AlvsPK Challenge: FACT SHEET

Title: ***Dr.***
Name, ***Gavin Cawley***
address, ***School of Computing Sciences, University of East Anglia***
email: gcc@cmp.uea.ac.uk

Acronym of your best entry:
Interim all prior (fact sheet describes best individual models rather than interim all prior, but the differences are minor)

Reference:

[1] Cawley, G. C. and Talbot, N. L. C., "Agnostic Learning via Prior Knowledge in the Design of Kernel Machines", Proc. IJCNN-2007, Orlando, Florida, 12-17 August 2007.

[2] Cawley, G. C., Janacek, G. J. and Talbot, N. L. C., "Generalised Kernel Machines", Proc. IJCNN-2007, Orlando, Florida, 12-17 August 2007.

Method:

Kernel Ridge Regression (KRR) models (a.k.a. LS-SVM) were used in all experiments, with the hyper-parameters set so as to minimize the virtual leave-one-out estimate of the squared error (i.e. Allen's PRESS statistic). Model selection was performed using the Nelder-Mead simplex optimization algorithm. The classification threshold was set so as to minimize the LOO BER. No explicit feature selection was used in any of the experiments.

ADA: Features with age, capital-gain and capital-loss have skew distributions. A Box-Tidwell power transformation was used (10th root) to stop extreme values of these features dominating the kernel function. Otherwise the features were as supplied, except that continuous features were standardized. Various kernels were used, the best results being obtained with the Automatic Relevance Determination (ARD) kernel, i.e. an RBF kernel with a separate scaling factor for each input feature.

GINA: The input features were scaled to lie in the range [0-1], for convenience. An hierarchical model was used. In the first level, 25 KRR machines were trained to distinguish between each even-odd pair of digits. At the second layer a KRR model is used to perform the overall even-odd classification using the outputs of the models in the first layer. The top level KRR was trained using the VLOO output of the first level machines in order to provide a statistically pure dataset and prevent over-fitting. The best results were obtained using a Multiple Receptive Field (MRF) kernel in the first level and an ARD kernel in the second. The MRF kernel is essentially an RBF kernel where the input features are weighted according to seven adaptive Gaussian receptive fields. During model selection, the amplitude, (x, y)-coordinates and spherical variance of the Gaussians are adapted to concentrate the sensitivity of the kernel on the most important areas of the image. This allows a more flexible kernel than a single RBF, while keeping the number of hyper-parameters manageable. It also builds in the prior knowledge that the importance of the inputs ought to be a fairly smooth function of the position on the image.

HIVA: The ChemTK suite (www.chemaxon.com) was used to generate a set of 1024-bit binary chemical structure and pharmacore fingerprints (using the generatemd tool), which are used for virtual screening in e.g. drug discovery. These are features of the 2-D graph representation of the chemical structure. A quadratic kernel provided the best results.

NOVA: Words on a stop list commonly used in text classification problems were deleted. These words are thought to be too short or too common to convey any discriminative power in any application. A stemmer developed at UEA was then used to remove affixes and suffixes to leave the root of the word, which conveys the bulk of the semantic meaning (e.g. “fisher”, “fishing” etc. become “fish”). A conventional tf-idf encoding is then used where each input feature represents the frequency of the word in the document divided by the log of the frequency of documents in the corpus containing this word. A quadratic kernel provided the best results. We also experimented with automatic spell-checking as a pre-processing step, reasoning that USENET messages posted in haste are likely to contain many spelling mistakes, and many noisy features might be eliminated by re-mapping incorrect spellings. However, it seems that the automatic spell checking proved too aggressive, and the results were no better. We also tried making hierarchical classifiers as the targets again represent a compound concept, so we made classifiers that distinguished between individual pairs of USENET groups from the positive and negative classes, however this also failed to improve results.

SYLVA: The input data represent two patterns of the same class. The pairing of these patterns is arbitrary, so we separated out the data to create twice as many patterns. In classifying the test data, we run the classifier twice and classify the pattern as belonging to the negative class if either of the sub-patterns is classified as negative. Investigating the data, we found that Ponderosa Pine only grow in the Comanche Peak and Cache La Poudra wilderness areas and in only 13 of the 40 soil types. We think this is because Ponderosa Pine prefers to grow at relatively high elevations. Pre-classifying the training data using these features leaves only 1335 difficult training patterns to be classified using a KRR model. Various kernels were used, with a linear kernel providing the best results.

Results:

Table 1: Our methods best results

Dataset	Entry name	Entry ID	Test BER	Test AUC	Score	Track
ADA	Ada interim #5	752	0.169961	0.914945	0.008180	prior
GINA	Gina final #15	887	0.019227	0.997356	0.004274	prior
HIVA	Hiva interim #3	813	0.263568	0.768676	0.004016	prior
NOVA	Nova #2b	731	0.036739	0.993509	0.006410	prior
SYLVA	Sylva test #5	816	0.005928	0.998993	0.010050	prior
Overall	Interim all prior	818	0.103463	0.9332	0.037628	prior

Table 2: Winning entries of the AlvsPK challenge

<table>
<tr><th colspan="7">Best results agnostic learning track</th></tr>
<tr><th>Dataset</th><th>Entrant name</th><th>Entry name</th><th>Entry ID</th><th>Test BER</th><th>Test AUC</th><th>Score</th></tr>
<tr><td><b>ADA</b></td><td>Roman Lutz</td><td>LogitBoost with trees</td><td>13, 18</td><td>0.166</td><td>0.9168</td><td>0.002</td></tr>
<tr><td><b>GINA</b></td><td>Roman Lutz</td><td>LogitBoost/Doubleboost</td><td>892, 893</td><td>0.0339</td><td>0.9668</td><td>0.2308</td></tr>
<tr><td><b>HIVA</b></td><td>Vojtech Franc</td><td>RBF SVM</td><td>734, 933, 934</td><td>0.2827</td><td>0.7707</td><td>0.0763</td></tr>
<tr><td><b>NOVA</b></td><td>Mehreen Saeed</td><td>Submit E final</td><td>1038</td><td>0.0456</td><td>0.9552</td><td>0.0385</td></tr>
<tr><td><b>SYLVA</b></td><td>Roman Lutz</td><td>LogitBoost with trees</td><td>892</td><td>0.0062</td><td>0.9938</td><td>0.0302</td></tr>
<tr><td><b>Overall</b></td><td>Roman Lutz</td><td>LogitBoost with trees</td><td>892</td><td>0.1117</td><td>0.8892</td><td>0.1431</td></tr>
<tr><th colspan="7">Best results prior knowledge track</th></tr>
<tr><th>Dataset</th><th>Entrant name</th><th>Entry name</th><th>Entry ID</th><th>Test BER</th><th>Test AUC</th><th>Score</th></tr>
<tr><td><b>ADA</b></td><td>Marc Boulle</td><td>Data Grid</td><td>920, 921, 1047</td><td>0.1756</td><td>0.8464</td><td>0.0245</td></tr>
<tr><td><b>GINA</b></td><td>Vladimir Nikulin</td><td>vn2</td><td>1023</td><td>0.0226</td><td>0.9777</td><td>0.0385</td></tr>
<tr><td><b>HIVA</b></td><td>Chloe Azencott</td><td>SVM</td><td>992</td><td>0.2693</td><td>0.7643</td><td>0.008</td></tr>
<tr><td><b>NOVA</b></td><td>Jorge Sueiras</td><td>Boost mix</td><td>915</td><td>0.0659</td><td>0.9712</td><td>0.3974</td></tr>
<tr><td><b>SYLVA</b></td><td>Roman Lutz</td><td>Doubleboost</td><td>893</td><td>0.0043</td><td>0.9957</td><td>0.005</td></tr>
<tr><td><b>Overall</b></td><td>Vladimir Nikulin</td><td>vn3</td><td>1024</td><td>0.1095</td><td>0.8949</td><td>0.095967</td></tr>
</table>

quantitative advantages (e.g. compact feature subset, simplicity, computational advantages)

KRR models with VLOO based model selection seems to provide good results for all datasets, providing a suitable kernel can be found.

- qualitative advantages (e.g. compute posterior probabilities, theoretically motivated, has some elements of novelty).

KRR is very simple and easily implemented. The automated model selection process is very handy as it enables the method to be used safely by non-specialists. Plenty of theoretical justification for kernel methods, regularization etc.

Code:

The models were implemented using a development version of a MATLAB toolbox for Generalised Kernel Machines [2], which will be made available shortly.

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

- Preprocessing or feature construction: standardization, Box-Tidwell transformation.
- Feature selection approach: embedded feature selection.
- Feature selection engine: none
- Feature selection search: none
- Feature selection criterion: none
- Classifier: Kernel Ridge Regression/LS-SVM/Regularisation Network
- Hyper-parameter selection: Virtual LOO, PRESS, Nelder-Mead simplex