MINICURS: INTRODUCCIÓ A LA INTEL·LIGÈNCIA ARTIFICIAL APLICADA ALS SISTEMES COMPLEXOS

INTRODUCCIÓ AL APRENENTATGE AUTÒMÀTIC

Professora: Dr. Maria Salamó Llorente
Facultat de Matemàtiques i Informàtica, UB

http://ubics.ub.edu
http://ubics.ub.edu/Al_course
Outline

Introduction to Machine learning

1. Introduction
2. Data collection and preparation
3. Machine learning approaches
4. Model evaluation and model selection
5. Example: Point Cloud Rockfall Detection
Introduction
The Learning Machine

Arthur Samuel (1959): Machine Learning: Field of study that gives the computer the ability to learn without being explicitly programmed

Look at the video:

https://youtu.be/6tzt64XKNyQ
DEFINITION OF MACHINE LEARNING

Machine Learning Definition by Tom Mitchell (1998)

- Study of algorithms that
  - at some task \( T \)
  - improve their performance \( P \)
  - based on experience \( E \)

well-defined learning task: \(<T,P,E>\)

Look at the video: From minute 0:55 to minute 7:53

https://youtu.be/m4NlfvrRCdg?list=PLL-BBnDxtUt1hLXmIwu27P22bTi6VwMkN
Applications

Object Detection

CAT, DOG, DUCK
THE MACHINE LEARNING PROCESS

- Data collection and Preparation
- Feature Selection
- Algorithm Choice
- Parameter and model selection
- Training
- Evaluation

Look at the video:
from 0:00 to 45:00
https://youtu.be/mbyG85GZ0Pl
Data collection and preparation

- Training Data
- Machine Learning Algorithm
- Test Data
- h
- Performance
- Feedback
Data

- It can be any unprocessed fact, value, text, sound or picture that is not being interpreted and analysed

- Data is the most important part of all Data Analytics, Machine Learning, Artificial Intelligence
  - Without data, we can’t train any model and all modern research and automation will go vain
Point clouds

- Point clouds are datasets that represent objects or space
- Each point has three geometric coordinates (X, Y, and Z) to position it in space and often also colour and/or intensity information
- It is important to realize that these points are always located on the surfaces of objects
- Point clouds are most commonly generated using 3D laser scanners and LiDAR technology.
Data preparation

- Raw data typically cannot be used directly
  - Machine learning algorithms require data to be numbers
  - Some machine learning algorithms impose requirements on the data
  - Statistical noise and errors in the data may need to be corrected
  - Complex nonlinear relationships may be teased out of the data

- Data preparation or Data pre-processing techniques generally refer to the addition, deletion, or transformation of training set data

- Tasks included in data preparation:
  - **Data cleaning**: Identifying and correcting mistakes or errors in the data
  - **Feature Selection**: Identifying those input variables that are the most relevant to the task
  - **Data Transforms**: Changing the scale or distribution of variables
  - **Feature Engineering**: Deriving new variables from available data
  - **Dimensionality Reduction**: Creating compact projections of the data
Data preparation - transform

- To deal with different ranges
  - Normalize or scale features

Alternatives

- **Standardisation**: Standardisation replaces the values by their Z scores. `sklearn.preprocessing.scale`

- **Mean normalisation**: This distribution will have values between -1 and 1 with $\mu=0$. `sklearn.preprocessing.StandardScaler`

- **Min-Max scaling**: This scaling brings the value between 0 and 1. `sklearn.preprocessing.MinMaxScaler`

- **Unit vector**: Scaling is done considering the whole feature vector to be of unit length. `sklearn.preprocessing.Normalizer`
Data preparation - transform

- To deal with different types

- Alternatives
  - **Label encoding**: convert to a number
    ```python
    sklearn.preprocessing.LabelEncoder
    ```
  - **One hot encoding**: where a categorical variable is converted into a binary vector, each possible value of the categorical variable becomes the variable itself with default value of zero and the variable which was the value of the categorical variable will have the value 1.
    ```python
    sklearn.preprocessing.OneHotEncoder
    ```
Data transform

To deal with missing values
IMBALANCE - Sampling methods

Fig 1. The data set having a between-class imbalance

TO PRACTICE:
https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/
IMBALANCE - Sampling methods

- How to create a “balanced” set
  - Treat the majority class \rightarrow \textit{undersampling}
    - Down-sample
    - Bootstrap (repeat downsampling with replacement)
  - Treat the small class \rightarrow \textit{oversampling}
    - Up-Sample the small class
    - Assign larger weights to the minority class samples
Feature selection

- High-dimensional data exists
  - For classification, clustering or regression
- A universal problem of intelligent (learning) systems is where to focus their attention
  - What aspects of the problem at hand are important/necessary to solve it?
  - Need to discriminate between relevant and irrelevant parts of experience
Feature selection

- **Definition**: Feature selection is a process that chooses an optimal subset of features according to a certain criterion [Liu and Motoda, 1998]

Why reducing dimensionality?

- Theoretically not useful:
  - More information means easier task
  - Models can ignore irrelevant features
    - “In theory, practice and theory are the same. But in practice, they are not”

- Lots of inputs means …
  - Lots of parameters
  - Large input space
  - **Curse of dimensionality** and risks of overfitting!
Curse of dimensionality

- The required number of samples (to achieve the same accuracy) grows **exponentially** with the number of variables!
Feature selection perspectives

1. **searching** for the best subset of features
2. **criteria** for evaluating different subsets
3. **principle for selecting**, adding, removing or changing new features during the search
Feature selection - searching
Search of a Subset of Features

- **Search Directions:**
  - Sequential Forward Generation (SFG)
  - Sequential Backward Generation (SBG)
  - Bidirectional Generation (BG)
  - Random Generation (RG)

- **Search Strategies:**
  - Exhaustive Search
  - Heuristic Search
  - Non-deterministic Search
Feature selection perspectives

Generally are classified according to the criterion function used in searching for good features

1. **Filter algorithm**: some feature evaluation function is used rather than optimizing the classifier’s performance.
2. **Wrapper algorithm**: the performance of the classifier is used to evaluate the feature subsets.
3. **Embedded feature selection algorithm**: performs variable selection (implicitly) in the course of model training. Similar to wrappers, but in this approach, the features are selected during the learning process.
FEATURE selection perspectives

**Feature selection**

**Searching**
- Directions
  - Forward
  - Backward
  - Bidirectional
- Strategies
  - Exhaustive
  - Heuristic
  - Non-deterministic
  - Random

**Criteria**
- Information measures
  - Shannon’s Entropy
- Distance measures
  - Euclidean, Manhattan, Cebishev
- Dependence Measures
  - Pearson Correlation coefficient
- Consistency Measures
  - Chernoff, KLK
- Accuracy Measures
  - Accuracy, Chi-Squared, Information gain

**Principle for selecting**
- Filter
- Wrapper
- Embedded

**TO PRACTICE:**
Machine Learning approaches
THE MACHINE LEARNING PROCESS

- Data collection and Preparation
- Feature Selection
- Algorithm Choice
- Parameter and model selection
- Training
- Evaluation
Framing the problem

Humans have to do a lot of work, up front, to set up a machine learning problem

- What do you want to predict?
- What kind of data can you get?
- What is the relative costs of different types of errors?
- How does the available data relate to future data?

In order to setup a problem, you first need to think about all of these questions
# Paradigms of machine learning

## Supervised Learning
- Labelled data
- Makes machine learn explicitly
- Data with clearly defined output is given
- Direct feedback is given
- Predict outcome/future
- Resolves classification and regression problems

## Unsupervised Learning
- No labels
- Machine understands the data (Identifies patterns/structures)
- Evaluation is qualitative or indirect
- Does not predict/find anything specific
- Find hidden structure in data

## Reinforcement Learning
- Decision process
- Reward based learning
- Machine learns how to act in a certain environment
- To maximise rewards
- Learn series of actions

---

Introduction to machine learning. All Rights Reserved
Paradigms of MACHINE LEARNING
Supervised MACHINE LEARNING TAXONOMY

- Supervised Learning
  - Regression
    - Linear Regression
    - Decision Trees
    - Random Forest
    - ANN
  - Classification
    - Deep Learning
    - Bayesian
    - SVM
    - k-NN
Unsupervised MACHINE LEARNING TAXONOMY

Unsupervised Learning

Clustering
- k-Means
- Hierarchical
- Fuzzy c-means

Dimensionality Reduction
- SVD
- PCA
- ICA

How many clusters?
- Two Clusters
- Six Clusters
Clustering

- In general a **grouping** of objects such that the objects in a **group (cluster)** are similar (or related) to one another and different from (or unrelated to) the objects in other groups.
Clustering Techniques

Partitioning methods
- k-Means algorithm [1957, 1967]
- k-Medoids algorithm
- k-Modes [1998]
- Fuzzy c-means algorithm [1999]

Hierarchical methods
- Divisive
  - DIANA [1990]

Agglomerative methods
- Density-based methods
  - STING [1997]
  - DBSCAN [1996]
  - CLIQUE [1998]
  - DENCLUE [1998]

  - OPTICS [1999]
  - Wave Cluster [1998]

- Model based clustering
  - EM Algorithm [1977]
  - Auto class [1996]
  - COBWEB [1987]
  - ANN Clustering [1982, 1989]

Divisive Agglomerative methods
- STING [1997]
- DBSCAN [1996]
- CLIQUE [1998]
- DENCLUE [1998]
- OPTICS [1999]
- Wave Cluster [1998]

- Model based clustering
  - EM Algorithm [1977]
  - Auto class [1996]
  - COBWEB [1987]
  - ANN Clustering [1982, 1989]
Principal Component Analysis

- **Principle**
  - Linear projection method to reduce the number of parameters
  - Transfer a set of correlated variables into a new set of uncorrelated variables
  - Map the data into a space of lower dimensionality
  - Form of unsupervised learning

- **Properties**
  - It can be viewed as a rotation of the existing axes to new positions in the space defined by original variables
  - New axes are orthogonal and represent the directions with maximum variability
Examples

- **Regression**: 
  - Given a picture of a person, we have to predict their age on the basis of the given picture 
  - The output is continuous

- **Classification**: 
  - Given a patient with a tumor, we have to predict whether the tumor is malignant or benign 
  - The output is discrete
Our problem

<table>
<thead>
<tr>
<th>Av. math grade</th>
<th>Grade in ML</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.0</td>
<td>4.2</td>
</tr>
<tr>
<td>6.2</td>
<td>5.9</td>
</tr>
<tr>
<td>7.4</td>
<td>8.1</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Supervised learning:
Given the "right" answer for each example in the data.

Regression problem:
Predict a real-value output.

Notation
Input (features): $x \in \mathbb{R}^d$ (student’s record)
Output (labels): $y \in \mathbb{R}$ (actual grade in ML)
Data: Examples of inputs and output pairs: $\{(x_1, y_1), \ldots, (x_N, y_N)\}$
**Hypothesis**

A model that maps from the input data to the output label, e.g., \( f(x) = 0.9x + 0.5 \)

**Learning algorithm:** The process for selecting the most appropriate hypothesis from a hypothesis set \( \mathcal{H} \), e.g., the set of linear models \( \mathcal{H}(w_0, w_1) = w_1x + w_0 \), where \((w_0, w_1)\) are called **parameters**.
Univariate linear regression model

Univariate linear regression hypotheses set
\[ \mathcal{H}(w_0, w_1) \equiv w_1 x + w_0. \]

Learning process idea:
We want to find the model defined by the parameters \((w_0, w_1)\) so that our prediction \(f(x_i; w_0, w_1)\) on sample \(x_i\) is as close as possible to \(y_i\). This is, the "distance" between \(f(x_i; w_0, w_1)\) and \(y_i\) is minimum for all elements in the training set.

\[
\min_{w_0, w_1} \frac{1}{N} \sum_{i=1}^{N} (f(x_i; w_0, w_1) - y_i)^2
\]
Visualizing the Parameter space
Visualizing the parameter space

$f_w(x)$

$J(w)$

$w_0 = 0, \ w_1 = 0$
Visualizing the parameter space

\[ f_w(x) \quad J(w) \]

\[ w_0 = 0, \quad w_1 = 0.5 \]
Visualizing the parameter space

$f_w(x)$

$J(w)$

$w_0 = 3, \ w_1 = 0.5$
Visualizing the parameter space

\[ f_w(x) \]

\[ J(w) \]

\[ w_0 = -0.3, \ w_1 = 1 \]
Traveling across the parameter space

Question:
How do we find the minimum in an automated way?

A basic technique:

1. Start with some starting point (guess) \((w_0^{(0)}, w_1^{(0)})\).
2. Change the parameters so that we reduce the cost function
   \(J(w_0^{(k+1)}, w_1^{(k+1)}) < J(w_0^{(k)}, w_1^{(k)})\)
3. Repeat (2) until the cost function is not reduced anymore.
Parameter space

Intuition
Gradient Descent method

Input: given a starting point $x \in \text{dom } f$
Output:
while stopping criterion is not satisfied: do
  (a) Determine a descent direction $\Delta x = -\nabla f(x)$;
  (b) Line search. Choose a step size $t > 0$;
  (c) Update. $x := x + t\Delta x$;
end

Algorithm 2: Gradient descent method
Steepest descent implementation

Input: given a starting point $w = (-10, -10)$, $t = 0.001$, $\tau = 1000$

Output: model $w$

for $k = 1 : \tau$ do

\[
\begin{align*}
    w_0^{(k+1)} &= w_0^{(k)} - t \frac{1}{N} \sum_{i=1}^{N} (w_0^{(k)} + w_1^{(k)} x_i - y_i) 1;
    \\
    w_1^{(k+1)} &= w_1^{(k)} - t \frac{1}{N} \sum_{i=1}^{N} (w_0^{(k)} + w_1^{(k)} x_i - y_i) x_i;
\end{align*}
\]

end
Gradient/Steepest Descent

- Gradient descent is an algorithm to minimize any function or parameter
  - It can be used in Linear Regression but is actually used in several ML algorithms

- $t$ is the learning rate
  - If $t$ is too small, gradient descent will be slow
  - If $t$ is too large, gradient descent can overshoot the minimum. It may fail to converge or even diverge
  - There is no need to decrease $t$ over time
  - Gradient descent can converge even if $t$ is fixed
Model evaluation and model selection
THE MACHINE LEARNING PROCESS

- Data collection and Preparation
- Feature Selection
- Algorithm Choice
- Parameter and model selection
- Training
- Evaluation
How do we evaluate a model’s performance?

- **Model Evaluation** is the process of assessing a property or properties of a model
  - Evaluation metrics
    - A learning algorithm must interpolate appropriate predictions of regions of the instance space that are not included in the training data
  - Evaluation techniques
    - Are designed to provide more reliable estimates of the accuracy of the models learned by an algorithm than would be obtained by assessing them on the training data

- **Model Selection** is the process of choosing among many candidate models for a predictive modeling problem
  - Probabilistic measures
  - Resampling methods
Evaluation metrics

- **Precision & Recall (and f1 score)**
  - Precision: What percent of our predictions are accurate?
  - Recall: How many of the accurate predictions did we capture
  - F1 score: A single number that combines the two values above. Good for ranking/sorting...
    \[
    F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}
    \]

- **Specificity/sensitivity** - very similar to precision/recall. Often used in medicine.

- **Precision at N**
  - How many accurate examples did we capture in our top N ranked examples. This is often used in information (document) retrieval

- **Area under curve (the ROC curve)**
  - A Receiver Operating Characteristic curve (ROC curve) plots the True positive rate (TPR) vs. the False positive rate (FPR). The maximum area under the curve (AUC) is 1. Completely random predictions have an AUC of 0.5. The advantage of this metric is that it is continuous.
Evaluation techniques

Partition training data into separate training/validation sets
Confusion matrix for 2-class problems

![Confusion Matrix Diagram]

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}
\]

\[
\text{Error Rate} = 1 - \text{Accuracy}
\]
Cross-validation

for $i = 1..L$:

- $D_1, D_2, \ldots, D_L$
- $D_{\text{train}} = \bigcup_{j \neq i} D_j$
- $D_{\text{val}} = D_i$

Train the model on $D_{\text{train}}$ and evaluate $e_{\text{val}}(D_i)$

endfor

$E_{\text{val}} = \sum_{i=1}^{L} e_{\text{val}}(D_i)$

if $L = N \implies K = 1$ the process is called **Leave-one-out**. Otherwise, the method is called **$L$-fold cross-validation**.
Example point clouds with classical machine learning algorithms
The problem

Rock falls

Landslide
Input data

LIDAR technology,
Laser Scanning 3D
Rockfall detection Degotalls
Rockfall detection Castellfollit
GOAL: Classification
Description of the process
Imbalance of the data

(a) Degotalls case study

(b) Castellfollit case study
Machine learning pipeline
# Overall Results

## (a) Degotalls case study.

<table>
<thead>
<tr>
<th>Method</th>
<th>$\text{Acc}_b$</th>
<th>Error (%)</th>
<th>$\text{Acc}_b^{\text{best}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>0.78</td>
<td>7.65</td>
<td>0.84</td>
</tr>
<tr>
<td>+Resampling</td>
<td>0.85</td>
<td>3.25</td>
<td>0.89</td>
</tr>
<tr>
<td>+Model Parameterization</td>
<td>0.89</td>
<td>4.65</td>
<td>0.94</td>
</tr>
<tr>
<td>+Feature Selection</td>
<td>0.91</td>
<td>4.45</td>
<td>0.95</td>
</tr>
</tbody>
</table>

## (b) Castellfollit case study.

<table>
<thead>
<tr>
<th>Method</th>
<th>$\text{Acc}_b$</th>
<th>Error (%)</th>
<th>$\text{Acc}_b^{\text{best}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>0.56</td>
<td>18.04</td>
<td>0.80</td>
</tr>
<tr>
<td>+Resampling</td>
<td>0.68</td>
<td>8.21</td>
<td>0.82</td>
</tr>
<tr>
<td>+Model Parameterization</td>
<td>0.79</td>
<td>15.93</td>
<td>0.93</td>
</tr>
<tr>
<td>+Feature Selection</td>
<td>0.82</td>
<td>11.75</td>
<td>0.94</td>
</tr>
</tbody>
</table>
Comments and questions

THANK YOU FOR YOUR ATTENTION