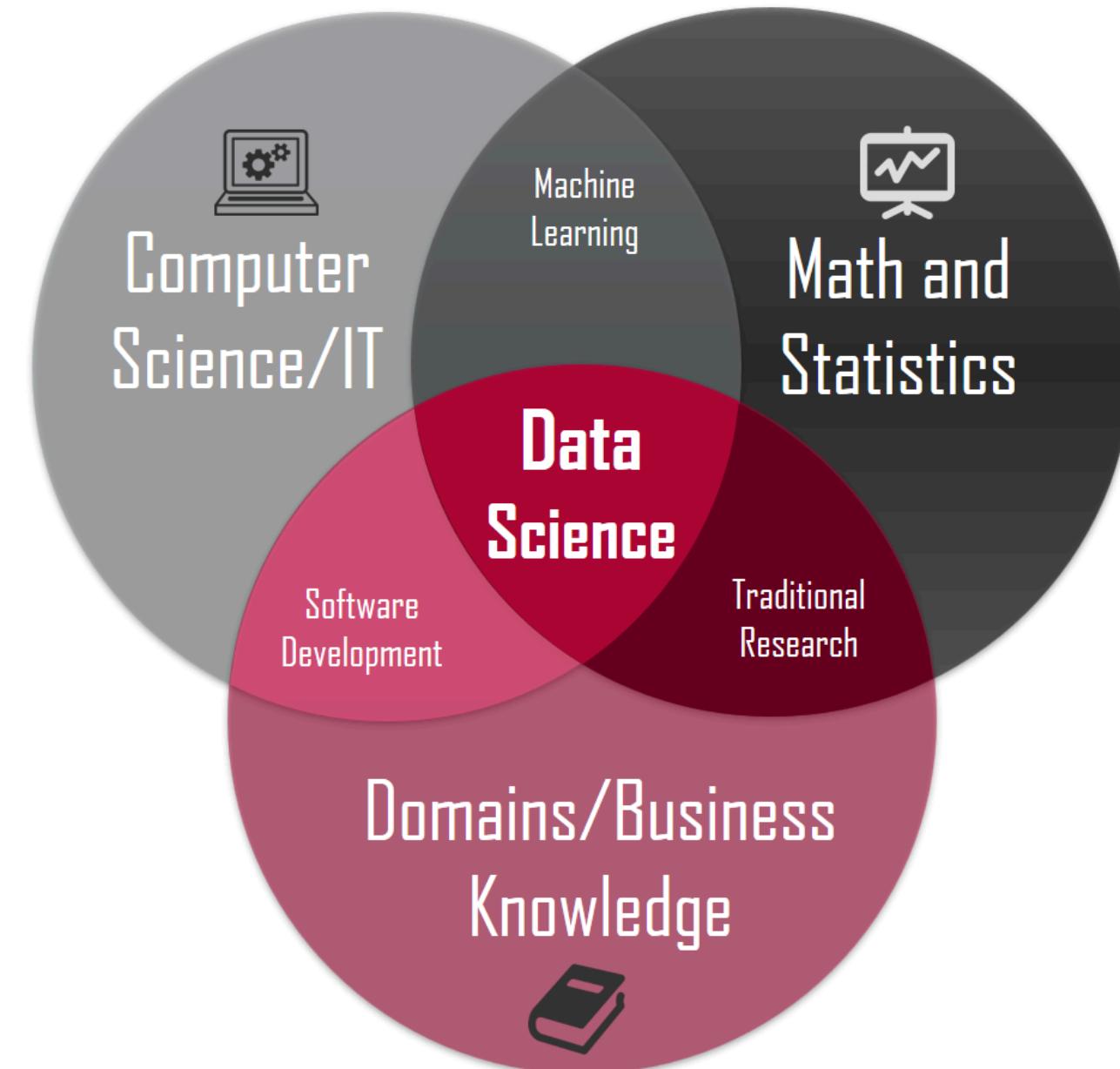
- Las ayudantías serán en la manera que sean necesarias.
- Estarán enfocadas principalmente en aplicaciones y código.

# Introducción al Curso

# Revolución de los Datos



# Nace el Data Science (Ciencia de Datos)



# ¿Cómo aprovechar la información que tenemos?

## Data Mining (Minería de Datos)

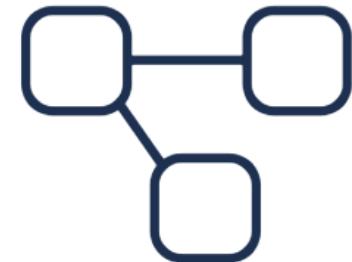
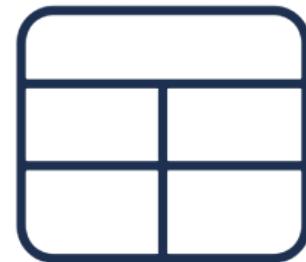
“The process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data.” (Fayyad, Piatetsky-Shapiro & Smith 1996)

## Machine Learning (Aprendizaje Automático)

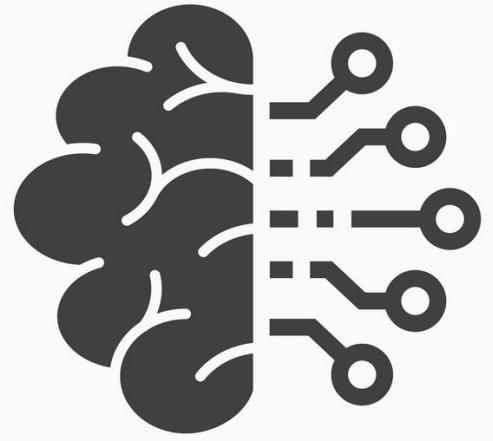
“A computer program is said to learn from experience **E** with respect to some class of tasks **T** and performance measure **P**, if its performance at tasks in **T**, as measured by **P**, improves with experience **E**.” (Mitchell, 2006)

# Tipos de Datos

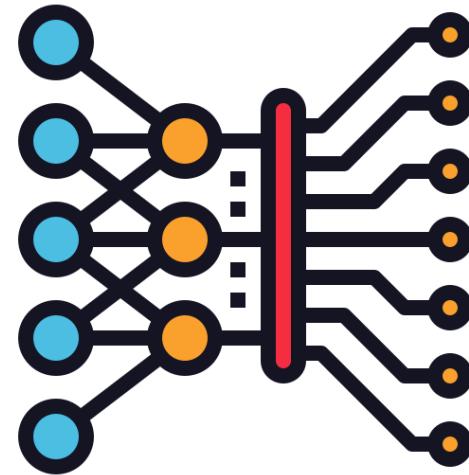
## Datos Estructurados



## Datos No Estructurados



MACHINE  
LEARNING



# Tipos de Datos: Datos Tabulares

	siteEvt	site	time	ml	depth	dist	azmth	lat	lon	peak	evid	epoch	Id
0	2431	1	2011-07-16 07:39:00 (UTC)	2.99	14.8	34.88	168	32.791	-115.453	10019300.0	15018412	1310801940	2431
1	4028	1	2010-04-06 03:09:53 (UTC)	3.05	10.0	88.57	171	32.312	-115.380	610667.0	14616764	1270523393	4028
2	7393	1	2008-01-18 09:52:29 (UTC)	2.70	19.0	47.68	164	32.686	-115.389	1863890.0	14343464	1200649949	7393
3	12447	1	2005-09-01 01:38:23 (UTC)	2.01	4.6	11.20	315	33.169	-115.615	13613300.0	14178480	1125538703	12447
4	7284	1	2009-12-30 18:48:57 (UTC)	5.80	6.0	77.43	156	32.464	-115.189	736620000.0	14565620	1262198937	7284



- Filas: Observaciones, instancias, registros. (Normalmente independientes).
- Columnas: Variables, Atributos, Features.



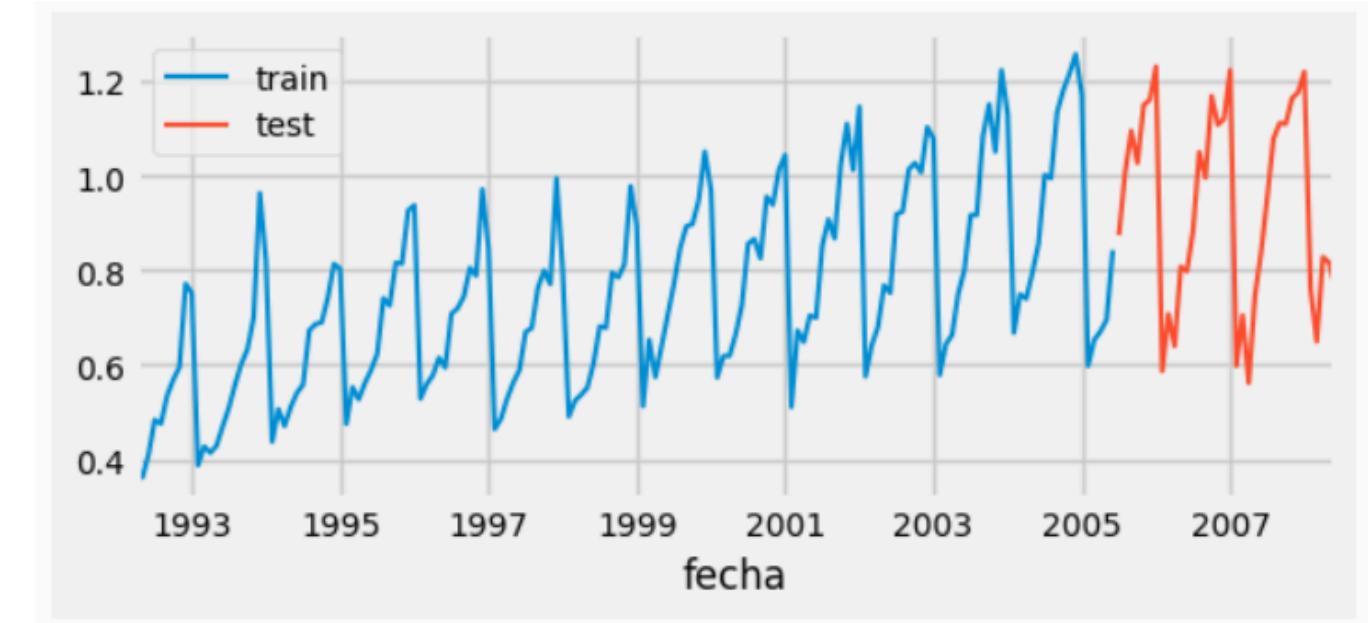
- Probablemente el tipo de datos más amigable.
- Requiere conocimiento de negocio (**Domain Knowledge**)



- Es un % bajísimo del total de datos existentes en el Mundo. También el que más disponible está en las empresas.
- Distintos **data types**, por lo que normalmente requiere de algún tipo de **preprocesamiento**.

# Tipos de Datos: Series de Tiempo

	y	exog_1	exog_2
fecha			
1992-04-01	0.379808	0.958792	1.166029
1992-05-01	0.361801	0.951993	1.117859
1992-06-01	0.410534	0.952955	1.067942
1992-07-01	0.483389	0.958078	1.097376
1992-08-01	0.475463	0.956370	1.122199

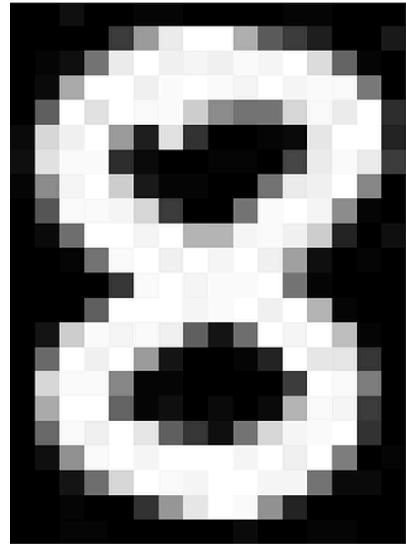


- Filas: Instancias temporales (Normalmente interdependientes).
- Columnas: Variables, Atributos, Features (Univariada o Multivariada).



- Es un % bajísimo del total de datos existentes en el Mundo.
- Propiedad temporal requiere **preprocesamiento** y modelos especiales.

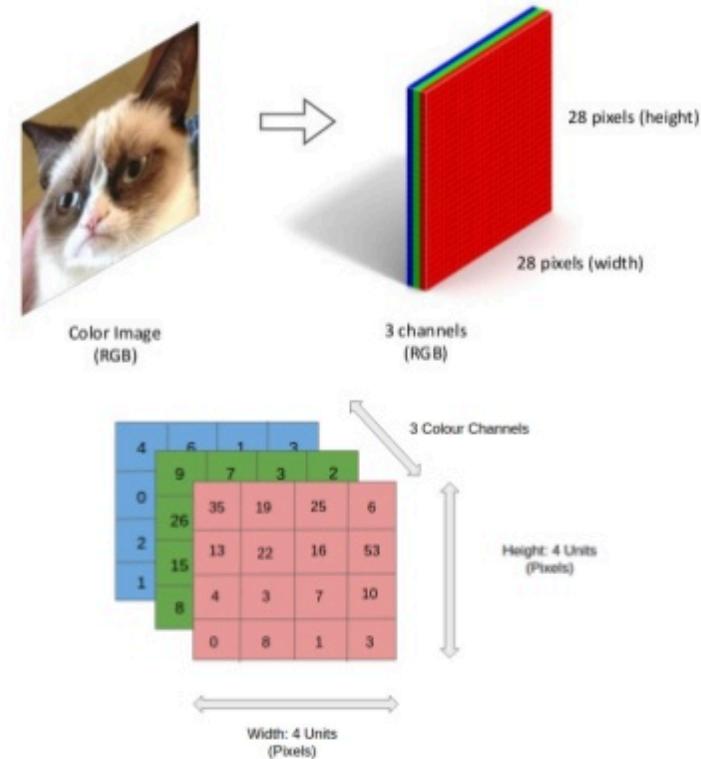
# Tipos de Datos: Imágenes



0	2	15	0	0	11	10	0	0	0	9	9	0	0	0
0	0	4	60	157	236	255	177	95	61	32	0	0	29	
0	10	16	119	238	255	244	245	243	250	249	255	222	103	10
0	14	170	255	255	244	254	255	234	245	249	253	251	124	1
2	98	255	228	255	251	254	211	141	116	122	215	251	238	49

13	217	243	255	155	33	226	52	2	0	10	15	232	255	36
16	229	252	254	49	12	0	0	7	7	0	70	237	252	62
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6
0	13	113	255	245	255	182	181	248	252	242	208	36	0	19

color image is 3rd-order tensor



- Este es el tipo de Datos que disparó la Inteligencia Artificial.
- ¿Cuántos computadores para identificar un Gato? 16,000

# Tipos de Datos: Texto Libre



- Datos Masivos.
- Difíciles de lidiar ya que deben ser llevarse a una representación numérica.
- Alto nivel de Sesgo y Subjetividad.



- Gracias a este tipo de datos se han producido los avances más increíbles del último tiempo: [Transformers](#)

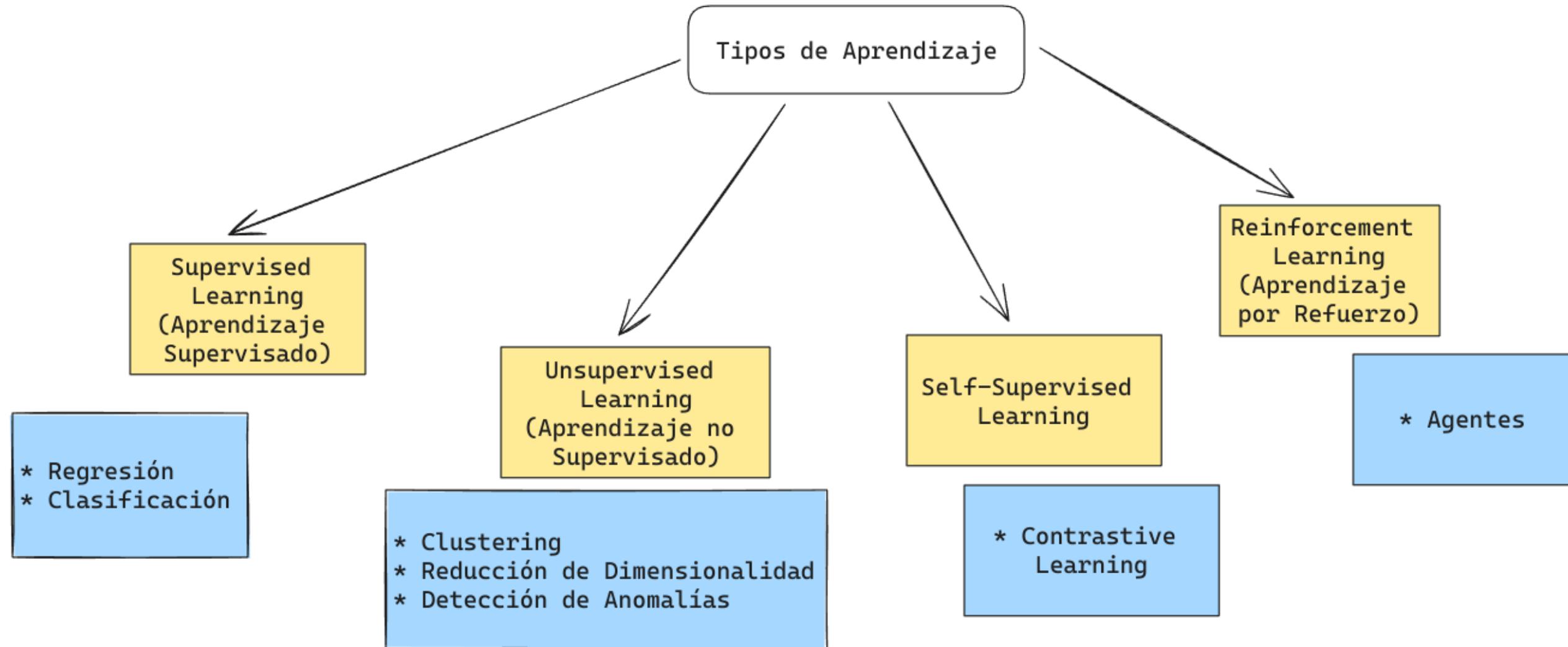
# Tipos de Datos: Videos



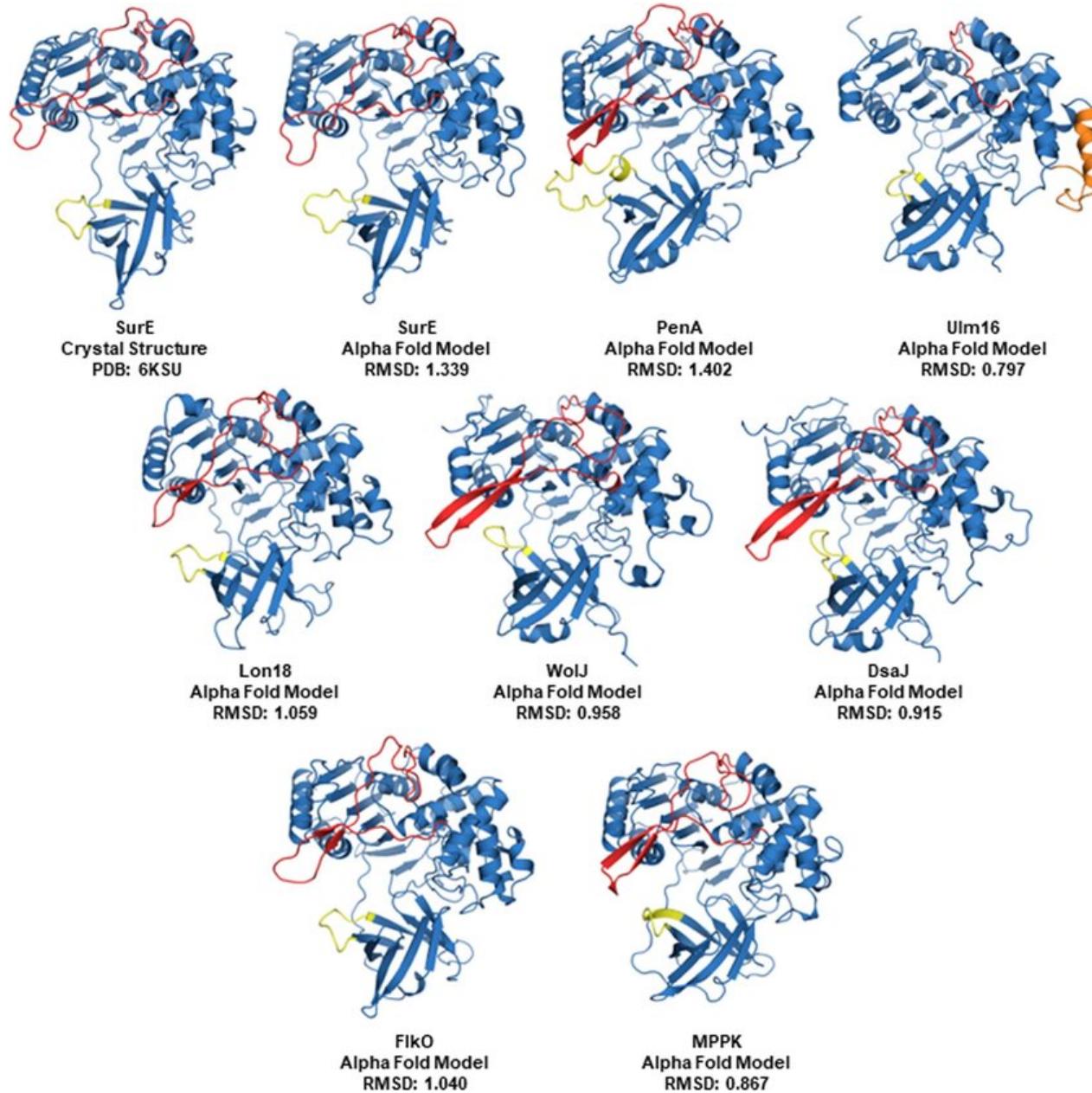
- Los videos no son más que arreglos de imágenes.
- Son un tipo de dato muy pesado y difícil de lidiar.
- Requiere alto poder de Procesamiento.

# ¿Cómo aprenden los Modelos?

# Tipos de Aprendizaje



# Reinforcement Learning



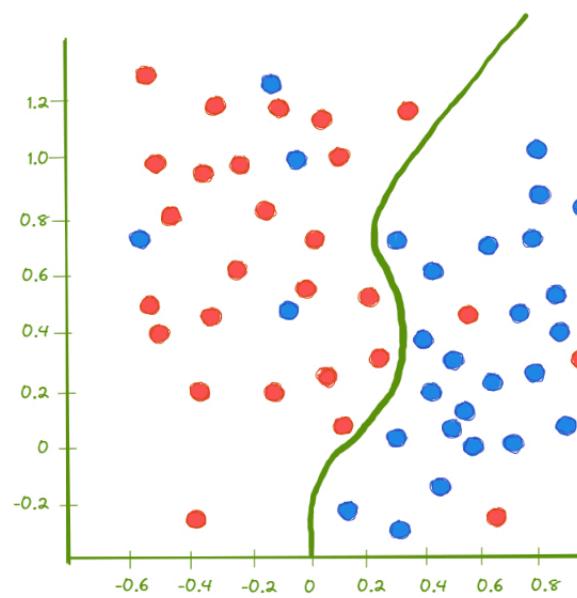
En este tipo de aprendizaje se enseña por refuerzo. Es decir se da una recompensa si el sistema aprende lo que queremos.

Si el premio es mayor, se pueden obtener aprendizajes mayores.

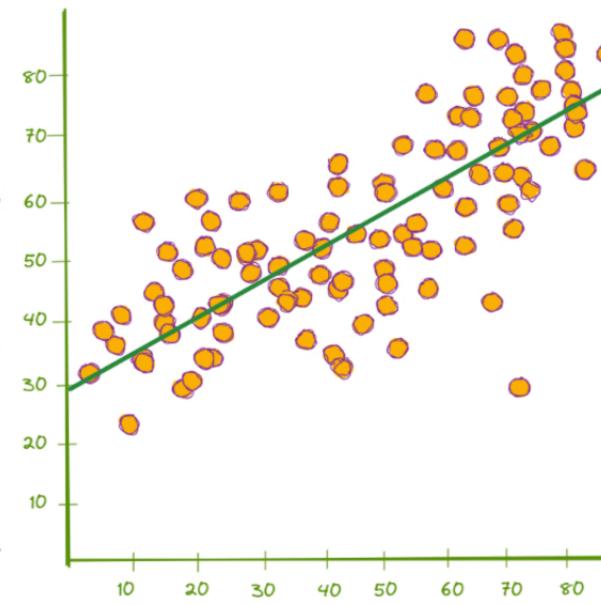
Un ejemplo de esto es **AlphaTensor** en el cual un modelo **aprendió** una nueva manera de multiplicar matrices que es más eficiente.

Otro ejemplo es **AlphaFold** donde el modelo **aprendió/descubrió** cómo se doblan las proteínas cuando se vuelven aminoácidos.

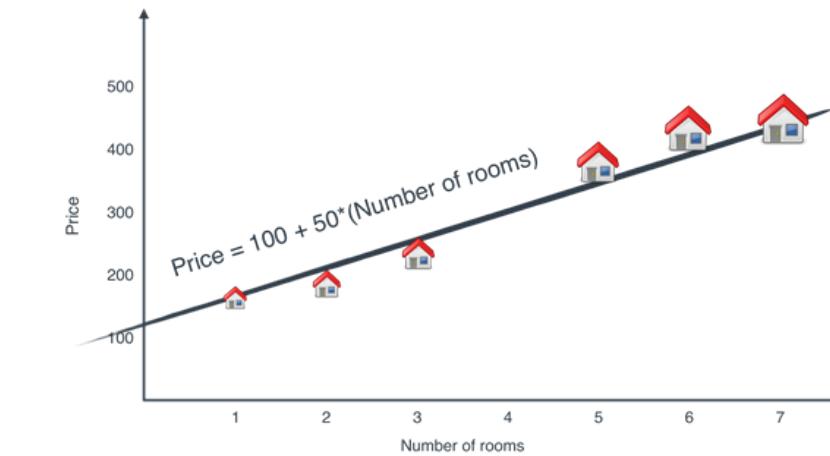
# Problemas Supervisados: Regresión y Clasificación



classification



regression

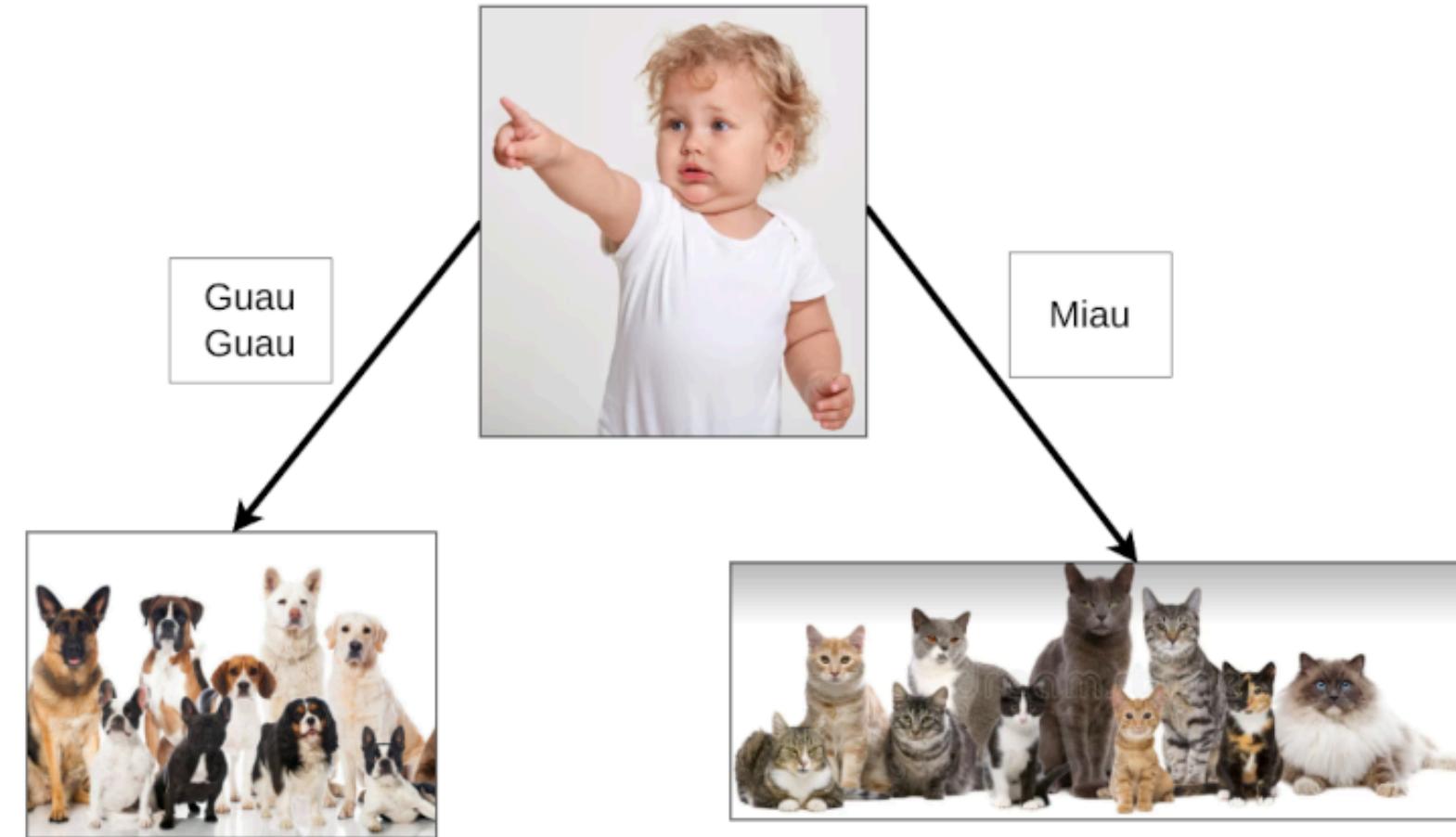


- 💡 • Regresión: Se busca estimar un valor continuo.
  - (Estimar el valor de una casa).
- Clasificación: Se busca encontrar una categoría o un valor discreto.
  - (Clasificar una imagen como Perro o Gato).

- ❗ • Para entrenar este tipo de modelos se necesitan **etiquetas**, es decir, la respuesta esperada del modelo.



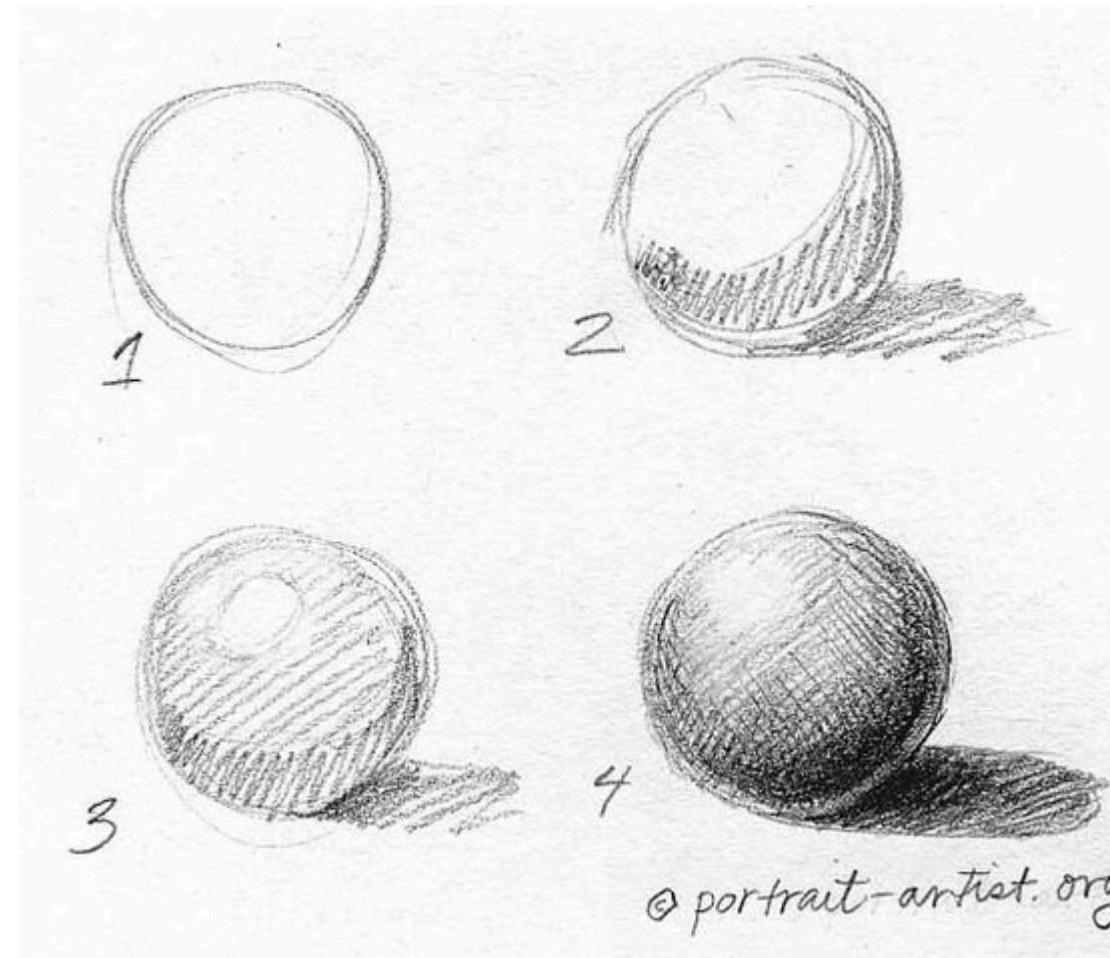
# Clustering



- 💡 • Clusters: Una categoría en la que sus componentes son similares. Los clusters normalmente no tienen un nombre propio, sino que uno les asigna uno.
- También se les llama segmentos. No usar la palabra **clase**.

- ⚠ • No requiere de etiquetas, por lo tanto, no es posible evaluar su desempeño de manera 100% acertada.

# Reducción de Dimensionalidad



- Reducción de la Dimensionalidad: Eliminar complejidad sin perder información clave para poder entender su comportamiento.

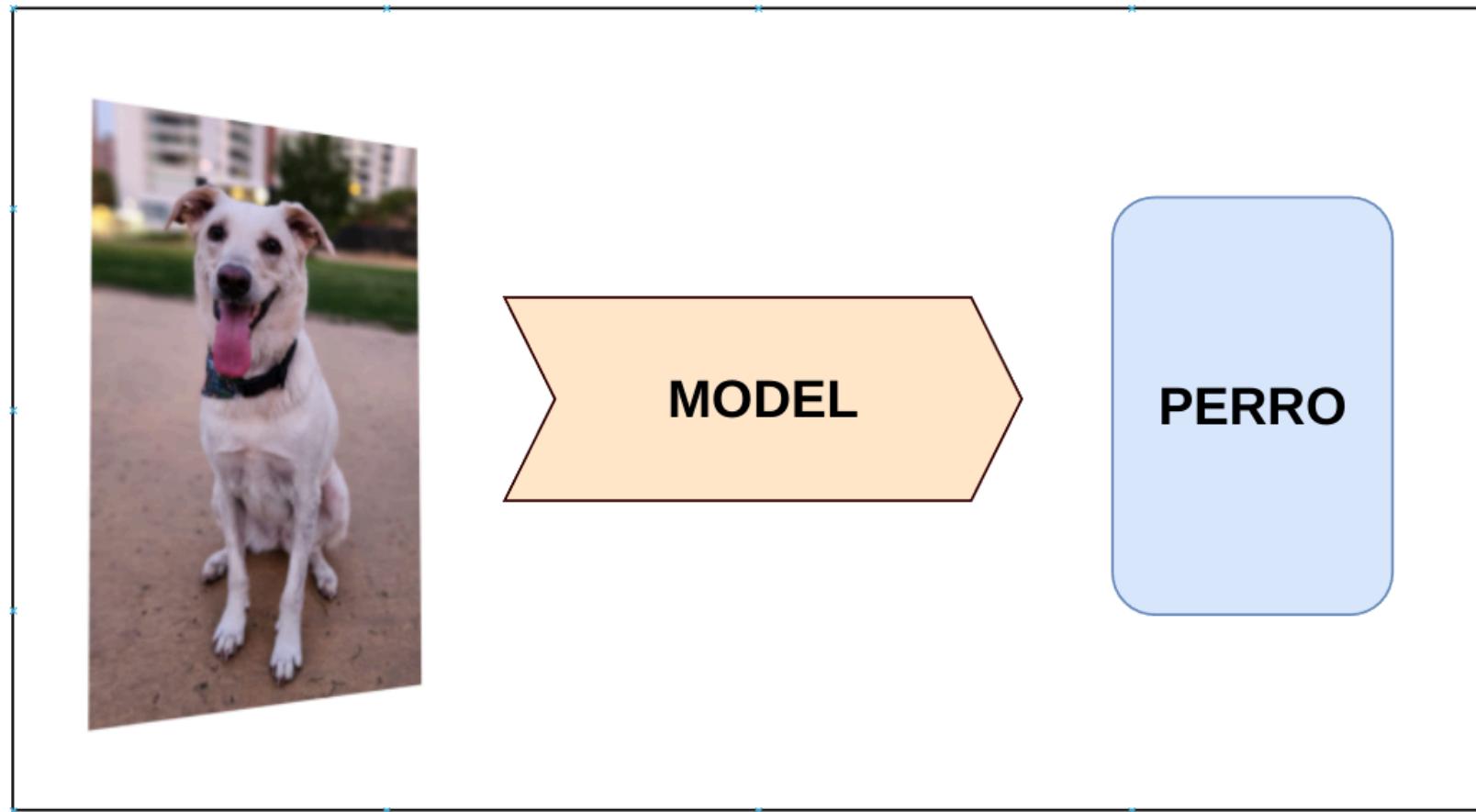
# Intuición

# Nuestro Sistema de ML

Creemos un Sistema de ML que sea capaz de ver una imagen y pronunciar correctamente el uso de la letra **C**.



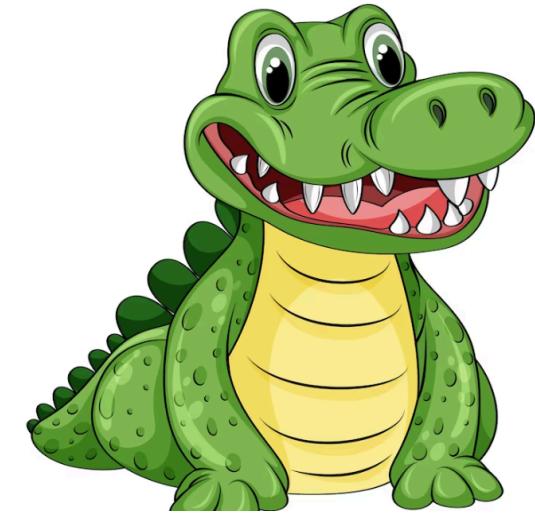
Vamos a **Entrenar** un Modelo.



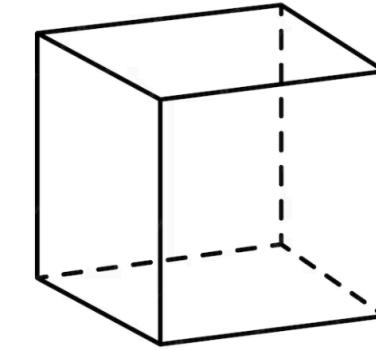
# Nuestro Sistema de ML: Entrenamiento



Kasa



Kokodrilo



Kubo



¿Qué patrones está aprendiendo el modelo?

## Entrenamiento

Es el proceso en el cuál se permite al modelo aprender. En este proceso se le entregan ejemplos (**Train Set**) para que el modelo de manera **autónoma** pueda aprender **patrones** que le permitan resolver la tarea dada.

# Nuestro Sistema de ML: Inferencia

## Inferencia/Predicción

Se refiere al proceso en el que el modelo tiene que demostrar cuál sería su decisión de acuerdo a los patrones aprendidos en el proceso de entrenamiento. Los ejemplos en los que se prueba se le denomina **Test Set**.



Kollar

Konejo

Kukillo

Bikikleta

# Nuestro Sistema de ML: Nuevas instancias de Entrenamiento



Kuchillo



Chokolate



Sinsel



No es bueno entrenar con las mismas instancias de entrenamiento. ¿Por qué?

# Nuestro Sistema de ML: Reevaluemos nuestro Modelo



Kollar



Konejo



Kuchillo



Bisikleta

## Evaluación

Utilizar una métrica que permita **ponerle nota** al modelo.

- 1er Modelo: 2 correctas de 4, es decir **50%**.
- 2do Modelo: 4 correctas de 4, es decir **100%**.

# Problemas del Aprendizaje

Supongamos que queremos utilizar nuestro modelo para pronunciar palabras en otro idioma (otro **Test Set**).

**¿Qué problemas podemos encontrar?**

# Problemas del Aprendizaje: Definiciones

## Overfitting (Sobreajuste)

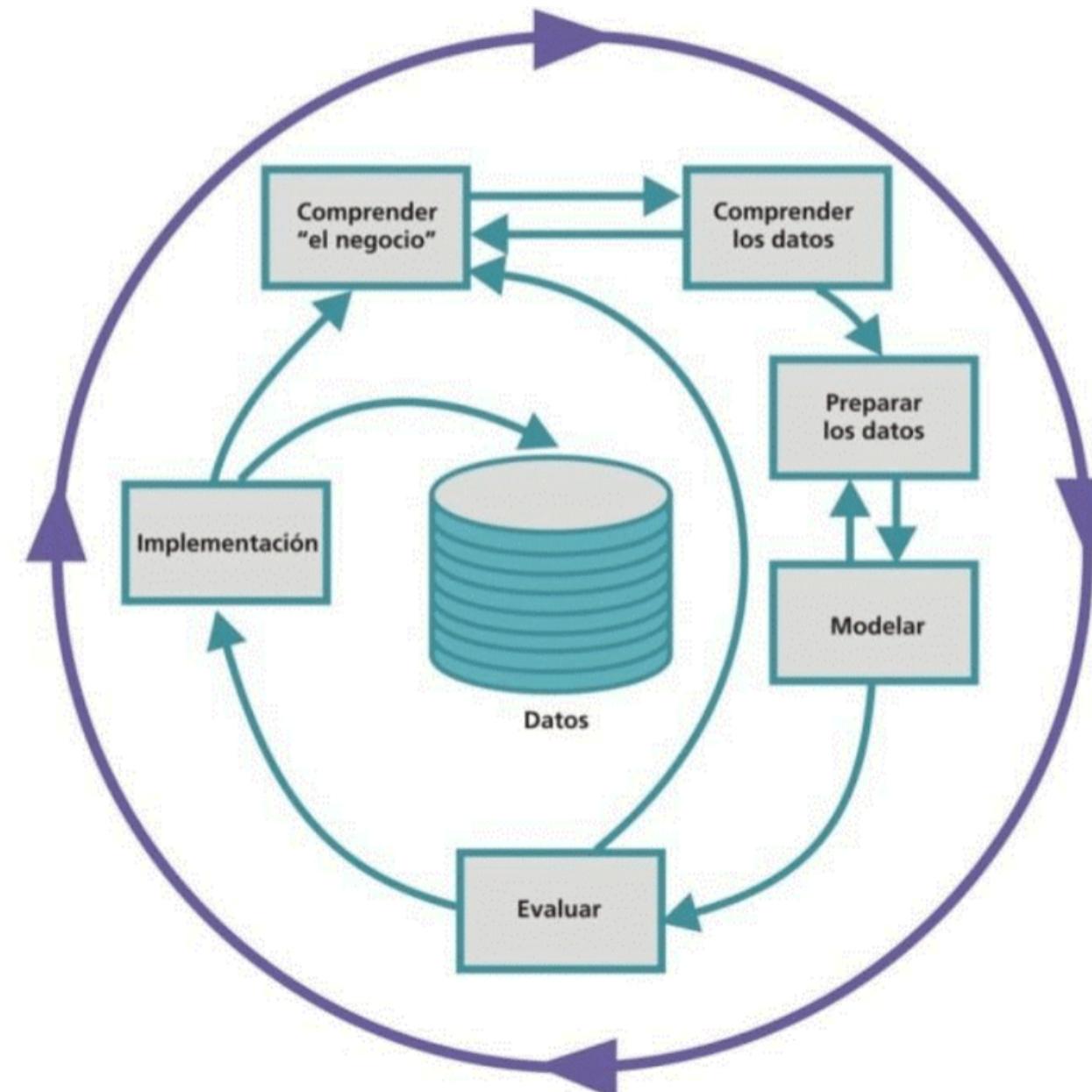
Se refiere a cuando un modelo no es capaz de generalizar de manera correcta, porque se ajusta **demasiado** bien (llegando a **memorizar**) a los datos de entrenamiento. **¿Cómo se puede mitigar este problema?**

 Se le tiende a llamar **sobreentrenamiento**, pero no es del todo correcto para el caso de modelos de Machine Learning. Lo más correcto es que el **sobreentrenamiento** provoca overfitting.

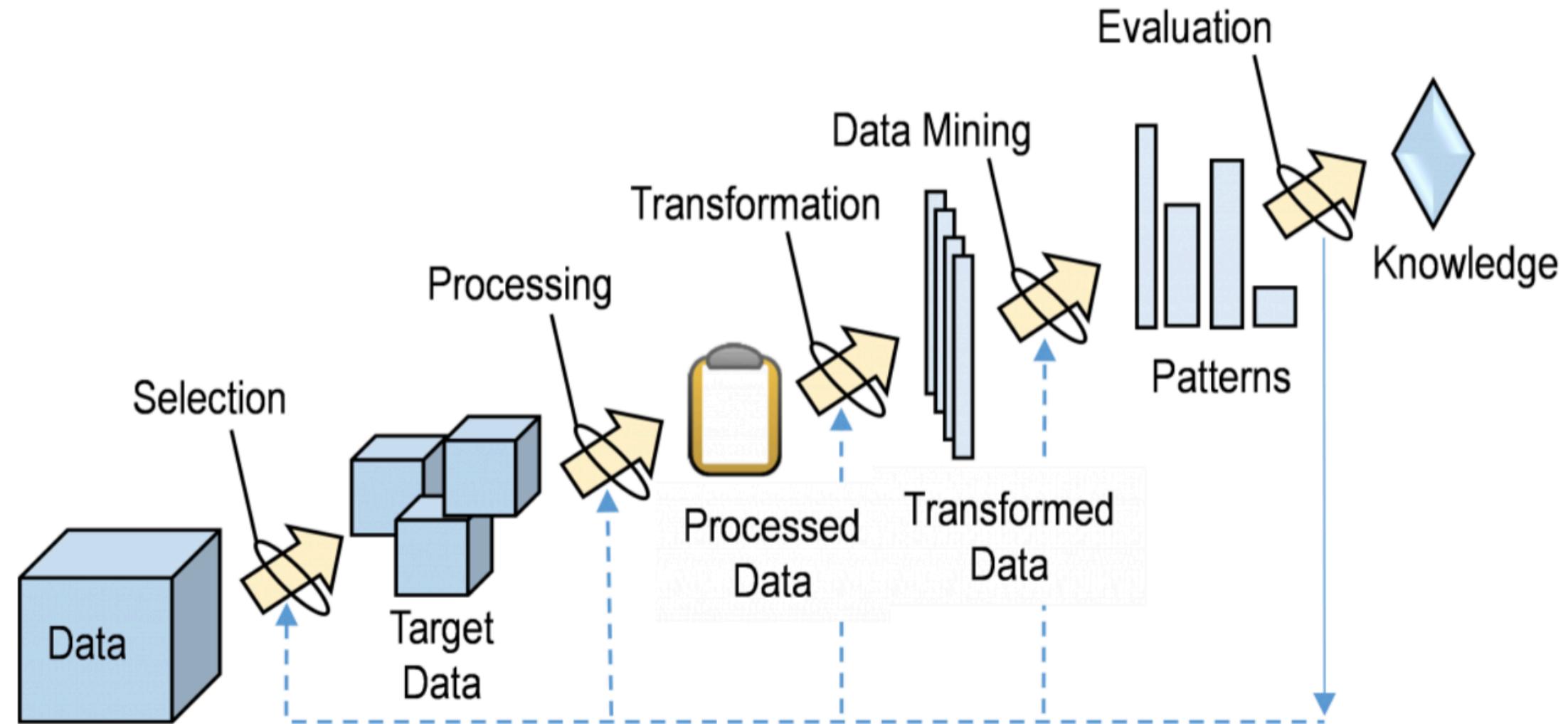
## Underfitting (Subajuste)

Se refiere a cuando un modelo no es capaz de generalizar de manera correcta, pero a diferencia del overfitting **no se ha ajustado** correctamente a los datos. **¿Cómo se vería el underfitting en nuestro ejemplo?**

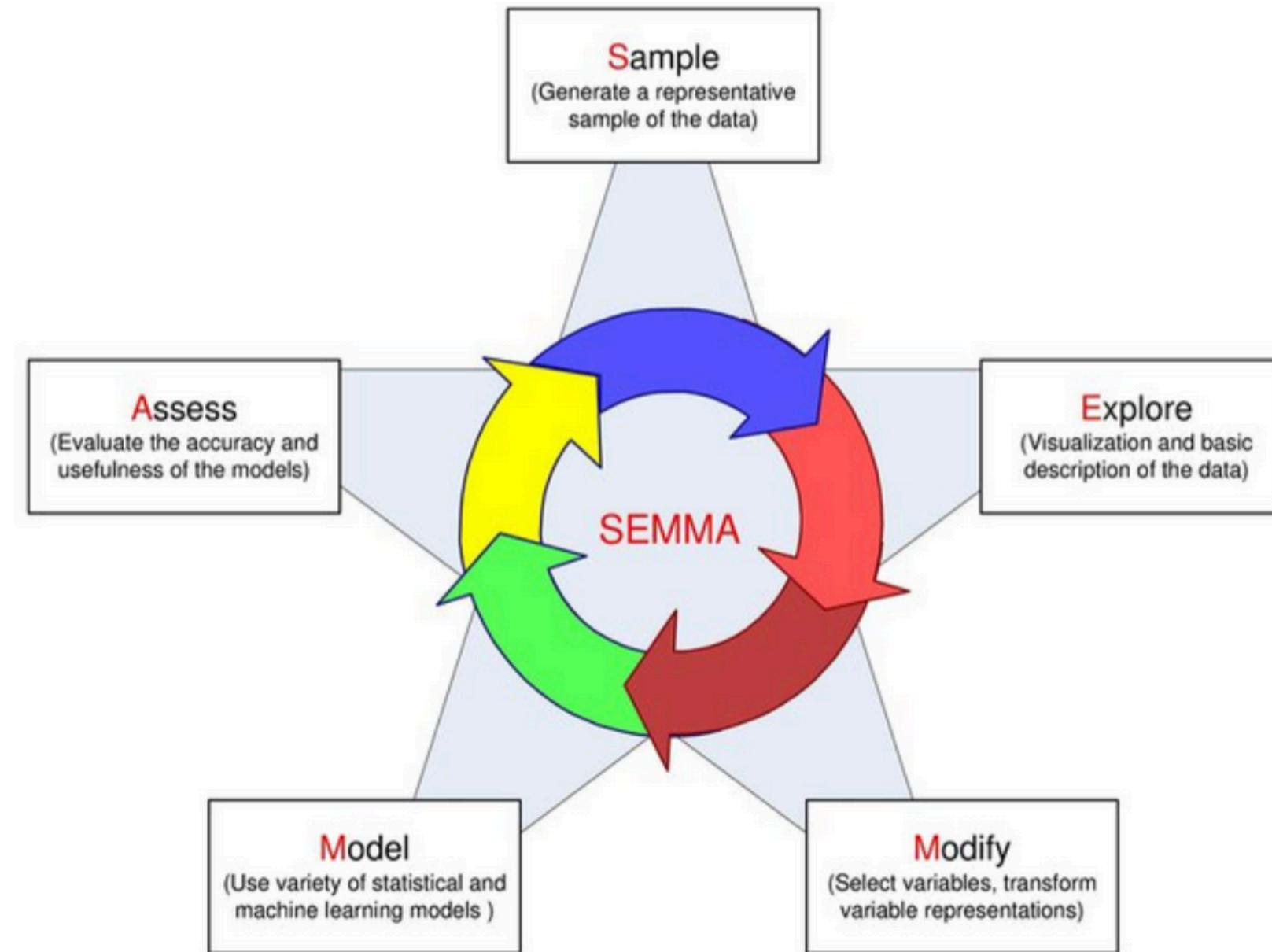
# Etapas del Modelamiento: Crisp-DM



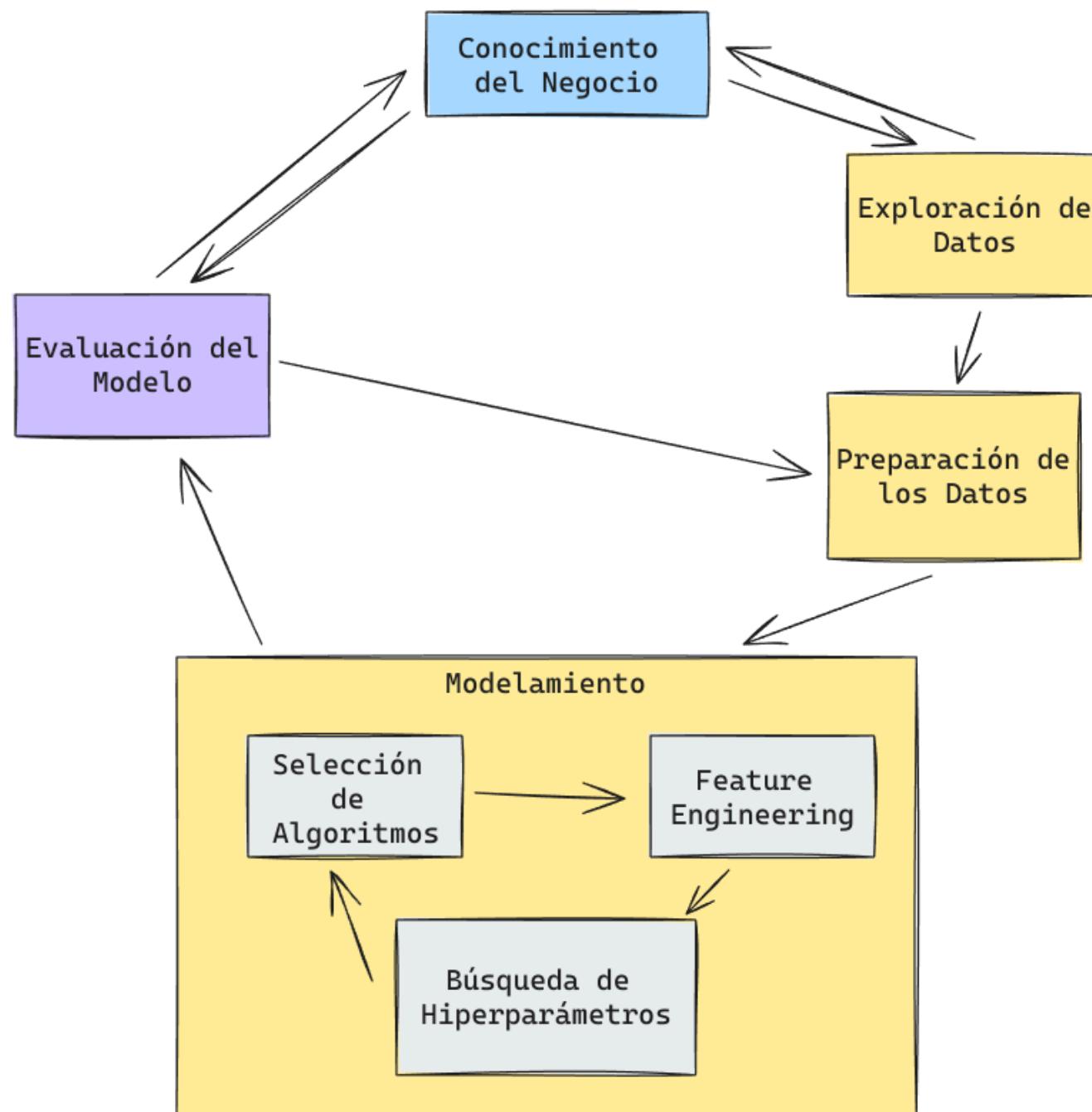
# Etapas del Modelamiento: KDD



# Etapas del Modelamiento: Semma



# Etapas del Modelamiento: Metodología Propia



# Preguntas para terminar

- ¿Qué tipo de modelo debo implementar si quiero estimar la temperatura del día de mañana?
- ¿Qué tipo de modelo debo implementar si es que quiero detectar barrios de acuerdo a su condición socio-económica?
- Si mi modelo aprende a resolver ejercicios de matemática.
  - ¿Cómo se vería el overfitting?
  - ¿Cómo se vería el underfitting?

# Gracias