Investigating Exploratory Capabilities of Uncertainty Sampling using SVMs in AL

Dominik Lang, Daniel Kottke, Georg Krempl, Myra Spiliopoulou

Knowledge Management & Discovery
Faculty of Computer Science
Otto-von-Guericke University
Magdeburg, Germany

Outline

- Motivation
- Experimental Evaluation Framework
- Results
- Conclusion

Motivation

Outline

- Motivation
- Experimental Evaluation Framework
- Results
- Conclusion

Motivation

The Exploration-Exploitation Dilemma in AL

Active learning algorithms should

- **explore** the data space to find regions with unexpected labels
- **exploit** the information from acquired labels to refine the classifiers decision boundary

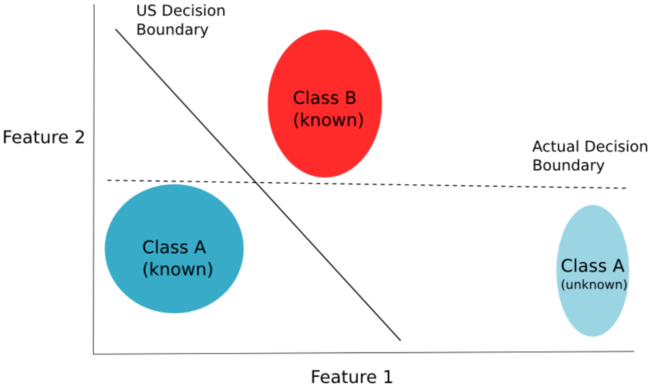
in order to be able to select the most useful instances for training [1, 2]

A basic, common AL approach, **Uncertainty Sampling (US)**, performs **pure exploitation** by selecting instances only based on the classifiers current decision boundary

Motivation cont.

Uncertainty Sampling alone is not enough

- US can become **'locked in'** between known clusters
- This lack of exploration is argued to be the main drawback of US [3]
- It is argued that this might be the reason for the mediocre performance of US compared to pure random sampling [4, 5]



Motivation cont.

Why it might be different with SVMs

- SVMs learn a linear discrimination of the data in a kernel-induced feature space
- In this kernel-induced feature space, US's pure exploitation strategy should perform well [6, 2]
- This might be the reason why US is often used in combination with SVMs
- Does the usage of SVMs fix the problems of US? Can dedicated exploration of the data space be neglected then?

Experimental Evaluation Framework

Outline

- Motivation
- Experimental Evaluation Framework
- Results
- Conclusion

Evaluation Framework

Goals of our work

- Investigate to what extend US covers the original data space when combined with differently tuned SVMs
- Propose an evaluation framework to measure the influence of this coverage (exploration) on the classification performance

Experiments

- Pool-based AL using **US** (simple margin[6]) and **Random Sampling** with 50 labels being acquired by the learner
- As classifiers we use SVMs with **polynomial**, **rbf**, **sigmoid** and **laplacian** kernels
- By performing a grid search on a separate tuning set both, a **best** and a **mediocre** performing hyperparameter set, is selected for each kernel

Evaluation Framework

Evaluation Measures

- **Learning curves** of the classifiers, i.e. the classification accuracy in relation to the number of sampled instances, illustrating the **learners performance**
- **Average minimum euclidean distance** (Eq. 1) of the instances in the set of acquired labels to the instances in the test set, illustrating the **degree of exploration**. Motivated by [7]

$$\frac{1}{|D_{test}|} \sum_{x \in D_{test}} \min_{x_l \in L} ||x - x_l||_2 \tag{1}$$

Results

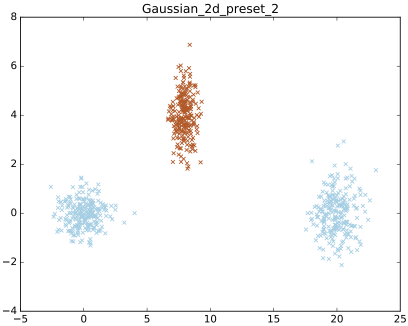
Outline

- Motivation
- Experimental Evaluation Framework
- Results
- Conclusion

Results - Three cluster datasets

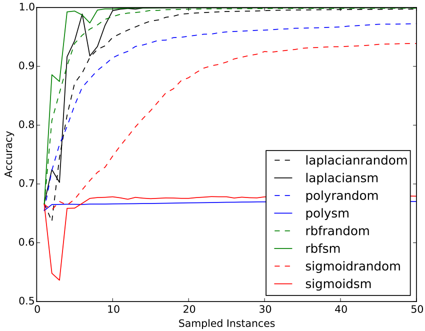
Overview

- Dataset with three clusters like shown earlier
- Only one of the blue clusters is known through initialization
- Sufficient coverage of the data space should find the other cluster

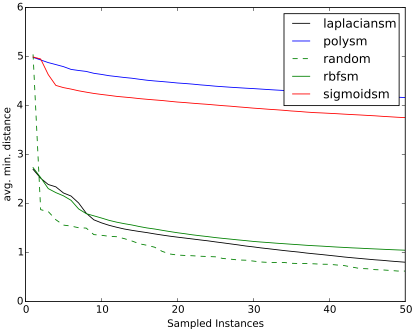


Results - Three cluster datasets cont.

Best hyperparameters



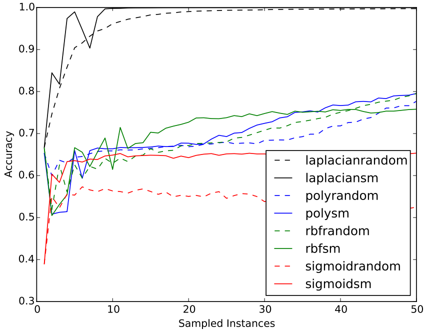
Learning curve



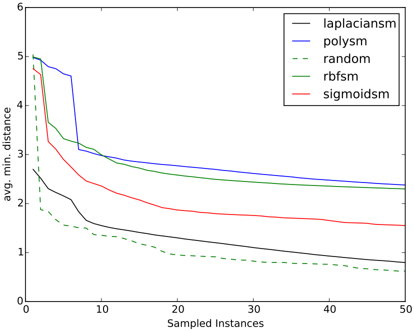
Avg. min. distance curve

Results - Three cluster datasets cont.

Mediocre hyperparameters



Learning curve



Avg. min. distance curve

Results - Three cluster datasets

Summary

- With the best hyperparameter setting:
  laplace and rbf kernel SVMs, consistently discover the unknown cluster, the rest does not
- With the mediocre hyperparameter setting:
  the discovery rate improves, but classifier can't exploit the acquired labels well
- Avg. min. distance measure manages to indicate the degree of exploration well

Params	Kernel/Data	p0	p1	p2
best	laplace	100%	100%	100%
	poly	23%	1%	0%
	rbf	100%	100%	100%
	sigmoid	18%	7%	22%
middle	laplace	100%	100%	100%
	poly	67%	49%	56%
	rbf	64%	56%	56%
	sigmoid	98%	94%	82%

Results - Synthetic datasets cont.

Overview

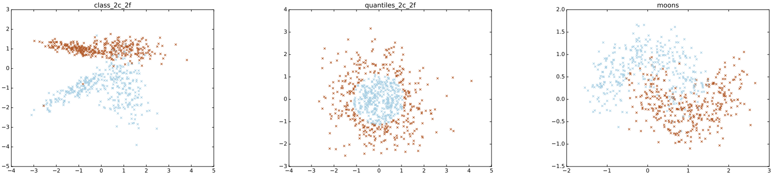
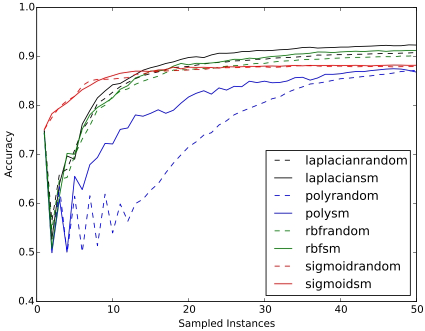


Figure: Examples for the 'classification', 'quantiles' and 'moons' datasets (left to right)

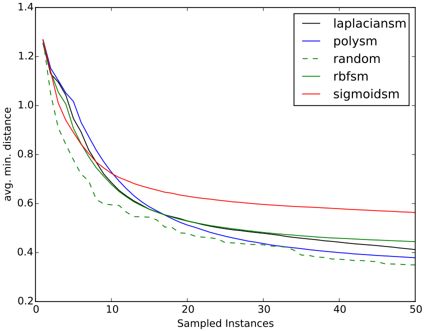
- Three common data models from the scikit-learn library[8]
- No unknown clusters, yet exploration necessary to cover the patterns well

Results - Synthetic datasets cont.

'Classification' - best hyperparameters



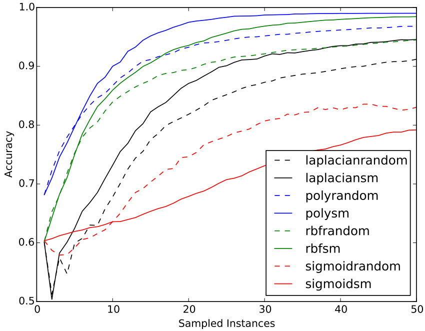
Learning curve



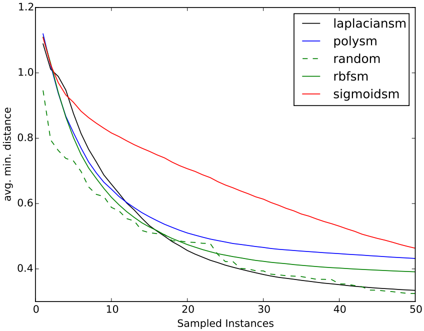
Avg. min. distance curve

Results - Synthetic datasets cont.

'Quantiles' - best hyperparameters



Learning curve



Avg. min. distance curve

Results - Synthetic datasets cont.

Summary

- Fast decrease in avg. min. distance (high exploration) in the early steps indicates a fast improvement in terms of accuracy
- This is also supported by the random sampler performing similar or better in the early steps
- Later it is beaten by US
- No single kernel performs best on all datasets - selecting the right kernel remains a challenge

Results - Real-world datasets

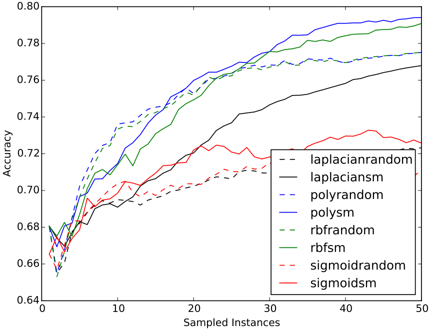
Overview

- 4 datasets taken from the UCI machine learning repository [9]
- Nominal attributes transformed into multiple binary attributes
- Numerical attributes normalized to [0,1]

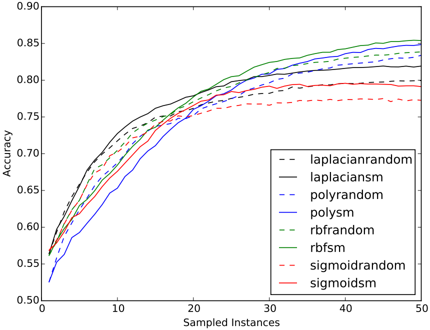
Name	Attributes	Size
Abalone	8	4177
Haberman	3	306
Mammo	11	830
Vertebral	6	310

Results - Real-world datasets cont.

Learning curves



Mammo



Vertebral

Results - Real-world datasets cont.

Summary

- On Mammo and Vertebral the advantage of exploration in the early steps is also visible, random sampling is surpassed only later in the process (except rbf on Vertebral)
- All simple margin learners are outperformed by their random sampling counterparts on Haberman, same applies on Abalone with exception of the laplacian kernel
- Again no single best performing kernel for all data sets

Conclusion

Outline

- Motivation
- Experimental Evaluation Framework
- Results
- Conclusion

Conclusion

Overall summary

- The avg. min. euclidean distance manages to give a good impression of the degree of exploration
- Even by combining US with SVMs, this approach cannot compensate its lack of exploration
- Using non-optimal hyperparameters the exploitation of US+SVM learners shows characteristics similar to exploration, but this is more a misbehavior rather than real exploration. A dedicated exploratory component can not be replaced by this
- The experiments confirm that exploration in the beginning of the AL process is indeed beneficial to the classification performance

Conclusion cont.

Practical implications

- Active Learning approaches with dedicated exploration components should be favored over US
- Test if the dataset is suited for use of US+SVM and carefully select the SVMs hyperparameters
- If possible consider using more complex AL approaches

References

[1] I. Guyon, G. Cawley, G. Dror, V. Lemaire, and A. Statnikov, eds., *Active Learning Challenge*, vol. 6 of *Challenges in Machine Learning*, Microtome Publishing, 2011.

[2] T. Osugi, D. Kim, and S. Scott, "Balancing exploration and exploitation: A new algorithm for active machine learning," in *Data Mining, Fifth IEEE International Conference on*, pp. 8–pp, IEEE, 2005.

[3] G. C. Cawley, "Baseline methods for active learning.," in *Active Learning and Experimental Design@ AISTATS*, pp. 47–57, 2011.

[4] D. Kottke, G. Krempl, D. Lang, J. Teschner, and M. Spiliopoulou, "Multi-class probabilistic active learning," in *ECAI 2016*, vol. 285 of *Frontiers in Artificial Intelligence and Applications*, pp. 586 – 594, 2016.

[5] B. Settles, *Active Learning*. No. 18 in Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan and Claypool Publishers, 2012.

[6] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *The Journal of Machine Learning Research*, vol. 2, pp. 45–66, 2002.

[7] G. Chen, T.-j. Wang, L.-y. Gong, and P. Herrera, "Multi-class support vector machine active learning for music annotation," *International Journal of Innovative Computing, Information and Control*, vol. 6, no. 3, pp. 921–930, 2010.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, and B. Thirion, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[9] A. Asuncion and D. J. Newman, "UCI machine learning repository," 2015.