PERFORMANCE PREDICTION CHALLENGE: FACT SHEET FORMAT (1 to 2 pages)

Title: Kernel Classifier
Name, address, email: Wei Chu, Center of Computational Learning Systems, Columbia University, chuwei@cs.columbia.edu
Acronym of your best entry: SVM/GPC

Reference:
I have not written up a document on the procedure I applied. For reference or code, see http://www.gatsby.ucl.ac.uk/~chuwei/.

Method:
Classifiers with different kernels were trained on the five datasets respectively. More specifically, I tried two kernel classifiers, support vector classifier and Gaussian process classifier. Profile of my methods as follows:

- Preprocessing: Training and test data, except NOVA, were jointly normalized to zero-mean and unit variance.
- Feature selection: On Gina, ranksum test was applied that reduced 970 features to 453 features; On Hiva, hypergeometry test was applied that reduced 1617 features to 425. On other datasets, we used all normalized features.
- Classification
  - A linear support vector classifier was used on Nova; Gaussian process classifier with a Gaussian kernel was used on Sylva; while non-linear support vector classifiers with Gaussian kernels were used on other datasets.
  - Did you use ensemble methods? No.
  - Did you use "transduction" or learning from the unlabeled test set? No.
- Model selection/hyperparameter selection: Model evidence was used to decide optimal values of hyperparameters, whereas 10-fold cross validation was applied for model selection in support vector classifiers.
- Performance prediction guess. (How did you compute the value in the .guess file). Validation outputs were used for support vector classifiers to estimate predictive performance, while leave-one-out validation outputs were used in Gaussian process classifiers.

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

Dataset	Our best entry					The challenge best entry				
	Test AUC	Test BER	BER guess	Guess error	Test score	Test AUC	Test BER	BER guess	Guess error	Test score
ADA	0.8101	0.1899	0.174	0.016	0.2059					
GINA	0.9619	0.0381	0.0379	0.0002	0.0381					
HIVA	0.7095	0.2905	0.27	0.0205	0.31					

NOVA	0.952	0.048	0.045	0.003	0.0503					
SYLVA	0.99	0.01	0.0076	0.0024	0.0124					
Overall	0.8847	0.1153	0.1069	0.0084	0.1233					As(Rk)

For the overall performance, provide the average test score (As) and in parentheses the average rank (Rk).

- challenge performances (group rank is fifth).
- quantitative advantages (e.g.)
- qualitative advantages (Gaussian process classifiers provide predictive probability.).

Code: An implementation of Gaussian process classifiers can be found http://www.gatsby.ucl.ac.uk/~chuwei/, which is designed for more general cases of ordinal regression. Binary classification is treated as a special case of ordinal regression.

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

- Preprocessing or feature construction: standardization.
- Feature selection approach: filter.
- Feature selection engine: miscellaneous classifiers.
- Feature selection search: feature ranking.
- Feature selection criterion: K-fold cross-validation.
- Classifier: SVM, kernel-method, Gaussian processes.
- Hyper-parameter selection: grid-search, evidence, K-fold cross-validation.
- Other: No.

EXAMPLE FACT SHEET:

Title: Ensemble of neural nets
Name, address, email: John Doe, University of Nowhere, United Zone, doe@nowhere.edu.
Acronym of your best entry: ST+NN+5CV

Reference: Ensemble of neural nets, John Doe et al. In Proceedings IJCNN06, to appear. http://www.nowhere.edu/~doe/ijcnn06.pdf.

Method:
Our method uses as preprocessing a simple centering and rescaling of the data (standardization). In some experiments, we used PCA as a further step to generate features. For feature selection, we used simple feature ranking with correlation coefficients and selected the number of features with 5-fold cross-validation (CV), after ranking the features using the whole training dataset. We use ensembles of neural networks for classification. All hyper-parameters are adjusted after feature selection, using again 5-fold CV with the same training data. Our test BER prediction is based also on 5-fold CV, but we did a separate drawing of the folds after the hyper-parameters were selected.

Results:
In the challenge, we rank 10^{th} as a group and our best entry is the 23^{rd}, according to the average rank computed by the organizers. We further analyzed the results of the challenge with other objectives and demonstrated that our method yields the best results for compact feature subsets (<10% of the total number of features). Our model selection and performance prediction methods are particularly accurate. They allowed us to also win the KDD06 challenge. We also conducted comparison experiments with single neural networks as classifiers and found that ensemble techniques significantly improve the results.

Dataset	Our best entry					The challenge best entry				
	Test AUC	Test BER	BER guess	Guess error	Test score	Test AUC	Test BER	BER guess	Guess error	Test score
ADA										
GINA										
HIVA										
NOVA										
SYLVA										
Overall					As(Rk)					As(Rk)

Code: Our implementation was done in Lisp using the Lush software http://www.gnu.org/directory/lush.html. Our scripts reproducing the benchmark results are available at http://the-great.university.edu/the-clean-code.html.

Keywords: standardization, PCA, filter, correlation coefficient, feature ranking, K-fold cross-validation, neural networks, ensemble method.