

Evaluación

- Presentaciones de trabajos 60%.
- Cuestionario (30 minutos, numpy, pandas, aspectos avanzados de programación) 20 %. Último día de clase.
- Ejercicio de programación (30 minutos, programación de una clase y declaración de objetos) 20%. Último día de clase.

Tema de los trabajos

1. Árboles de decisión ID3 y C4.5 (otros tipos de árboles de decisión).

<https://towardsdatascience.com/id3-decision-tree-classifier-from-scratch-in-python-b38ef145fd90>

<https://medium.com/geekculture/step-by-step-decision-tree-id3-algorithm-from-scratch-in-python-no-fancy-library-4822bbfdd88f>

<https://www.youtube.com/watch?v=cEXau63bdOs>

2. Procesos Gaussianos.

<https://towardsdatascience.com/an-intuitive-guide-to-gaussian-processes-ec2f0b45c71d>

https://scikit-learn.org/stable/modules/gaussian_process.html

<https://www.youtube.com/watch?v=92-98SYOdIY>

<https://peterroelants.github.io/posts/gaussian-process-tutorial/>

3. **Gradient Boosting.**

Greedy Function Approximation: A Gradient Boosting Machine. Jerome H. Friedman.

https://en.wikipedia.org/wiki/Gradient_boosting

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html>

XGBoost.

<https://xgboost.readthedocs.io/en/latest/>

4. Dynamic Ensemble selection

<https://arxiv.org/abs/1802.04967>

DESlib: <https://github.com/scikit-learn-contrib/DESlib>

Entrega de trabajo

Presentación el último día de clase (miércoles 25 de enero).

20-25 minutos + ejemplos de programación en Jupiter explicando los compañeros los aspectos de programación (clasificación/regresión).

Rúbrica

Requisitos obligatorios:

- Participación de ambos componentes en la representación.
- Descripción de ejemplos (clasificación/regresión) en un Notebook de Jupiter.
- Presentación ajustada al tema asignado.

Aspectos que se valorarán de manera positiva:

- Presentación ajustada al tiempo asignado. Presentación bien estructurada: introducción, metodología, código, ejemplos, etc.
- Originalidad y novedad de los dataset utilizados en los ejemplos.
- **Detalles de bajo nivel (clases, herencia de clases, etc.) sobre la implementación de los algoritmos.** Esto es en la parte teórica