# EJERCICIO 1

Aprender a entrenar y evaluar un model de machine learning.

Se entrega subiendo el notebook al campus virtual antes del **Lunes 6 de Marzo**

**0. Inicia un entorno virtual con CONDA:**

Vamos a trabajar en LINUX

Crear un entorno virtual e installa jupiter.

**. /usr/local/anaconda/ini_python-anaconda.sh**

**conda create -n EJE1**
**source activate EJE1**
**conda install jupyter**
**jupyter notebook**

**1. Cargar un el conjunto de datos del IRIS data set**

http://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_iris.html

**2. Una vez cargado convertir a pandas**

http://stackoverflow.com/questions/38105539/how-to-convert-a-scikit-learn-dataset-to-a-pandas-dataset

**3. Obtener información básica:**

- Numero de features
- Nombre de las features
- Rango de valores del target
- Valor medio de las features

**4. Obtener información estadística:**

- La media de los valores de cada feature, para cada tipo de flor
- Los valores de la flor sepal length más grande y más pequeño

**5. Obtener la correlación cruzada de todas las features**

http://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.corr.html

**6. Visualiza las con Seaborn Pairplot la correlación de las features**

http://seaborn.pydata.org/generated/seaborn.pairplot.html

**7. Utiliza train/split y KNN para entrenar un modelo**

http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

- Repítelo para diferentes valores K del, 1 al 50
- ¿Cuál es el mejor modelo? ¿Cuál es el peor?
- ¿Qué matriz de confusión de da el mejor model? ¿y el peor?

http://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html

**8. Cross validation**

- Repite el proceso anterior pero usando Cross Validation

http://scikit-learn.org/stable/modules/cross_validation.html

- Una vez elegido el mejor K, guarda el modelo

http://scikit-learn.org/stable/modules/model_persistence.html

**9. Haz un pequeño programa que a partir de la lineas de comandos prediga la categoría de los datos**

./predictor.py 1 2 4 4
>> 1.0

**10. En lugar de KNN utiliza otros métodos y comprueba cual te da mejor score:**

- SVM (support vector classification)
- Random Forest