Machine Learning review

Data Mining

Edgar Roman-Rangel.
edgar.roman@itam.mx

Department of Computer Science.
Instituto Tecnológico Autónomo de México, ITAM.

Machine Learning

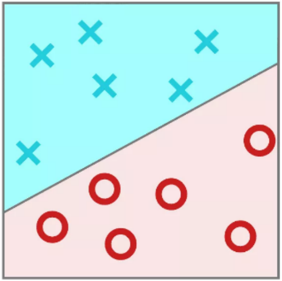
Gives computers the ability to learn without being explicitly programmed.

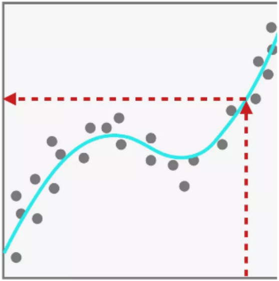
- Supervised, $f : x \in \mathbb{R}^N \to y \in \mathbb{R}$.
- Unsupervised, $f : x \in \mathbb{R}^N \to z \in \mathbb{R}^M$.
- Reinforcement: let an agent, who interacts with and environment, to learn by actions and rewards.

Both input and output data could be scalars, x, or vectors, $\mathbf{x}$.

Supervised learning

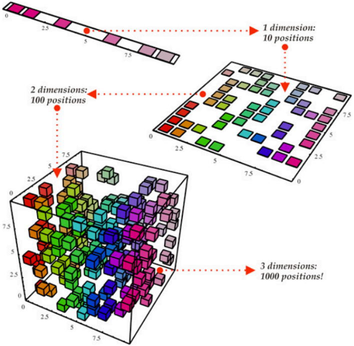
Classification and regression.

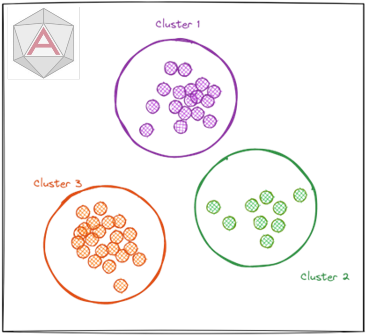




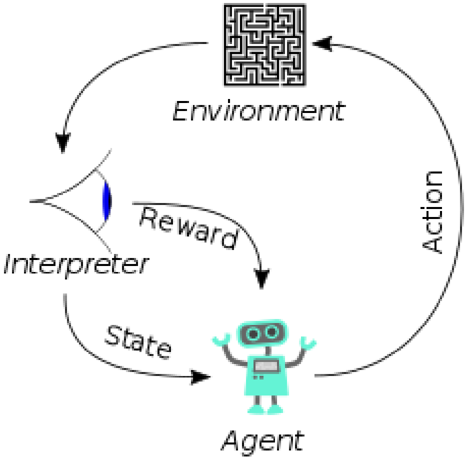
Unsupervised learning

Dimensionality reduction and clustering.



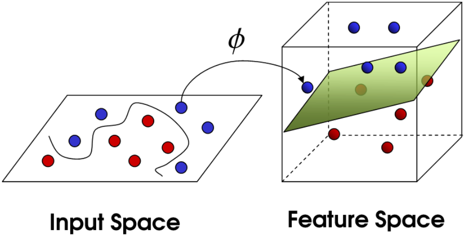


Reinforcement learning



Machine learning (Non-linear problems)

Often, ML deals with non-linear data. For this, Data Mining (DM) can provide tools to project data onto linearly-separable spaces.



- Machine learning can help mine data.
- Data mining can help preparing data for machine learning.

Training, validation, and test

We often define three sets of data in supervised machine learning.

- Training: used for learning the parameter $\{\omega_i\}$ of the model.
- Validation: used to validate the learning process, and to modify hyper-parameters if needed, e.g., (k in kNN, or regularizer α in Lasso regression).
- Test: used for final evaluation of the model. Gives a hint on what level of performance we can expect once the model is released.

Keep in mind the following associations:

- training, learning from data, and parameter; vs
- validation, manually setting, and hyper parameter.

Set splitting

Ideal splitting

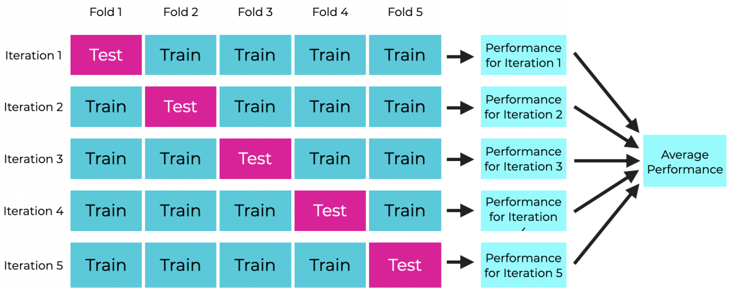
33% training, 33% validation, 33% test.

Common splitting

- 70% training, 20% validation, 10% test.
- 80% training, 10% validation, 10% test.
- 60% training, 20% validation, 20% test.
- 90% training, 5% validation, 5% test.

Cross validation

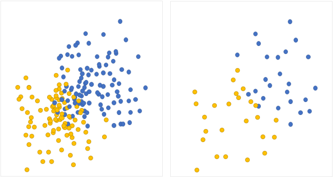
In some cases, we have very small data, or for some reason we only got the chance to have training and test sets (no validation). So we use part of the training set for validation.



	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	
Iteration 1	Test	Train	Train	Train	Train	Performance for Iteration 1
Iteration 2	Train	Test	Train	Train	Train	Performance for Iteration 2
Iteration 3	Train	Train	Test	Train	Train	Performance for Iteration 3
Iteration 4	Train	Train	Train	Test	Train	Performance for Iteration
Iteration 5	Train	Train	Train	Train	Test	Performance for Iteration 5

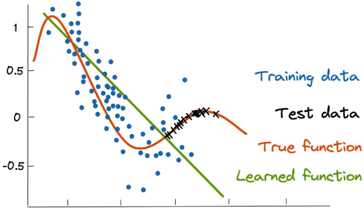
Data distribution

We work under the assumption that all three sets come from the same statistical distribution, i.e., all points are generated from the same phenomenon, all points live in the same space.



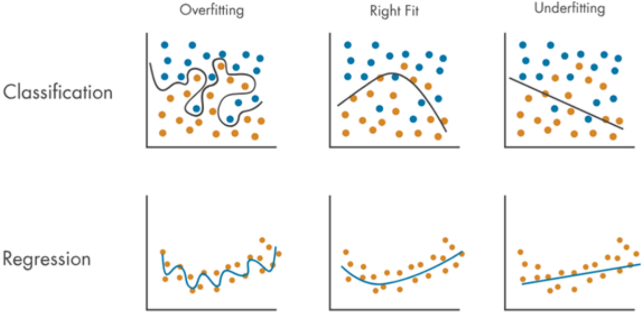
TRAINING DATA

TEST DATA



Under-fitting and over-fitting

We might observe under- or over-fitting when data has high levels of bias or variance, respectively.



Data Leakage

Happens when our training data contains information about the target. This data will not be available at prediction time.

Target leakage

- Predictors are registered after the target variable, e.g., someone takes antibiotics when sick.
- Predictors contain (or is the same as) the target variable, e.g, if we include a time-sample as both predictor and target.

Train-test contamination

Happens when the same point, or a copy of it, exists in both the training and test sets. It might happen in large data bases.

Q&A

Thank you!

edgar.roman@itam.mx