Dynamic Selection of Classifiers

Why/How/Where?

by

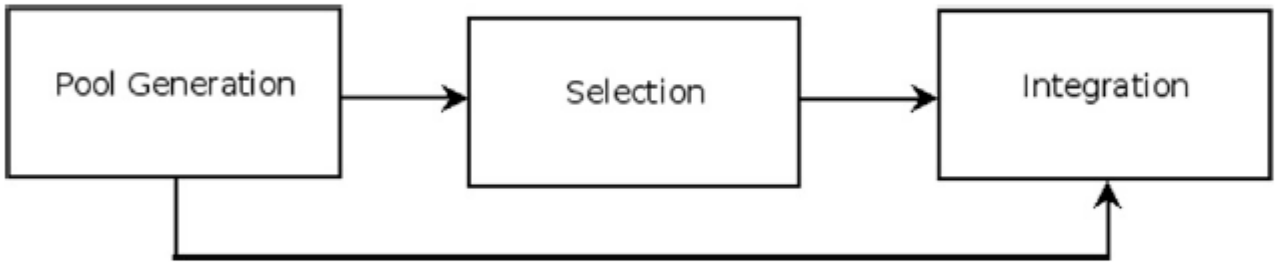
Alceu de Souza Britto Jr.
Pontifícia Universidade Católica do Paraná
Programa de Pós-Graduação em Informática (PPGIa)

Introduction - Motivation

- **Classification**: a fundamental task in Pattern Recognition.
- Although the methods available in the literature may differ in many aspects, the latest research results lead to a common conclusion:
  - "Creating a monolithic classifier to cover all the variability inherent to most pattern recognition problems is somewhat unfeasible".

Introduction - Motivation

- **Alternative**: constrution of Multiple Classifier Systems (MCS).
- **Main idea**: combination of diverse classifiers.
- An MCS is composed of three possible phases:



Introduction - Motivation

- Pool generation
  - Heterogeneous – different base classifiers
  - Homogeneous – same base classifier.
  - The main strategy consists in generating diversity, in other words, classifiers that make different errors.

- Diversity (how to obtain?)
  - Manipulating the training data:
    - Bagging, Boosting and Random Subspace Selection (RSS) techniques
  - Manipulating the classifier parameters
  - Considering different base classifiers (Neural Net, SVM, KNN, ...)

Introduction - Motivation

- Selection of classifiers
  - A single or an ensemble of classifiers can be selected.
  - **Static:** performed during training, the same selected classifiers are used for all testing samples.
  - **Dynamic**: performed during operational phase, a single classifier or a subset is selected for each test instance.

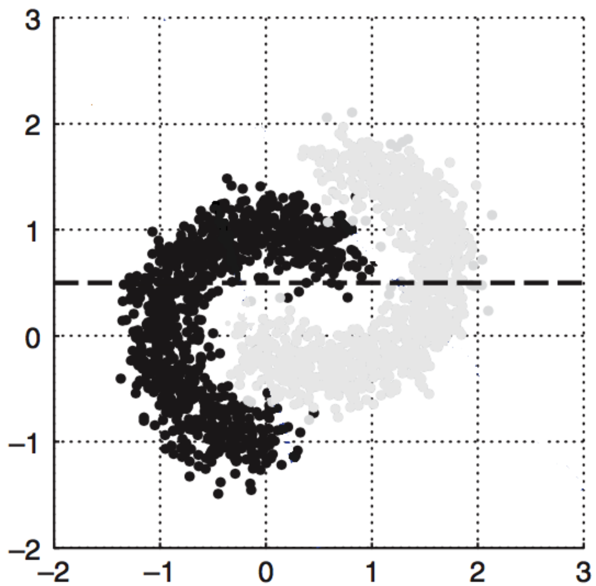
- Fusion
  - Combination of the results provided by the selected classifiers.
  - Different approaches in the literature (max, sum, product, vote, and so on).

Introduction - Motivation

- Our research:
  - Dynamic Selection (DS) of Classifier/Ensembles
- Main directions:
  - New DS-based methods.
    - The KNORA method (proposed in 2007)
    - The DSOC (under construction/evaluation)
  - Application of DS methods in different classification problems (forest species recognition, music genre classification, parking space classification, etc…)
  - A meta classifier to predict when a DS can be better then a monolithic classifier or the combination of all available classifiers.

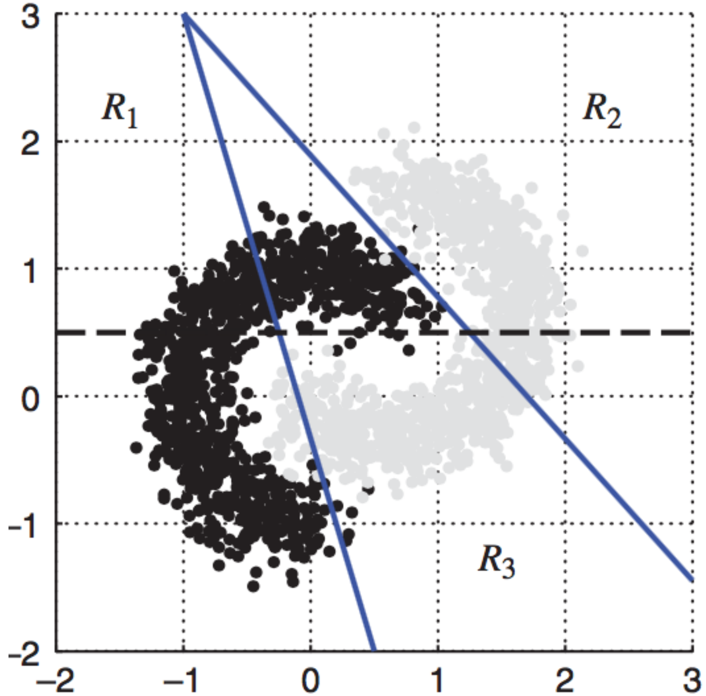
DS – Why it works

- Consider for example the problem below and three classifiers:
  - C1: predicts always the class black.
  - C2: predicts always the class grey.
  - C3: discriminant function is the dashed line.
  - Individual accuracy is about 0.5
- Majority vote won't work.



DS – Why it works

- Define three competence zones
  - Assign one classifier to each competence zone.
  - D1 in R1, D2 in R2, and D3 in R3.
  - Cruz et al, 2015



Current Projects

- A Cascade Strategy for Designing Efficient Multiple Classifier Systems
- DSOC – Dynamic Classifier Selection Based on Data Complexity Analysis
- A meta-classifier to predict the most promising classification strategy for a given problem

A Cascade Strategy for Designing Efficient Multiple Classifier Systems

Master Project

Eunelson Silva Junior

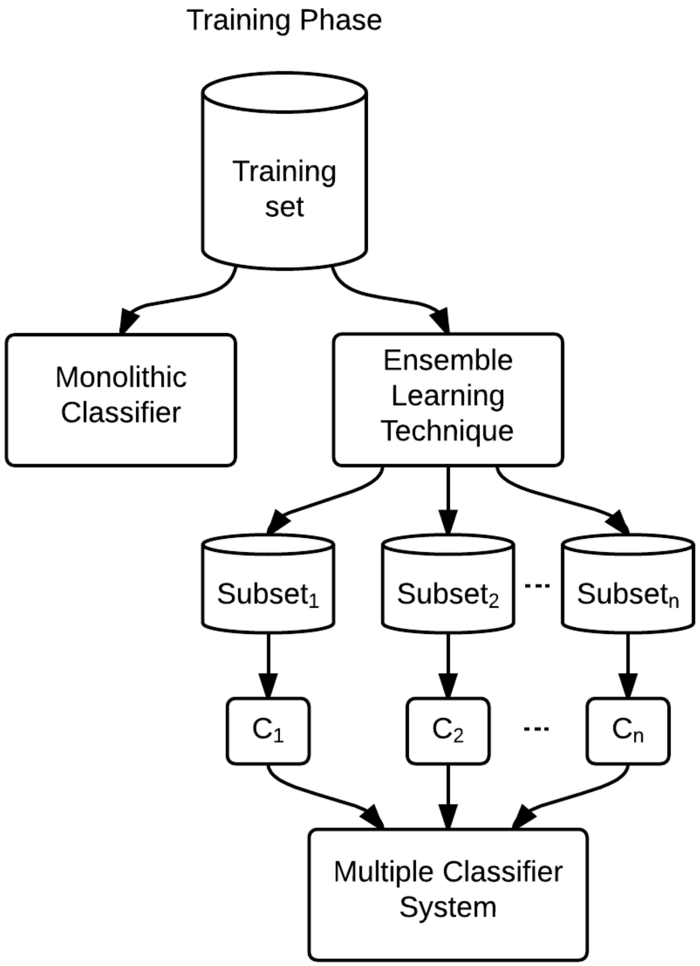
A Cascade Strategy for Designing Efficient Multiple Classifier Systems (Problem)

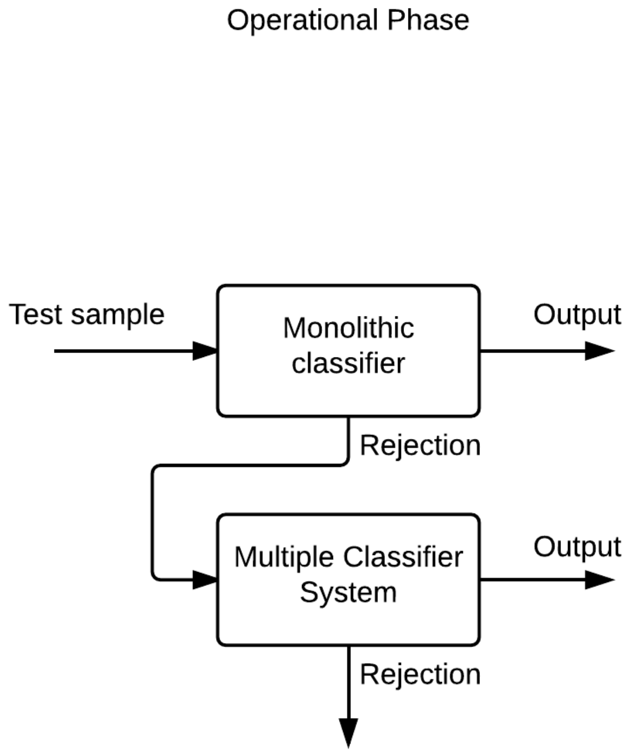
- Attaining high classification accuracy may frequently lead us to the construction of classification systems with an increasing complexity.
- Such a trend in increasing complexity has been a source of frequent criticism against MCS, mainly when the gain in terms of accuracy is not substantial enough to justify that.

A Cascade Strategy for Designing Efficient Multiple Classifier Systems (Hipothesis)

- Considering that a classification problem is usually composed of easy and hard patterns:
  - By combining a monolithic classifier with an MCS composed of diverse experts in a cascading approach, we will be able to deal with problems composed of different levels of difficulty while reducing the efforts necessary to accomplish the classification task.
  - In other words, it could means better compromise between accuracy and complexity.

A Cascade Strategy for Designing Efficient Multiple Classifier Systems (Method Overview)





A Cascade Strategy for Designing Efficient Multiple Classifier Systems (Classification Problems)

Table 1. Description of the used Datasets

Dataset	# classes	# training samples	# testing samples	# features
Liver Disorder (LD)	2	172	173	6
Haberman (HB)	2	153	153	3
Blood (BD)	2	374	374	4
Pima Diabetes (PD)	2	384	384	8
Vehicle (VE)	4	423	423	18
Sonar (SO)	2	104	104	60
Ionosphere (IO)	2	175	176	34
Forest Species (FS)	41	11768	35304	1352
Wine (WI)	3	89	89	13
Wisconsin Breast Cancer (WC)	2	284	285	30
Image Segmentation (IS)	7	210	2100	19
Iris (IR)	3	75	75	4

A Cascade Strategy for Designing Efficient Multiple Classifier Systems (Problem Difficulty)

Table 2. Datasets ranked by difficulty. The values of F1, N2 and N4 complexity measures and the Mean Rank (MR) for each dataset

	F1	N2	N4	MR
LD	0.055	0.91	0.342	1.7
HB	0.189	0.76	0.364	2.7
BD	0.298	0.63	0.396	3.3
PD	0.577	0.84	0.274	4.3
VE	0.451	0.62	0.299	5.3
SO	0.466	0.74	0.094	7.0
IO	0.614	0.63	0.159	7.0
FS	1.854	0.79	0.103	7.3
WI	3.831	0.54	0.131	10.0
WC	3.405	0.56	0.013	11.0
IS	8.809	0.05	0.111	11.6
IR	7.097	0.13	0.081	12.0

- F1 is the Fisher's discriminant ratio
- N2 compares the intraclass dispersion with the interclass Separability.
- N4 describes the nonlinearity of the KNN classifier.

A Cascade Strategy for Designing Efficient Multiple Classifier Systems (Experimental Results)

Table 8. The best cascade result for each dataset. The recognition, error and rejection rates of the first and second steps on the test sets.

	1st Step				2nd Step						Cascade	
Data	Base Clas.	Recog.	Error	Rej.	Base Clas.	Learning Tech.	Dynamic Selection	Recog.	Error	Rej.	Recog.	Cost Reduction
LD	SVM	5.78	1.16	93.06	KNN	Bagging	DS-OLA	13.66	8.70	77.64	18.50	-3.06
HB	SVM	9.15	1.31	89.54	SVM	Bagging	-	10.22	2.92	86.86	18.30	0.46
BD	SVM	2.14	0.27	97.59	SVM	RSS	-	35.34	3.84	60.82	36.63	-30.92
PD	SVM	8.59	0.52	90.89	SVM	Bagging	-	21.20	1.72	77.08	27.86	-0.89
VE	SVM	53.19	1.89	44.92	SVM	Boosting	-	14.21	10.53	75.26	59.57	45.08
SO	SVM	46.15	4.81	49.04	KNN	Boosting	-	82.35	17.65	0.00	86.54	40.96
IO	SVM	55.11	0.57	44.32	SVM	RSS	-	70.51	5.13	24.36	86.36	13.18
FS	SVM	37.69	15.79	46.52	SVM	Bagging	-	43.02	5.03	51.95	57.70	43.48
WI	SVM	100.0	0.0	0.0	-	-	-	-	-	-	100.0	90.00
WC	SVM	93.33	0.70	5.96	SVM	Bagging	-	64.71	11.76	23.53	97.19	84.04
IS	SVM	81.00	1.62	17.38	KNN	Boosting	-	65.75	34.25	0.0	92.43	72.62
IR	SVM	93.33	1.33	5.33	SVM	BoostW*	-	100.0	0.0	0.0	98.67	84.67

Rejection rate

- Considering error rate <= 1% (validation set)

A Cascade Strategy for Designing Efficient Multiple Classifier Systems (Experimental Results)

Table 9. Recognition rates (%) of the proposed cascade approach compared with both the performer monolithic and MCS approach for each dataset. All results considering the rejection scheme. The best results are in boldface

Dataset	Monolithic	MCS	Proposed Cascade
LD	5.78	16.18	**18.50**
HB	9.15	13.07	**18.30**
BD	2.14	35.86	**36.63**
PD	8.59	27.86	**27.86**
VE	53.19	53.90	**59.57**
SO	46.15	84.62	**86.54**
IO	55.11	**86.36**	**86.36**
FS	37.69	42.51	**57.70**
WI	**100.00**	98.88	**100.00**
WC	93.33	**97.19**	**97.19**
IS	81.00	91.62	**92.43**
IR	93.33	97.33	**98.67**

A Cascade Strategy for Designing Efficient Multiple Classifier Systems (Conclusions)

- The experiments have shown that the cascade method can really contribute to reduce the cost of classification task.
- For easy problems, the reduction was very significant, on 8 over 12 datasets some reduction were observed, being 4 superior to 70%.
- Finally, we can say that the observed cost reduction is problem dependent, and it is related to the its level of difficulty.

Dynamic Classifier Selection based on Complexity Analysis

PhD Project

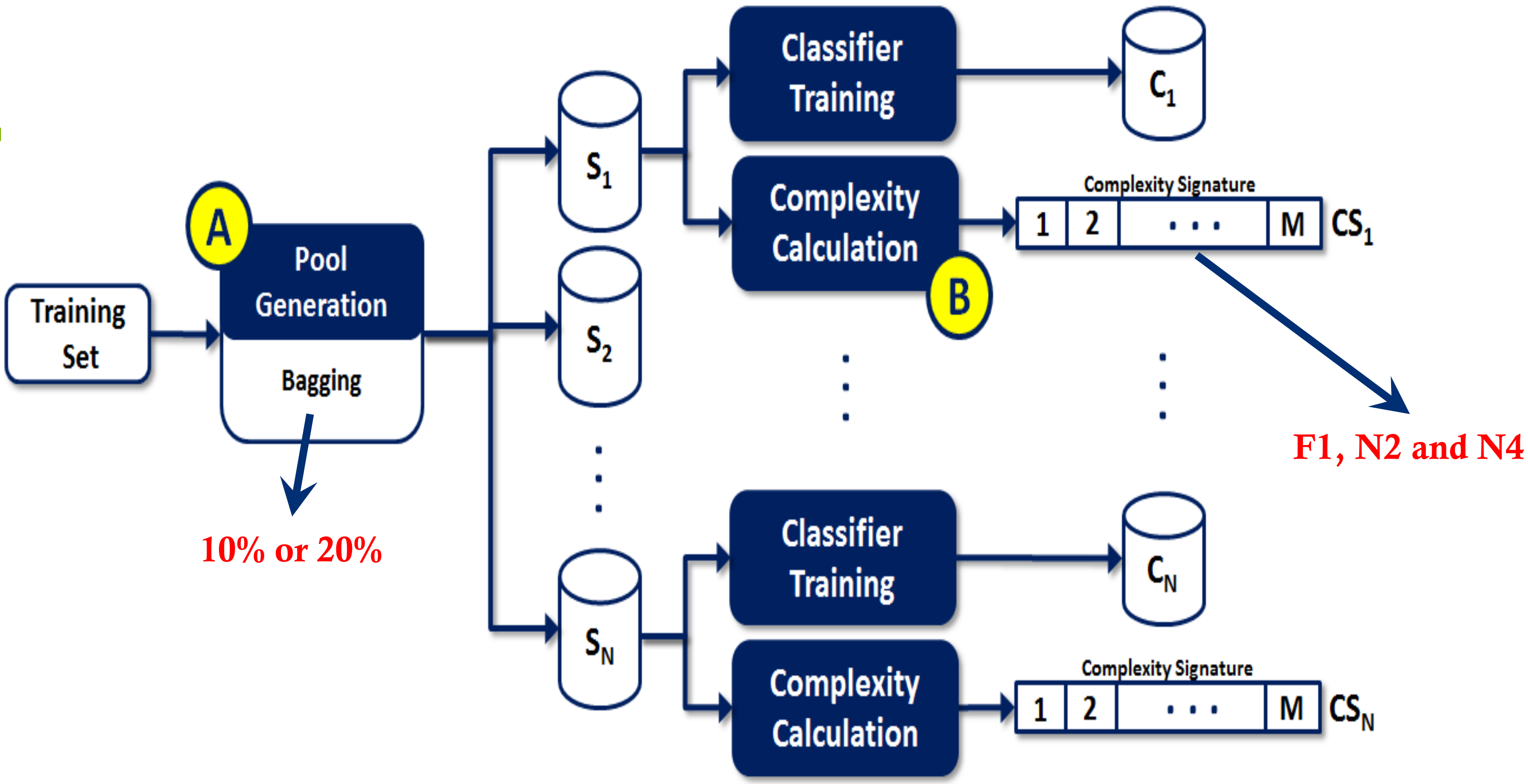
André Luiz Brun

Introduction

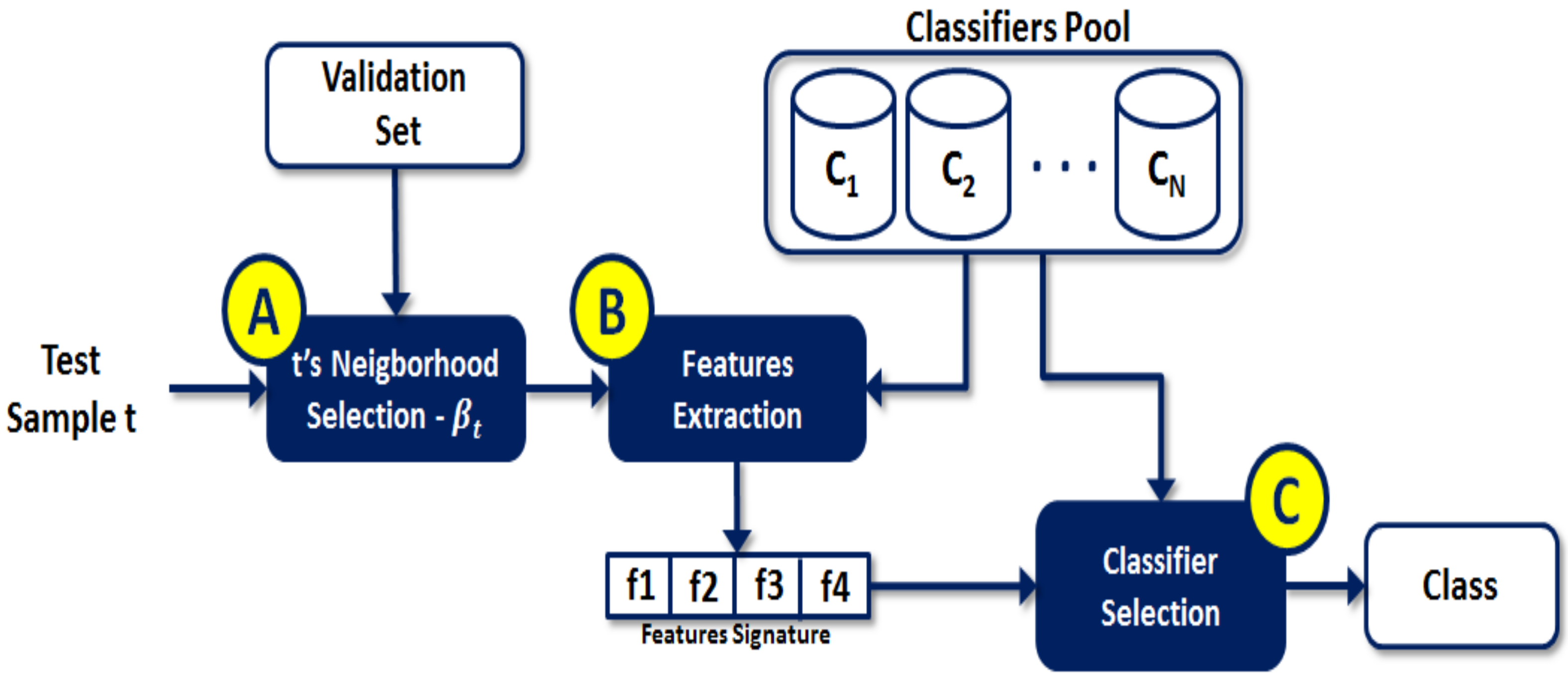
Hypothesis: The most promising classifiers for a given test pattern *t* are those trained on subsets of samples presenting similar complexity (difficulty) than that estimated for the neighborhood of *t* in the training or validation set, and also showing high local accuracy.

Main idea: dynamically select the classifier(s) trained on data with similar complexity than that observed in the local region where the test pattern is located.

Proposed Method (Tranning Phase)



Proposed Method (Operational Phase)



Proposed Method

- f1 - Complexity similarity: the similarity between the neighborhood of t and each classifier complexity signature using the euclidean distance.
- f2 - Centroid Distance: Based on the class predicted by each classifier for the new query, this feature represents the distance (in the feature space) of the test instance to the centroid of the class assigned by the classifier.
- f3 - Local Accuracy: Consists on the local accuracy of each classifier estimated on the test neighborhood.
- f4 - Classifier Complexity: the classifier complexity signature CS obtained in the training phase of the process.

Experimental Results (Classification Problems)

- 30 different datasets

	Instances	Train	Test	Validation	Features	Classes	% Bag	Source
Adult	690	345	172	173	14	2	10	UCI
Banana	2000	1000	500	500	2	2	10	PRTools
Blood	748	374	187	187	4	2	10	UCI
CTG	2126	1063	531	532	21	3	10	UCI
Diabetes	766	383	192	191	8	2	10	UCI
Ecoli	336	168	84	84	7	8	10	UCI
Faults	1941	971	485	485	27	7	10	UCI
German	1000	500	250	250	24	2	10	STATLOG
Glass	214	107	53	54	9	6	20	UCI
Haberman	306	153	76	77	3	2	20	UCI
Heart	270	135	67	68	13	2	20	STATLOG
ILPD	583	292	145	146	10	6	10	UCI
Segmentation	2310	1155	577	578	19	7	10	UCI
Ionosphere	350	176	87	87	34	2	10	UCI
Laryngeal1	213	107	53	53	16	2	20	LKC
Laryngeal3	353	177	88	88	16	3	10	LKC
Lithuanian	2000	1000	500	500	2	2	10	PRTools
Liver	345	173	86	86	6	2	20	UCI
Magic	19020	9510	4755	4755	10	2	10	KEEL
Mammo	830	415	207	208	5	2	10	KEEL
Monk	432	216	108	108	6	2	10	KEEL
Phoneme	5404	2702	1351	1351	5	2	10	ELENA
Sonar	208	104	52	52	60	2	20	UCI
Thyroid	692	346	173	173	16	2	10	LKC
Vehicle	847	423	212	212	18	4	10	STATLOG
Vertebral	300	150	75	75	6	2	20	UCI
WBC	569	285	142	142	30	2	10	UCI
WDVG	5000	2500	1250	1250	21	3	10	UCI
Weaning	302	151	75	76	17	2	20	LKC
Wine	178	89	44	45	13	3	20	UCI

Experimental Results

Data	Single Best	ALL	OLA	LCA	A Priori	A Posteriori	KNORA-U	KNORA-E	DSOC	Oracle
Adult	83.60	**86.72**	82.41	82.27	80.58	78.72	76.60	71.02	**85.12**	99.65
Banana	85.30	84.09	89.22	89.53	86.05	81.85	89.19	84.35	89.53	89.81
Blood	76.44	76.39	74.20	74.17	69.04	22.91	76.39	76.39	74.06	100.00
CTG	69.78	86.63	87.93	88.36	84.06	82.35	85.33	81.30	87.90	99.92
Diabetes	66.02	64.53	69.87	69.97	58.62	57.06	65.52	65.10	70.42	92.29
Ecoli	63.69	42.14	77.92	79.88	55.06	10.06	63.99	42.14	76.13	97.08
Faults	31.18	63.54	64.94	66.38	51.43	50.02	53.64	36.73	65.41	99.21
German	59.52	75.70	68.72	70.02	66.66	62.44	70.08	70.00	**71.56**	100.00
Glass	56.60	58.02	59.91	60.66	46.42	2.17	49.25	33.58	59.62	99.81
Haberman	75.26	73.68	75.26	74.93	73.88	73.49	73.75	73.68	73.55	88.82
Heart	79.10	83.81	76.87	75.67	75.75	67.91	70.82	68.21	**79.33**	100.00
ILPD	68.10	70.55	66.90	67.72	64.62	57.34	71.72	71.72	67.38	99.97
Image	16.13	36.32	68.55	70.88	47.87	32.60	49.86	27.82	68.28	77.79
Ionosphere	78.30	71.99	80.28	86.14	72.05	65.00	79.49	56.31	80.06	98.18
Laryngeal1	80.00	78.58	79.43	79.81	76.23	70.28	69.15	66.89	80.47	99.91
Laryngeal3	66.02	66.53	65.40	66.19	61.53	54.03	57.10	50.06	70.45	99.60
Lithuanian	67.87	50.79	95.94	95.80	85.85	51.70	72.32	50.00	96.14	99.90
Liver	65.58	59.53	64.53	66.69	54.13	12.50	49.94	41.86	**64.59**	100.00
Magic	60.22	78.28	80.69		77.41		77.93	77.34		89.95
Mammo	64.18	81.01	78.86	78.82	77.51	76.76	75.94	72.58	**80.17**	98.31
Monk	78.38	80.46	86.48	86.48	77.51	67.82	63.84	55.09	83.75	100.00
Phoneme	62.16	76.34	81.55	81.99	76.09		74.95	72.90	82.68	96.52
Sonar	61.44	54.62	68.85	70.29	53.56	14.90	53.56	53.17	65.77	100.00
Thyroid	93.32	94.36	94.25	95.55	90.14	49.86	71.99	21.36	86.56	100.00
Vehicle	26.42	36.02	59.05	59.50	35.31	0.00	46.52	25.71	55.66	100.00
Vertebral	80.93	81.33	81.53	81.80	76.47	71.87	75.73	68.67	82.40	100.00
WBC	85.28	53.59	92.68	93.24	81.65	22.71	88.27	62.92	92.22	100.00
WDVG	44.64	83.41	80.07	80.36	77.20		65.23	61.51	78.09	99.88
Weaning	76.93	79.33	77.00	76.93	71.73	59.47	58.40	53.47	**77.40**	100.00
Wine	59.20	32.84	70.00	70.23	61.48	8.52	66.93	32.84	66.59	100.00

Experimental Results

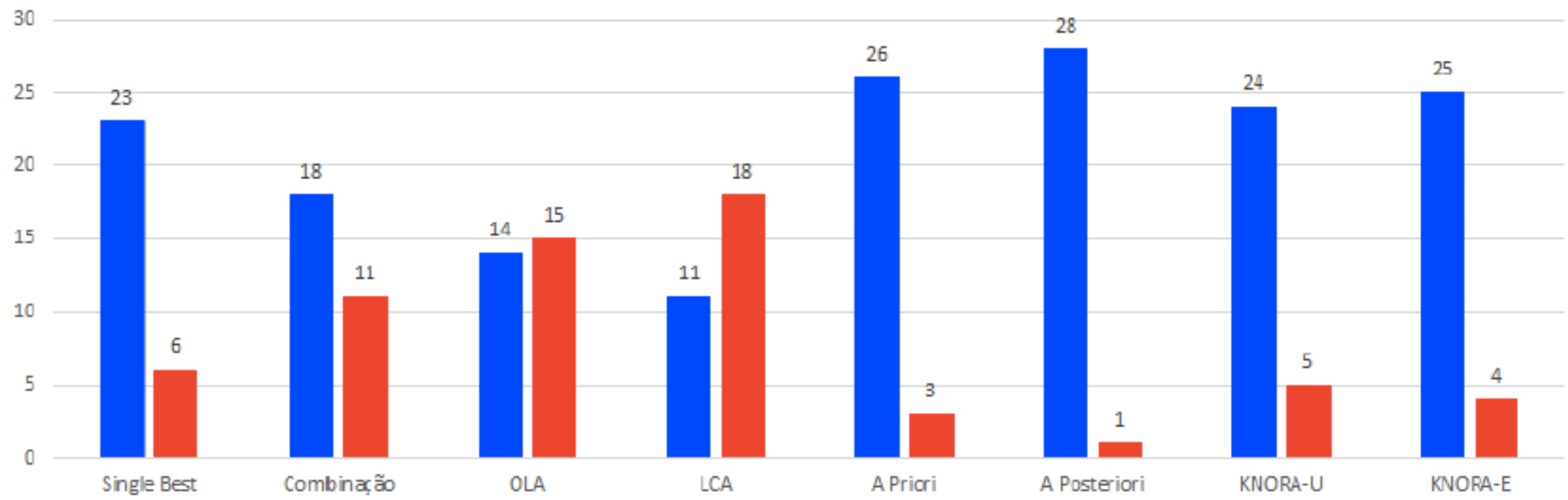


Fig. 4. Pairwise comparison with all methods. The blue bars represent the number of problems where the adoption of complexity outperformed competitor method. Since the red bars refer to the number of times that the proposed approach loses.

Conclusions

- Although the promising results achieved, there is still the need for further study on the influence of the complexity of the data on the selection process.
- It is necessary to have ensembles that better cover complexity space.
- An alternative would be to generate the pool of classifiers taking into account the complexity of the data.

Further Analysis

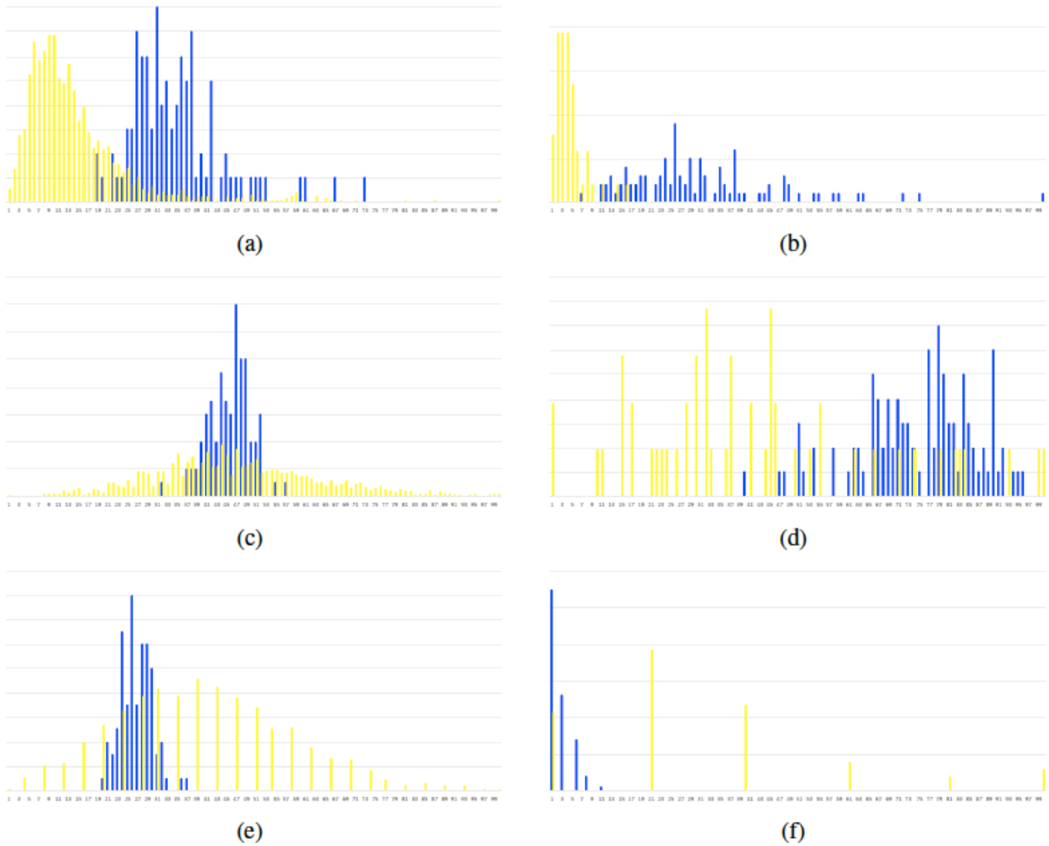


Fig. 3. Overlapping between complexity distributions, in yellow the distribution estimated from the neighborhood of each test instance, and in blue the distribution estimated from the training subsets: (a),(c) and (e) are related to the measures F1, N2 and N4 for the phonema dataset; similarly (b),(d) and (f) are related to the sonar dataset

Meta Classifier to Predict the Classification Strategy

Kelly Laís Wiggers

PhD Project

Hypothesis

- We may predict the most promising classification strategy (monolothic or MCS) for a given problem from data complexity analysis.

Methodology - Databases

- 300 problems presenting different levels of difficulty
- Artificial datasets
- Two-class problems.

Dataset	# Instances
D100	302
D101	461
D102	307
D103	300
D104	533
D105	626
D106	627
D107	324
D108	301
D109	304
D10	302
D110	237
D111	301
D112	348
D113	543
D114	493
D124	314

Methodology – Meta Classifier

- Monolithic Classifiers: a decision tree (DT) for each problem
- MCS
  - N classifiers (an ensemble without selection)
  - Homogeneous pool - DT as base classifier
- The 300 datasets were labeled considering the most promising classification strategy (monolithic or ensemble).

Methodology – Meta Classifier

Labeling procedure:
- Critical difference between the results (monolithic and ensemble) based on statistical analysis of significance

- Classes:
    - 2 Monolithic or ensemble
    - 3 Monolithic, ensemble or anyone

Base	Mono	Ensemble	Result
100	90,7285	89,4040	Anyone
101	91,1063	93,7093	Ensemble
102	85,6678	87,9479	Ensemble
103	89,0000	89,3333	Anyone
104	92,4953	93,6210	Anyone
105	93,4505	94,4089	Anyone
106	94,5774	94,0989	Anyone
107	83,0247	85,4938	Ensemble
108	84,7176	88,0399	Ensemble
109	84,8684	87,5000	Ensemble
10	92,7152	93,3775	Anyone
110	83,1224	82,7004	Anyone
111	88,3721	87,3754	Anyone
112	82,1839	84,7701	Ensemble
113	97,2376	97,2376	Anyone
114	79,3103	80,5274	Anyone
115	81,0631	81,7276	Anyone
116	75,6667	80,6667	Ensemble
117	77,7457	78,3237	Anyone
118	79,1749	79,1749	Anyone
119	78,0992	83,0579	Ensemble

Methodology – Meta Classifier

- After labeling the datasets, 12 complexity measures were calculated for each base:
  - F1: Maximum Fisher's discriminant ratio
  - F1v: Directional-vector maximum Fisher's discriminant ratio
  - F2: Overlap of the per-class bounding boxes
  - F3: Maximum (individual) feature efficiency
  - F4: Collective feature efficiency (sum of each feature efficiency)
  - L1: Minimized sum of the error distance of a linear classifier (linear SMO)
  - L2: Training error of a linear classifier (linear SMO)
  - L3: Nonlinearity of a linear classifier (linear SMO)
  - N1: Fraction of points on the class boundary
  - N2: Ratio of average intra/inter class nearest neighbor distance
  - N3: Leave-one-out error rate of the one-nearest neighbor classifier
  - N4: Nonlinearity of the one-nearest neighbor classifier

Methodology – Meta Classifier

Dataset	F1	F1V	F2	F3	F4	L1	L2	L3	N1	N2	N3	N4	CLASS
D100	0.041	0.411	0.308	0.652	1.000	0.259	0.129	0.500	0.354	0.917	0.268	0.101	Anyone
D101	0.120	0.505	0.001	0.627	0.993	0.256	0.128	0.500	0.310	0.885	0.200	0.114	Ensemble
D102	3.457	8.066	0.072	0.599	0.909	0.444	0.134	0.070	0.352	0.930	0.277	0.088	Ensemble
D103	2.677	6.757	0.010	0.613	0.997	0.384	0.173	0.500	0.313	0.872	0.227	0.107	Anyone
D104	3.917	8.466	0.060	0.606	0.795	0.519	0.169	0.386	0.270	0.856	0.159	0.103	Anyone
D105	3.596	7.958	0.020	0.641	0.826	0.508	0.168	0.420	0.235	0.844	0.128	0.089	Anyone
D106	3.625	7.989	0.024	0.641	0.826	0.511	0.159	0.387	0.234	0.844	0.128	0.099	Anyone
D107	3.038	6.651	0.145	0.562	0.858	0.475	0.154	0.083	0.426	0.927	0.262	0.099	Ensemble
D108	3.376	7.656	0.042	0.571	0.890	0.480	0.146	0.098	0.389	0.920	0.276	0.075	Ensemble
D109	2.569	6.147	0.021	0.572	0.885	0.563	0.178	0.201	0.401	0.905	0.263	0.087	Ensemble
D10	0.022	0.284	0.005	0.523	0.993	0.676	0.338	0.500	0.573	0.994	0.450	0.132	Anyone
D110	0.784	1.008	0.044	0.367	0.506	0.376	0.186	0.500	0.401	0.804	0.316	0.283	Anyone
D111	2.929	6.972	0.047	0.585	0.867	0.492	0.153	0.090	0.462	0.936	0.332	0.100	Anyone
D112	0.520	1.766	0.013	0.184	0.517	0.388	0.193	0.500	0.362	0.910	0.261	0.145	Ensemble
D113	0.020	0.264	0.006	0.177	0.976	0.461	0.230	0.500	0.361	0.883	0.215	0.192	Anyone
D114	0.038	0.297	0.028	0.055	0.333	0.373	0.187	0.500	0.438	0.927	0.294	0.222	Anyone
D115	0.018	0.208	0.026	0.143	0.518	0.352	0.176	0.500	0.409	0.910	0.296	0.179	Anyone
D116	1.080	3.156	0.042	0.150	0.490	0.602	0.163	0.155	0.493	0.944	0.347	0.108	Ensemble
D117	0.040	0.349	0.039	0.043	0.358	0.428	0.214	0.500	0.436	0.940	0.301	0.207	Anyone
D118	0.022	0.149	0.081	0.031	0.248	0.417	0.208	0.500	0.444	0.933	0.316	0.196	Anyone
D119	0.863	2.607	0.043	0.070	0.202	0.688	0.347	0.500	0.264	0.747	0.169	0.153	Ensemble
D11	0.025	0.307	0.011	0.484	1.000	0.704	0.352	0.500	0.605	0.985	0.484	0.163	Anyone
D120	0.742	1.173	0.370	0.032	0.049	0.496	0.237	0.500	0.396	0.802	0.288	0.280	Ensemble
D121	0.442	0.968	0.127	0.093	0.142	0.547	0.267	0.500	0.370	0.787	0.250	0.273	Ensemble
D122	0.744	1.694	0.482	0.051	0.076	0.568	0.278	0.500	0.389	0.803	0.272	0.218	Ensemble
D123	0.939	2.229	0.040	0.104	0.264	0.683	0.199	0.139	0.429	0.816	0.307	0.167	Anyone

Preliminary Results
Three classes

- correctly classified instances: 73.66% (cross validation)
- correctly classified instances: 89% (use training set)

Confusion Matrix

Cross validation

a	b	c	<-- classified as
3	5	5	a = Monolithic
2	60	44	b = Ensemble
1	22	158	c = Anyone

Use training set

a	b	c	<-- classified as
13	0	0	a = Monolithic
0	83	23	b = Ensemble
1	9	171	c = Anyone

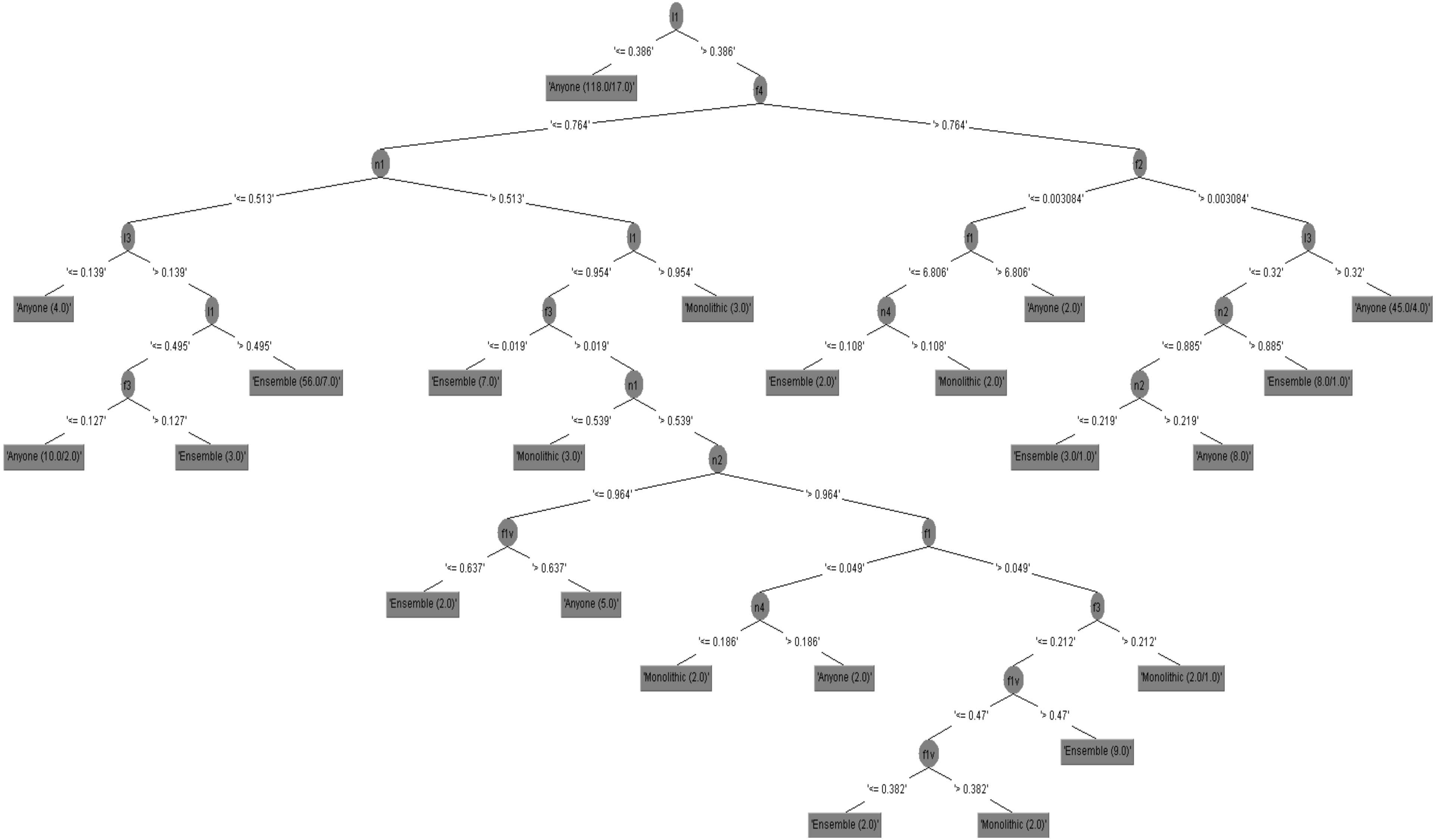


Figure 2. Decision tree of the complexities (2% difference)

Preliminary Results
Two classes

- correctly classified instances: 75,33% (cross validation)
- correctly classified instances: 81,66% (use training set)

Confusion Matrix

Cross validation

a	b		<-- classified as
153	28	|	a = Monolithic
46	73	|	b = Ensemble

Use Training Set

a	b		<-- classified as
169	12	|	a = Monolithic
43	76	|	b = Ensemble

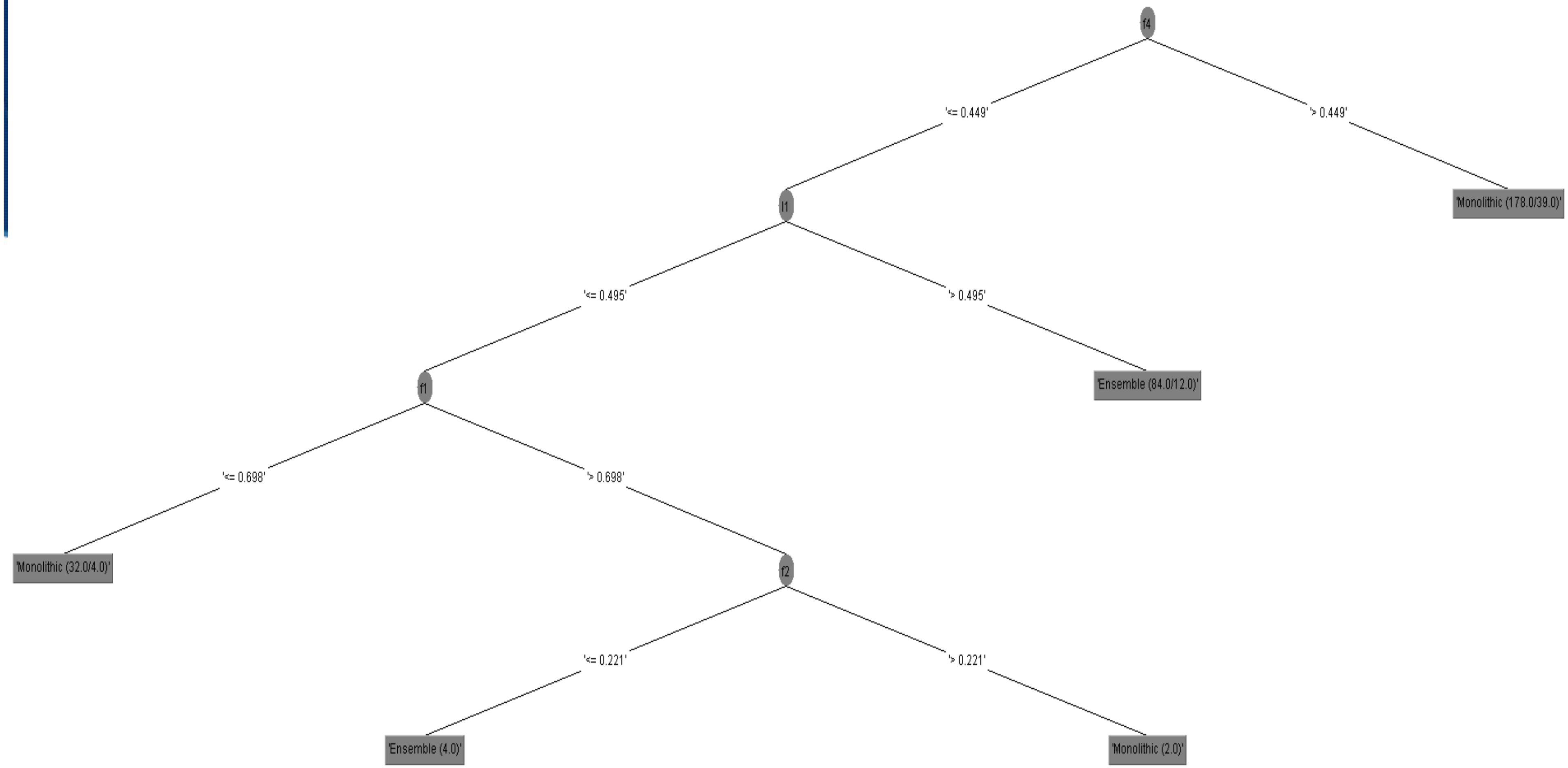


Figure 2. Decision tree of the complexities (2% difference – 2 classes)

Future work

- T-student test in the labeling process: Monolithic, Ensemble or Anyone
- Use leave-one-out strategy
- Evaluate other complexity measures
- Add new datasets in the protocol.