Results of the KDD'99 Classifier Learning

Charles Elkan

University of California
San Diego

elkan@cs.ucsd.edu

The task for the classifier learning contest organized in conjunction with the KDD'99 conference was to learn a predictive model (i.e. a classifier) capable of distinguishing between legitimate and illegitimate connections in a computer network. Here is a detailed description of the task. The training and test data were generously made available by Prof. Sal Stolfo of Columbia University and Prof. Wenke Lee of North Carolina State University.

In total 24 entries were submitted for the contest. There was a data quality issue with the labels of the test data, which fortunately was discovered by Ramesh Agarwal (IBM Fellow) and Mahesh Joshi (University of Minnesota Ph.D. candidate) before results were announced publicly. Ramesh Agarwal and Mahesh Joshi have analyzed the data quality issue with great detail and precision, so we are confident that the test data with corrected labels is now correct. Other participants also detected and analyzed the data quality issue, including Itzhak Levin of LLSoft, Inc.

It is important to note that the data quality issue affected only the labels of the test examples. The training data was unaffected, as was the unlabeled test data. Therefore it was not necessary to ask participants to submit recomputed entries.

Each entry was scored against the corrected test data by a scoring awk script using the published cost matrix (see below) and the true labels of the test examples.

THE WINNING ENTRIES

The winning entry was submitted by Dr. Bernhard Pfahringer of the Austrian Research Institute for Artificial Intelligence. The method used is described at this web site: http://www.ai.univie.ac.at/~bernhard/kddcup99.html

Second-place performance was achieved by Itzhak Levin from LLSoft, Inc. using the tool Kernel Miner. For details see http://www.llsoft.com/kdd99cup.html.

Third-place performance was achieved by Vladimir Miheev, Alexei Vopilov, and Ivan Shabalin of the company MP13 in Moscow, Russia. For details, see the following web site: http://www-cse.ucsd.edu/users/elkan/mp13method.html

The difference in performance between the three best entries is only of marginal statistical significance; see below for a discussion of this question. Congratulations to all the winners and thanks to all the participants!

PERFORMANCE OF THE WINNING ENTRY

The winning entry achieved an average cost of 0.2331 per test example and obtained the following confusion matrix:

Actual → Prediction ↓	0	1	2	3	4	%corr-ect
0	60262	243	78	4	6	99.5%
1	511	3471	184	0	0	83.3%
2	5299	1328	223226	0	0	97.1%
3	168	20	0	30	10	13.2%
4	14527	294	0	8	1360	8.4%
%correct	74.6%	64.8%	99.9%	71.4%	98.8%	

In the table above the five attack categories are numbered as follows:

0	normal
1	probe
2	denial of service (DOS)
3	user-to-root (U2R)
4	remote-to-local (R2L)

Individual attack types were placed in the five categories using a categorization awk script made publically available at: http://www-cse.ucsd.edu/users/elkan/tabulate.html .

The top-left entry in the confusion matrix shows that 60262 of the actual "normal" test examples were predicted to be normal by this entry. The last column indicates that in total 99.5% of the actual "normal" examples were recognized correctly. The bottom row shows that 74.6% of test examples said to be "normal" were indeed "normal" in reality.

STATISTICAL SIGNIFICANCE

Non-winning entries obtained an average cost per test example ranging from 0.2356 to 0.9414. In rank order, the average costs obtained by all 24 entries are:

0.2331	0.2474	0.2552	0.3344
0.2356	0.2479	0.2575	0.3767
0.2367	0.2523	0.2588	0.3854
0.2411	0.2530	0.2644	0.3899
0.2414	0.2531	0.2684	0.5053
0.2443	0.2545	0.2952	0.9414

It is difficult to evaluate exactly the statistical significance of differences between entries. However it is important not to read too much into differences that may well be statistically insignificant, i.e. due to randomness in the choice of training and test examples.

Statistical significance can be evaluated roughly as follows. The mean score of the winning entry as measured on the particular test set used is 0.2331, with a measured standard deviation of 0.8834. Assuming that the mean is computed from *N* independent observations, its standard error is 0.8334/sqrt(*N*). The test dataset contains 311,029 examples, but these are not all independent. An upper bound on the number of independent test examples is the number of distinct test examples, which is 77291. The standard error of the winning mean score is then at least 0.8334/sqrt(77291) = 0.0030.

If the threshold for statistical significance is taken to be two standard errors, then the winning entry is significantly superior to all others except the second and third best.

The first significant difference between entries with adjacent ranks is between the 17th and 18th best entries. This difference is very large: 0.2952 - 0.2684 = 0.0268, which is about nine standard errors. One can conclude that the best 17 entries all performed well, while the worst 7 entries were definitely inferior.

A SIMPLE METHOD PERFORMS WELL

Most participants achieved results no better than those achievable with very simple methods. According to its author, one entry was simply "the trusty old 1-nearest neighbor classifier." This entry scored 0.2523 with confusion matrix at the bottom of this page.

Only nine entries scored better than 1-nearest neighbor, of which only six were statistically significantly better. Compared to 1-nearest neighbor, the main achievement of the winning entry is to recognize correctly many more "remote-to-local" attacks: 1360 compared to 95. This success is impressive, since only 0.23% of training examples were in this category compared to 5.20% of the test examples. The detailed task description did point out that "It is important to note that the test data is not from the same probability distribution as the training data".

COST-BASED SCORING AND TRAINING VS. TEST DISTRIBUTION

The cost matrix used for scoring entries was given as

	normal	probe	DOS	U2R	R2L
normal	0	1	2	2	2
probe	1	0	2	2	2
DOS	2	1	0	2	2
U2R	3	2	2	0	2
R2L	4	2	2	2	0

Here, as in the confusion matrices above, columns correspond to predicted categories, while rows correspond to actual categories. The cost matrix says that the cost incurred by classifying all examples as "probe" is not much over 1.0, if the categories U2R and R2L are rare. These two categories are in fact rare in the training dataset, and the mean cost incurred by classifying all examples as "probe" is 0.994 on the training dataset.

Some basic domain knowledge about network intrusions suggests that the U2R and R2L categories are intrinsically rare. The actual distributions of attack types in the training and test 10% datasets are:

	training	test
0:	19.69%	19.48%
1:	0.83%	1.34%
2:	79.24%	73.90%
3:	0.01%	0.07%
4:	0.23%	5.20%

Together, the U2R and R2L attacks constitute 5.27% of the test dataset, which is a substantial increase compared to the training dataset, but still a small fraction. The mean cost incurred by classifying all test examples as "probe" is 0.5220 on the test dataset, which is better than the average cost achieved by the worst entry submitted.

Actual →	0	1	2	3	4	%correct
Prediction ↓						
0	60322	212	57	1	1	99.6%
1	697	3125	342	0	2	75.0%
2	6144	76	223633	0	0	97.3%
3	209	5	1	8	5	3.5%
4	15785	308	1	0	95	0.6%
%correct	72.5%	83.9%	99.8%	88.9%	92.2%	