# AlvsPK Challenge: FACT SHEET

**Title:**

Dimensionality Reduction Techniques

**Name, address, email:**

Stijn Vanderlooy & Laurens van der Maaten
MICC-IKAT, Universiteit Maastricht, P.O. Box 616,
6200 MD Maastricht, the Netherlands

**Acronym of your best entry:**

micc-ikat

**Reference:**

-

**Method:**

Our method to the challenge was to apply dimensionality reduction techniques in order to find a small set of discriminative features. As a preprocessing step we made the data zero mean and unit variance. We then applied principal components analysis and linear discriminant analysis to find a linear subspace of the original data space. In addition, the following six nonlinear dimensionality reduction techniques were applied: isomap, kernel principal components analysis with Gaussian kernel, (Hessian) locally linear embedding, Laplacian eigenmaps, and local tangent space alignment. Algorithms are run with default settings.

For classification we tried the following five classifiers: naïve Bayes, linear discriminant classifier, quadratic discriminant classifier, one nearest neighbour, and least squares support vector machine with Gaussian kernel. Complexity parameter $C$ and bandwidth $h$ are optimized with values $C$ = [1 5 10 20 50 100 500] and $h$ = 0.1 to 1.5 in steps of size 0.1 for each dataset separately using a ten-fold cross validation procedure.

The best overall results, measured by means of error rate, were obtained using the support vector machine on the data representation found by linear discriminant analysis.

**Results:** The reader should also know from reading the fact sheet what the strength of the method is. To that end, we will provide a comparison table in the following format:

Table 1: Our methods best results

| Dataset | Entry name | Entry ID | Test BER | Test AUC | Score | Track |
|---|---|---|---|---|---|---|
| **ADA** | micc-ikat | 1059 | 0.2805 | 0.7195 | 0.953 | Agnos |
| **GINA** | LDA and LSSVM | 945 | 0.1648 | 0.835 | 0.9145 | Agnos |
| **HIVA** | LDA and LSSVM | 945 | 0.3837 | 0.6157 | 0.9237 | Agnos |
| **NOVA** | LDA and LSSVM | 945 | 0.4248 | 0.5741 | 0.9679 | Agnos |
| **SYLVA** | LDA and LSSVM | 945 | 0.0495 | 0.9505 | 0.9397 | Agnos |
| **Overall** | micc-ikat | 1059 | 0.2606 | 0.739 | 0.9398 | Agnos |

Table 2: Winning entries of the AlvsPK challenge

| Best results agnostic learning track | | | | | | |
|---|---|---|---|---|---|---|
| **Dataset** | **Entrant name** | **Entry name** | **Entry ID** | **Test BER** | **Test AUC** | **Score** |
| **ADA** | Roman Lutz | LogitBoost with trees | 13, 18 | 0.166 | 0.9168 | 0.002 |
| **GINA** | Roman Lutz | LogitBoost/Doubleboost | 892, 893 | 0.0339 | 0.9668 | 0.2308 |
| **HIVA** | Vojtech Franc | RBF SVM | 734, 933, 934 | 0.2827 | 0.7707 | 0.0763 |
| **NOVA** | Mehreen Saeed | Submit E final | 1038 | 0.0456 | 0.9552 | 0.0385 |
| **SYLVA** | Roman Lutz | LogitBoost with trees | 892 | 0.0062 | 0.9938 | 0.0302 |
| **Overall** | Roman Lutz | LogitBoost with trees | 892 | 0.1117 | 0.8892 | 0.1431 |
| **Best results prior knowledge track** | | | | | | |
| **Dataset** | **Entrant name** | **Entry name** | **Entry ID** | **Test BER** | **Test AUC** | **Score** |
| **ADA** | Marc Boulle | Data Grid | 920, 921, 1047 | 0.1756 | 0.8464 | 0.0245 |
| **GINA** | Vladimir Nikulin | vn2 | 1023 | 0.0226 | 0.9777 | 0.0385 |
| **HIVA** | Chloe Azencott | SVM | 992 | 0.2693 | 0.7643 | 0.008 |
| **NOVA** | Jorge Sueiras | Boost mix | 915 | 0.0659 | 0.9712 | 0.3974 |
| **SYLVA** | Roman Lutz | Doubleboost | 893 | 0.0043 | 0.9957 | 0.005 |
| **Overall** | Vladimir Nikulin | vn3 | 1024 | 0.1095 | 0.8949 | 0.095967 |

The table will be filled out after the challenge is over by the organizers. Comment about the following:

- <u>quantitative advantages</u> (e.g. compact feature subset, simplicity, computational advantages)
- <u>qualitative advantages</u> (e.g. compute posterior probabilities, theoretically motivated, has some elements of novelty).

**Code:** If CLOP or the Spider were used, fill out the table:

| **Dataset** | **Spider command used to build the model** |
|---|---|
| **ADA** | |
| **GINA** | |
| **HIVA** | |
| **NOVA** | |
| **SYLVA** | |

If new Spider functions were written or if CLOP or the Spider were not used, briefly explain your implementation. Provide a URL for the code (if available). Precise whether it is a push-button application that can be run on benchmark data to reproduce the results, or resources such as modules or libraries.

**Keywords:** Put at *least one keyword in each category*. Try some of the following keywords and add your own:

- Preprocessing or feature construction: centering, scaling, standardization, PCA.
- Feature selection approach: filter, wrapper, embedded feature selection.
- Feature selection engine: correlation coefficient, Relief, single variable classifier, mutual information, miscellaneous classifiers, including neural network, SVM, RF.
- Feature selection search: feature ranking, ordered FS (ordered feature selection), forward selection, backward elimination, stochastic search, multiplicative updates
- Feature selection criterion: training error, leave-one-out, K-fold cross-validation.
- Classifier: neural networks, nearest neighbors, tree classifier, RF, SVM, kernel-method, least-square, ridge regression, L1 norm regularization, L2 norm regularization, logistic regression, ensemble method, bagging, boosting, Bayesian, transduction.
- Hyper-parameter selection: grid-search, pattern search, evidence, bound optimization, cross-validation, K-fold.
- Other: ensemble method, transduction.