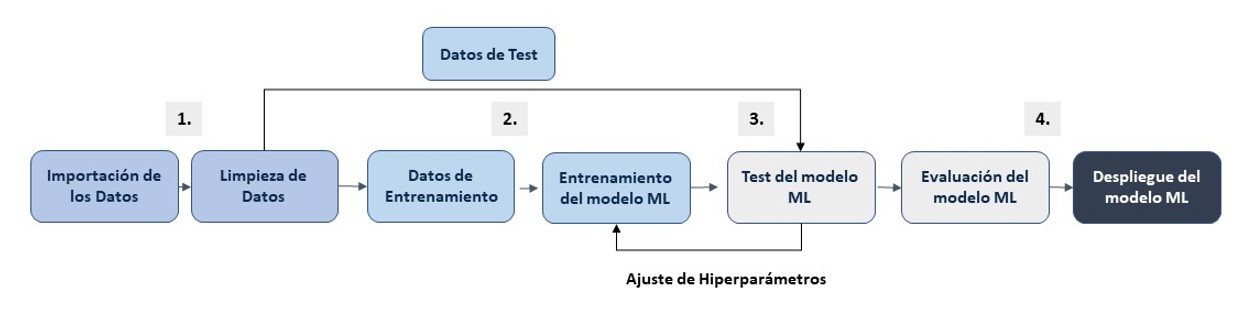
El objetivo de esta lectura previa es que os familiaricéis con los fundamentos de ML y autoML. Por ello, si estáis familiarizados con ambos conceptos podéis continuar al siguiente ejercicio.

**Machine Learning**

Dado que programaremos diferentes modelos supervisados de ML es importante que os familiaricéis con una serie de conceptos básicos. Os recomiendo que reviséis la siguiente [lectura](https://towardsdatascience.com/machine-learning-basics-part-1-a36d38c7916)

A continuación, tenéis algunos conceptos que deberéis conocer:

* **Etapas básicas de ML.** La gran mayoría de modelos de ML siguen los siguientes pasos: importación de datos, preprocesamiento de datos (Feature Engineering y Feature Selection), entrenamiento y selección de modelos, optimización de modelos (ajuste de hiperparametros), test-evaluación de modelos y despliegue.



* **Importar y dividir los datos:** Para poder entrenar un modelo hay que importar los datos y dividirlos en los datos de entrenamiento y de test. Normalmente estos datos se dividen en una proporción de (0/20, es decir, el 80% para el entrenamiento y el 20% para el test.
* **Ingeniería o pre-procesamiento de variables**: Este es el proceso previo a la creación del modelo de predicción en el que se realiza un análisis, limpieza y estructuración de los datos. Es uno de los pasos más importantes y costosos. El objetivo es eliminar los campos que no ayudan a realizar la predicción y organizarlos adecuadamente para que el modelo no reciba información que no sea útil.
* **Tipos de modelos:** hay diferentes tipos de modelos como son: supervisados, no supervisados y aprendizaje profundo. En este curso nos centraremos en los del tipo supervisado, que se diferencian por ser un conjunto de modelos que se entrenan con un conjunto de ejemplos. Dentro de este tipo nos encontramos los de regresión, es decir se entrenan con una variable continua, y los de clasificación, se entrenan con una variable categórica.
* **Evaluación de un modelo:** Para comprobar la precisión de los modelos, las predicciones se prueban en los datos de test, siempre y cuando se trate de modelos supervisados.
* **Validación cruzada (**cross-validation): es una técnica utilizada para evaluar los resultados de un análisis estadístico y garantizar que son independientes de la partición entre datos de entrenamiento y prueba. Consiste en repetir y calcular la media aritmética obtenida de las medidas de evaluación sobre diferentes particiones

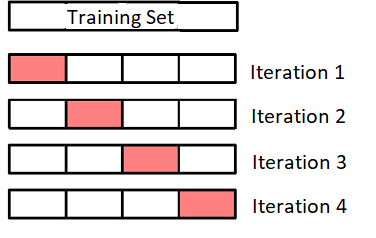
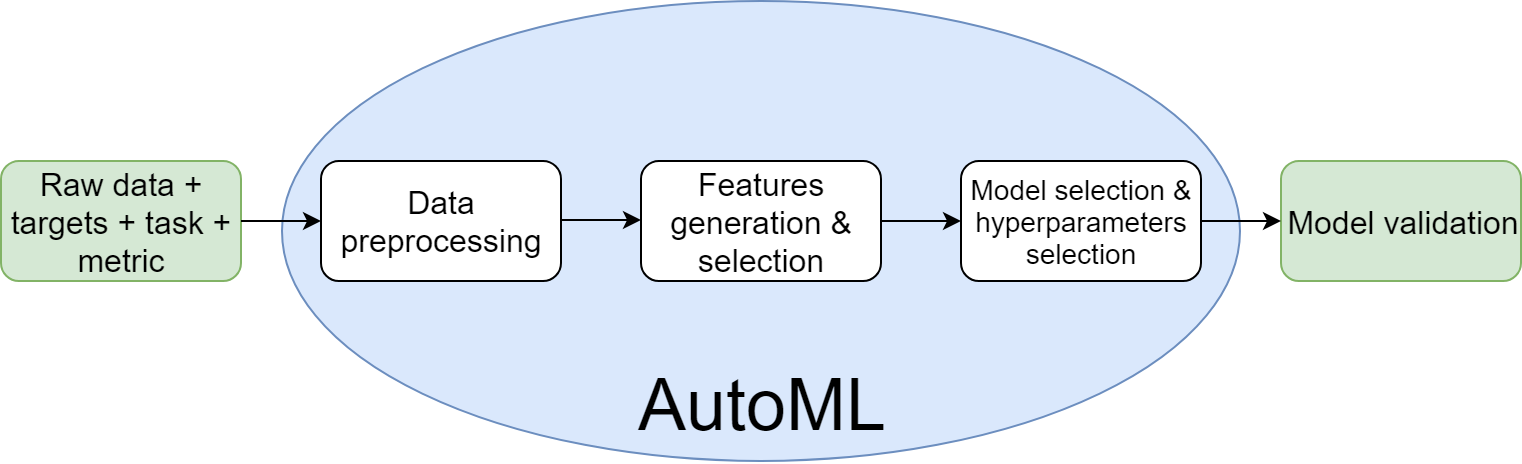


Imagen. Se muestra el proceso de validación cruzada, donde para cada iteración, se selecciona un sub-conjunto de los datos de entrenamiento. Gracias a esto se pueden ir ajustando los hiperparametros del modelo.

* **Ajuste y optimización de hiperparametros**: Los hiperparámetros son parámetros que no pueden ser estimados por el modelo en sí, pero hay que ajustar para que el modelo tenga la máxima precisión. Para optimizarlos el programador establece un conjunto de parámetros y en función de la precisión obtenida selecciona los mejores.
* **Sobreajuste (overfitting):** Aparece en ocasiones cuando el modelo tiene funciones complejas (como es el caso del autoML). El modelo puede ajustar muy bien los datos, pero no es capaz de generalizar para predecir nuevos datos. Esto se observa en que hay una variación de la precisión muy marcada entre las predicciones con los datos de train y test. Hay diferentes técnicas para evitar el sobre-ajuste como: reducir la cantidad de funciones, regularización, etc.



**Librerías de AutoML**

A lo largo de los últimos años se han desarrollado varios paquetes de aprendizaje automático automático. Aunque hay más paquetes que los que se enumeran a continuación, algunos de los más conocidos son:

* **AutoWEKA** es un enfoque para la selección simultánea de un algoritmo de aprendizaje automático y sus hiperparámetros; combinado con el paquete WEKA, produce automáticamente buenos modelos para una amplia variedad de conjuntos de datos.
* **Auto-sklearn** basado en la biblioteca de Python scikit-learn, que es un reemplazo directo para clasificadores y regresores regulares de scikit-learn.
* **TPOT** es un asistente de ciencia de datos que optimiza las canalizaciones de aprendizaje automático mediante programación genética.
* **H2O AutoML** proporciona selección automatizada de modelos y se asemeja a la plataforma de aprendizaje automático y análisis de datos H2O.
* **Pycaret** es una de las librerías de autoML más nueva y completas, es de código abierto

**PYCARET**

PyCaret es una herramienta de automatización del flujo de trabajo para el aprendizaje automático supervisado y no supervisado. Todos los módulos en PyCaret admiten la preparación de datos (más de 25 técnicas de preprocesamiento esenciales, ajuste automático de hiperparámetros, análisis e interpretación de modelos, selección automática de modelos, registro de experimentos y opciones de implementación en la nube)