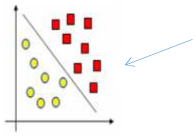
Dynamic Classifier Selection (DCS)
and
Dynamic Ensemble Selection (DES)

ALCEU BRITTO

Pontifical Catholic University of Parana (PUCPR), Brazil

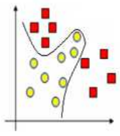
Introduction - Problem Definition

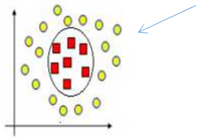
- Classification
  - The most important task in Machine Learning
  - Why?
    - Because, we do it frequently everyday.



- Find the frontier between the problem classes
- Sometimes, it is very easy !!
- A linear separable problem

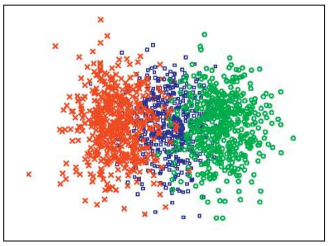
- But, sometimes, it is not so easy.
- Here, we have non linear separable problems





Introduction - Problem Definition

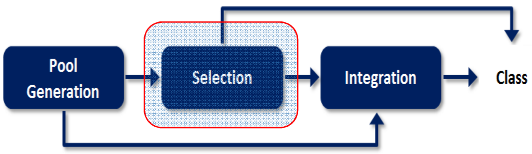
- Frequently,
  - Training a classifier to be capable of learning the wide variability found in a pattern recognition problem is a BIG challenging task.



- A monolithic classification system (based on a single classifier) sometimes is not able to cover well the whole feature space

Introduction - Problem Definition

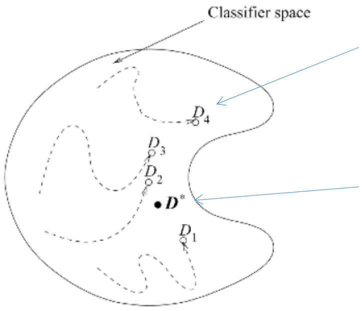
- Alternative
  - Multiple Classifier Systems (MCS)
    - Ensemble in which is expected the elements make different errors (diversity)
    - Diversity => different and sometimes complementary errors.
    - General overview of the MCS phases:



- Our focus today will be the selection model:
  - The use of dynamic selection strategy -> when the selection of classifiers is done during the testing phase.

Introduction – Problem Definition

- Why MCS is an interesting alternative?
  - Avoid the risk involved in the choice of one individual classifier



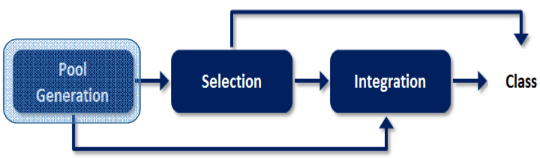
D_1, D_2, D_3 and D_4 are possible solutions (classifiers)

D* is the hypothetical optimal solution

Rationale behind:
by combining the classifiers we expect an "average" solution closer to the optimal solution.

Multiple Classifier Systems (ensembles)

- To select classifiers => we need a pool of them.
- What kind of pools can we generate?
- What is important to observe in a pool of classifiers?
- How can we create a pool of classifiers (or ensemble)?

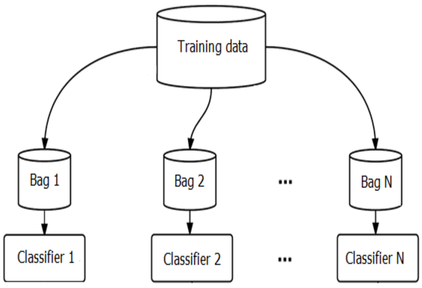


Pool Generation Stage

- Expected: a pool of diverse and accurate classifiers.
- Diversity?
  - Important to observe in a pool of classifiers.
  - Classifiers may compete each other making different and perhaps complementary errors.
- Kind of pools:
  - **Homogeneous**
    - Pool of classifiers based on the same base classifier (or learner)
    - Diversity is obtained by manipulating the training data
  - **Heterogeneous**
    - Pool of classifiers based on different base classifiers (learners)
    - Diversity is obtained by considering different learners (different concepts)

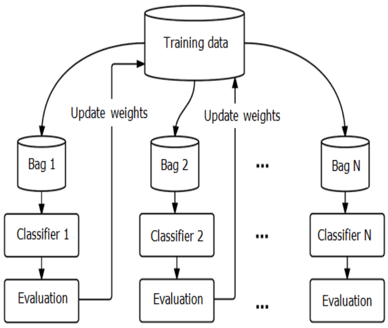
Homogeneous Pool (How to create?)

- Bagging (Breiman, 1996)
  - Bootstrapped Aggregation
  - **Random sampling with replacement** from the original dataset
  - Any element has the same probability to appear in a new Bag
  - Each Bag_i corresponds to X% of the training samples



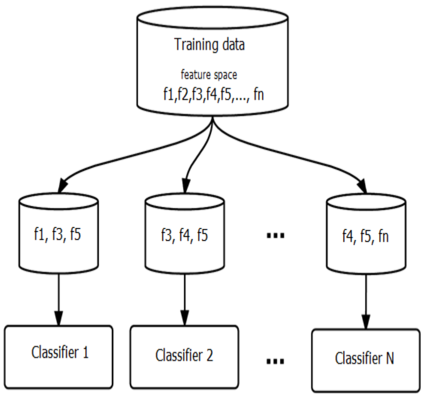
Homogeneous Pool (How to create?)

- Boosting (R.E. Schapire, et al., 1997)
  - **Random sampling with replacement** over weighted data
  - **Misclassified data have its weights increased** to emphasize the most difficult instances. Thus, subsequent classifiers will focus on them during their training.
  - Each Bag$_i$ corresponds to X% of the training samples



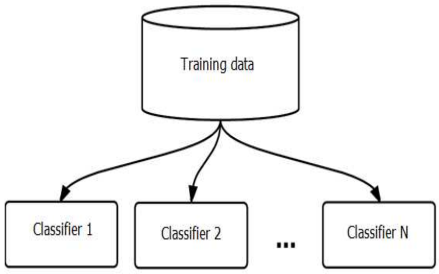
Homogeneous Pool (How to create?)

- Random Subspace (T.K. Ho, 1998)
  - Features ("attributes") are randomly sampled, with replacement, for each learner.



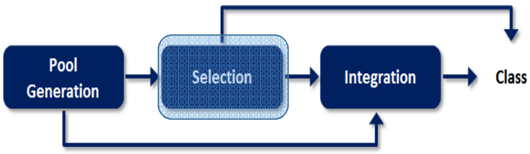
Heterogeneous Pool (How to create?)

- Classifiers trained using different learners
- Different inducers (KNN, Decision Trees, SVM, ...) mean different concepts



Multiple Classifier Systems (ensembles)

- With the pool created, we can combine all of them or perform some selection.
- What kind of selection is possible?



Selection Stage

- Types of selection (in terms of number of classifiers)
  - **Classifier selection** => selection of just one classifier from the pool
  - **Ensemble selection** => selection of a subset of classifiers
- Types of selection (in terms of when the selection is done)
  - **Static** => selection done during the training phase
    - The classifier(s) selected are used to classify the whole testing set
  - **Dynamic** => selection done during the testing phase
    - For each each testing sample is done a specific selection.
- Where to measure the competence of each classifier?
  - Feature space is divided in partitions (one or more **local regions)**
- How to measure the compentence?
  - The **most capable classifier(s)** for each local region is (are) determined.

Static Selection

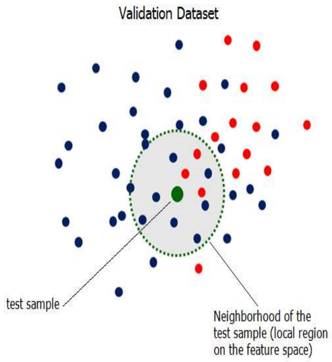
- The **partitioning** is usually based on clustering or evolutionary algorithms, and it is executed **during the training phase**.
- The competence of each classifier is determined during the training phase of the system
- Strategies: Exhaustive search or Optimization processes (GA or PSO, for instance).
- Strategies (Ruta and Gabrys, 2005)
  - Exhaustive Search
  - Forward Search
  - Backward Search
  - Optimization based approaches (GA and PSO based)

Dynamic Selection

- The **partitioning** scheme is usually based on the NN-rule during the **testing phase**.
- The neighborhood of the unknown pattern is defined to measure the classifiers competence.
- Thus, the competence of each classifier is defined on a local region on the entire feature space defined in a validation dataset (DSel).

Dynamic Selection

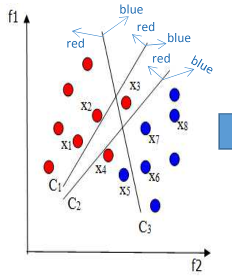
- Where to measure the competence of each classifier?
  - Local region in the feature space defined by the NN rule (K-Nearest Neighbors of the test sample)
- How to measure the competence of each classifier?
  - Different strategies in the literature.



Dynamic Selection

- Why dynamic selection is interesting?
- Given 3 classifiers (C1, C2 and C3)

difficult samples



	(1) - Correct classification / (0) - Misclassification							
Classifiers	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
C_1	1	1	0	0	1	1	1	1
C_2	1	1	1	0	1	1	1	1
C_3	1	1	0	1	0	1	1	1
Fusion (vote)	1	1	0	0	1	1	1	1

- only C_2 is able to correctly classify x_3
- only C_3 is able to correctly classify x_4
- C_1 or C_2 can correctly classify x_5

Fusion: 6/8 (75%)
Best classifier: C_2 => 7/8 (87,5%)
Oracle: 100% (pool upperlimite)
Research question:

How can we select the most promising classifier for each test sample?

Dynamic Selection (Strategies)

- Dynamic Selection Methods [Britto et al, 2014]
  - Competence estimated from each classifier:
    - Ranking scheme: [Sabourin, 1993]
    - Accuracy: OLA and LCA Methods [Woods et at, 1997]
    - Probability information: a Priori, a Posteriori methods [Didaci et al., 2005]
    - Oracle information: Knora method [Ko et al, 2008]
    - Classifier Behaviour: [Giacinto & Roli, 2011]
    - Meta classifier: [Cruz et. Al., 2014]


Pattern Recognition 47 (2014) 3665–3680

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

ELSEVIER

PATTERN RECOGNITION

Dynamic selection of classifiers—A comprehensive review

CrossMark

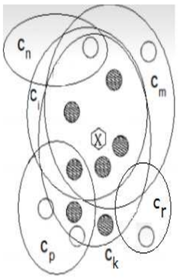
Alceu S. Britto Jr.[a,b,*], Robert Sabourin[c], Luiz E.S. Oliveira[d]

[a] Pontifícia Universidade Católica do Paraná (PUCPR), Curitiba, PR, Brazil
[b] Universidade Estadual de Ponta Grossa (UEPG), Ponta Grossa, PR, Brazil
[c] École de technologie supérieure (ÉTS), Université du Québec, Montreal, QC, Canada
[d] Universidade Federal do Paraná (UFPR), Curitiba, PR, Brazil

  - Competence estimated from the group
    - Diversity: [Santana et al., 2006]
    - Consensus: [Santos at al., 2007]
    - Other group-based measures: [Xiao et al, 2009]

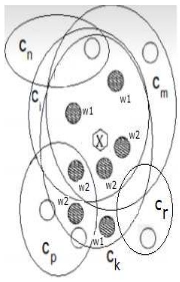
OLA Method (Classifier Selection)

- OLA - Overall Local Accuracy (Woods, 1997)
  - It selects a single classifier
  - Given:
    - pool with N classifiers (c_i, c_k, c_m, c_n, c_p, c_r)
    - test sample x (hexagon in the figure)
    - neighborhood size = 7 (grey circles)
  - Strategy: classifier with best accuracy in the neighborhood of x
  - Example:
    - c_i = 5/7 = 71.4%
    - c_k = 7/7 = 100% (c_k will be selected)
    - c_m = 5/7 = 71.4%
    - c_n = 0/7 = 0%
    - c_p = 2/7 = 28.5%%
    - c_r = 0/7 = 0%



LCA Method (Classifier Selection)

- LCA – Local Class Accuracy (Woods, 1997)
  - It selects a single classifier
  - Given:
    - pool with N classifiers (c_i, c_k, c_m, c_n, c_p, c_r)
    - validation dataset
    - test sample x (hexagon in the figure)
    - neighborhood size = 7 (grey circles)
  - Strategy: classifier with best accuracy considering the class predicted to the test sample (x)
  - Example:
    - c_i = 3/4 = 75% (predicted to x class w_2)
    - c_k = 4/4 = 100% (predicted to x class w_2) (c_k will be selected)
    - c_m = 2/3 = 66% (predicted to x class w_1)
    - c_n = 0/3 = 0% (predicted to x class w_1)
    - c_p = 2/4 = 50% (predicted to x class w_2)
    - c_r = 0/4 = 0% (predicted to x class w_2)



MCB

- MCB – Multiple Classifier Behavior (Giacinto and Roli, 2001)
  - It selects a single classifier
  - Given:
    - pool with N classifiers (c_i, c_k, c_m, c_n, c_p, c_r)
    - Decision space on the validation dataset (output of the classifiers)
    - test sample x (hexagon in the figure)
    - neighborhood size = k (grey circles)
  - Strategy: OLA on neighbors for which the classifiers present similar behavior in terms of decision
  - Example:
    - Similar Behavior:
      - Neighbors 1, 3, 7
    - Final decision: OLA considering neighbors 1, 3 and 7
    - the selected classifier must be significantly better than the others, otherwise the pool is used.

Decision space	c_i	c_k	c_m	c_n	c_p	c_r
Neighbor 1	w_1	w_1	w_1	w_1	w_1	W_1
Neighbor 2	w_2	w_2	w_1	w_2	w_1	w_2
Neighbor 3	w_1	w_1	w_1	w_1	w_1	w_1
Neighbor 4	w_2	w_2	w_2	w_1	w_1	w_1
Neighbor 5	w_1	w_1	w_1	w_2	w_2	w_2
Neighbor 6	w_1	w_2	w_2	w_2	w_1	w_1
Neighbor 7	w_1	w_1	w_1	w_2	w_1	w_1
Test sample	w1	w1	w1	w2	w1	w1

Knora (K-Nearest Oracles)

ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

Pattern Recognition 41 (2008) 1718–1731

PATTERN RECOGNITION

THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/pr

From dynamic classifier selection to dynamic ensemble selection

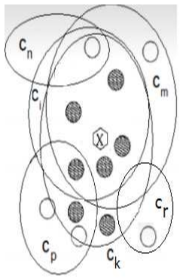
Albert H.R. Ko[a,*], Robert Sabourin[a], Alceu Souza Britto, Jr.[b]

[a]LIVIA, École de Technologie Supérieure, University of Quebec, 1100 Notre-Dame West Street, Montreal, Que., Canada H3C 1K3
[b]PPGIA, Pontifical Catholic University of Parana, Rua Imaculada Conceicao, 1155, PR 80215-901, Curitiba, Brazil

Received 5 March 2007; received in revised form 22 August 2007; accepted 9 October 2007

Knora-Eliminate

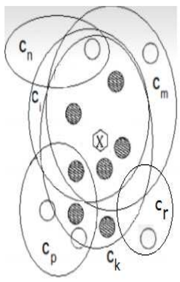
- KNE – Knora-Eliminate (Ko et. al, 2008)
  - It selects an ensemble
  - Given:
    - pool with N classifiers (c_i, c_k, c_m, c_n, c_p, c_r)
    - meta-space constructed using the validation dataset
    - test sample x (hexagon in the figure)
    - neighborhood size = k (grey circles)
  - Strategy: ensemble that correctly classify all neighbors of x
  - Example:
    - Ensemble: c_k



Meta Space	c_i	c_k	c_m	c_n	c_p	c_r
Neighbor 1	1	1	1	0	0	0
Neighbor 2	1	1	1	0	0	0
Neighbor 3	1	1	1	0	0	0
Neighbor 4	1	1	1	0	0	0
Neighbor 5	1	1	1	0	1	0
Neighbor 6	0	1	0	0	1	0
Neighbor 7	0	1	0	0	0	0

Knora-Union

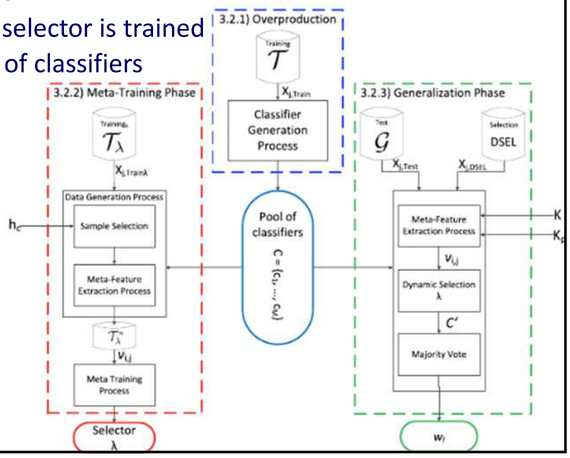
- KNU – Knora-Union (Ko et. al, 2008)
  - It selects an ensemble
  - Given:
    - pool with N classifiers (c_i, c_k, c_m, c_n, c_p, c_r)
    - meta-space constructed using the validation dataset
    - test sample x (hexagon in the figure)
    - neighborhood size = k (grey circles)
  - Strategy: ensemble that correctly classify at least one neighbor of x
  - Example:
    - Ensemble: c_i, c_k, c_m, c_p



Meta Space	c_i	c_k	c_m	c_n	c_p	c_r
Neighbor 1	1	1	1	0	0	0
Neighbor 2	1	1	1	0	0	0
Neighbor 3	1	1	1	0	0	0
Neighbor 4	1	1	1	0	0	0
Neighbor 5	1	1	1	0	1	0
Neighbor 6	0	1	0	0	1	0
Neighbor 7	0	1	0	0	0	0

META-DES

- Meta-DES – Meta Learning based Dynamic Ensemble Selection (Cruz, et al. 2014)
- It selects an ensemble
- Composed of three Phases
  - Meta-Training: a classifier selector is trained
  - Pool Generation: the pool of classifiers is generated
  - Generalization: the operational phase of the system.

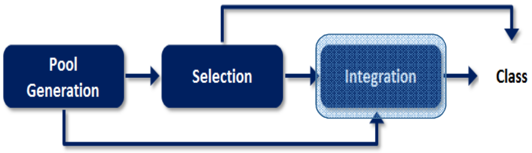


Library for Dynamic Selection

- DesLib
  - https://arxiv.org/pdf/1802.04967.pdf
  - https://github.com/scikit-learn-contrib/DESlib
  - https://pypi.org/project/DESlib/

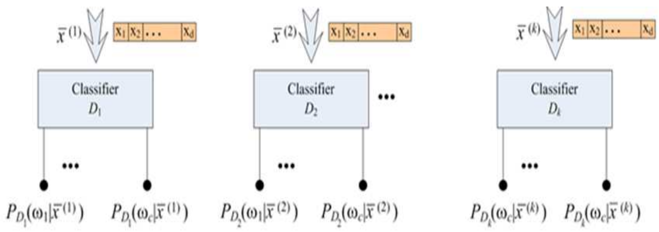
Multiple Classifier Systems (ensembles)

- How can we combine classifiers?



Fusion of Classifiers

- Different strategies in the literature, but:
  - It depends on the classifier output
    - only the class label (few alternatives)
    - a confidence value on its decision (many alternatives)
  - The classifier output can be a probability for each class: $P_{Di}(w_j|x^{(i)})$ is the output probability of the classifier *Di* for class w_j given the test sample $x^{(i)}$.



Fusion of Classifiers

- Majority vote rule (MVR)
  - Possible to use when the classifier output is the class label only.

$$\hat{\omega} = \max_{i \in [1,k]} count\left[\arg\max_{\omega \in \Omega} P_{D_i}(\omega \mid x) \right]$$

Fusion of Classifiers

- Maximum (Max)

$$\hat{\omega} = \arg\max_{\substack{i \in [1,k] \\ \omega \in \Omega}} P_{D_i}(\omega \mid x)$$

- It is usually combined with other rules (final decision).

Fusion of Classifiers

- Sum

$$\hat{\omega} = \arg\max_{\omega \in \Omega} \sum_{i=1}^{k} P_{D_i}(\omega \mid x)$$

- Weighted Sum

$$\hat{\omega} = \arg\max_{\omega \in \Omega} \sum_{i=1}^{k} w_i P_{D_i}(\omega \mid x)$$

the weigth w_i can be learned from the training set.

Fusion of Classifiers

- Product

$$\hat{\omega} = \arg\max_{\omega \in \Omega} \prod_{i=1}^{k} P_{D_i}(\omega \mid x)$$

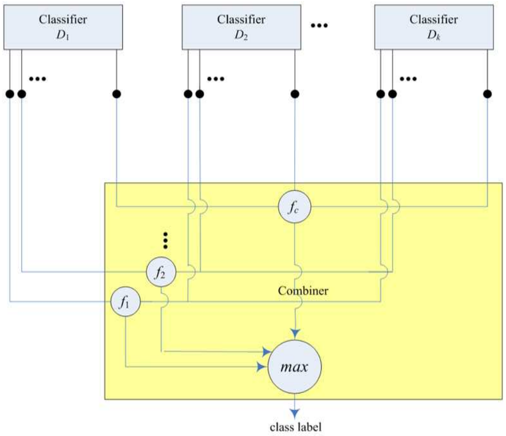
- Weighted product

$$\hat{\omega} = \arg\max_{\omega \in \Omega} \prod_{i=1}^{k} \left[P_{D_i}(\omega \mid x)\right]^{w_i}$$

the weigth w_i can be learned from the training set.

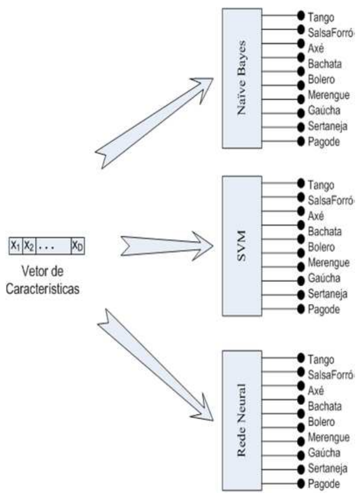
Fusion of Classifiers

- $f_1, \ldots, f_c$: fusion schema at class level
- Max is used to provide the final decision

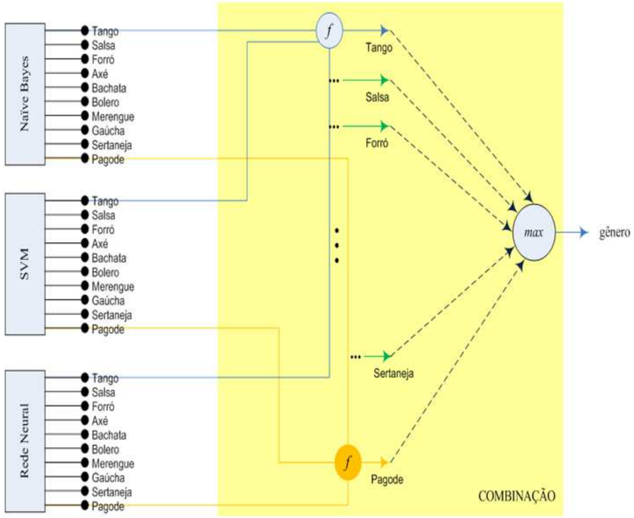


Fusion of Classifers

- Example

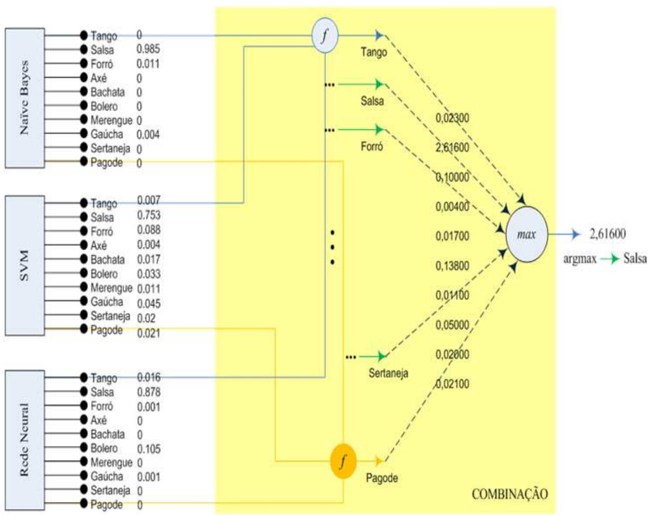


- Fusion of each class first

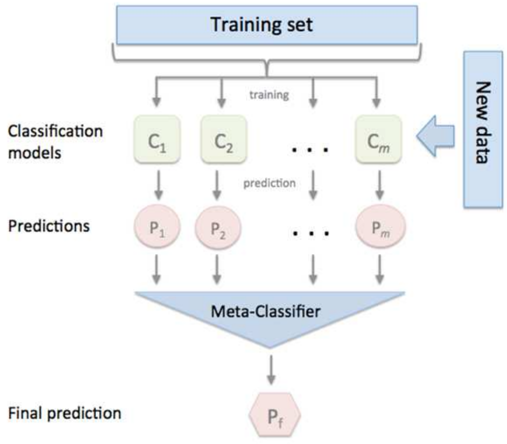


Fusion of Classifiers

- Final Decision



Fusion based on Stacking



- Source: http://rasbt.github.io/mlxtend/user_guide/classifier/StackingClassifier/

References

Duda R., Hart P., Stork D. *Pattern Classification* 2*ed*. Willey Interscience, 2002. Capítulo 8.

Mitchell T. *Machine Learning*. WCB McGraw–Hill, 1997. Capítulo 3.

A. Ko, R. Sabourin, A. Britto Jr., From dynamic classifier selection to dynamic ensemble selection, Pattern Recognition 41 (5) (2008) 1718–1731.

K. Woods, W. P. Kegelmeyer, Jr., K. Bowyer, Combination of multiple classifiers using local accuracy estimates, IEEE Trans. Pattern Anal. Mach. Intell. 19 (4) (1997) 405–410

A. S. Britto Jr., R. Sabourin, L. E. S. Oliveira, Dynamic selection of classifiers - a comprehensive review, Pattern Recognition 47 (11) (2014) 3665 – 3680.

Ruta, D. and Gabrys, B. (2005) Classifier selection for majority voting. Information Fusion, 6, 63-81.

Giacinto Giorgio, and Fabio Roli. "Dynamic classifier selection based on multiple classifier behaviour." Pattern Recognition 34.9 (2001): 1879-1881.

R. M. O. Cruz, R. Sabourin, and G. D. Cavalcanti, "Dynamic classifier selection: Recent advances and perspectives," Information Fusion, vol. 41, pp. 195 – 216, 2018.