OUTLIER DETECTION IN BENCHMARK CLASSIFICATION TASKS

Hongyu Li & Mahesan Niranjan

Department of Computer Science, The University of Sheffield, Regent Court, S1 4DP, Sheffield, UK
H.Li/M.Niranjan@dcs.shef.ac.uk

ABSTRACT

We present a new outlier detection method which is appropriate for classification problems. It combines estimating the overall probability density and sequential ranking of the data according to observed changes in performance on validation sets. The method has been implemented on ten widely used benchmark datasets and a spam email filtering application. Evaluated by six popular machine learning methods, classification performances are shown to improve after removing outliers in comparison to removing the same number of examples at random from the datasets.

1. INTRODUCTION

Inference problems solved by data driven models involve the collection of a certain amount of data that characterize a problem domain. Reliability with which we can estimate parameters of a model, either in the form of a probabilistic generative model or in the form of a function approximator in the space of the data, and the subsequent use of this model in making predictions depends crucially on the collected data being representative about the underlying mechanism. When the data is not representative, our ability to learn a model that can generalize to unseen data is severely restricted. By this notion of data not being representative we do not refer to the variability in the data (systematic - e.g. coarticulation in speech, or random - e.g. instrument errors). These are factors what are handled by the modelling process. Occasional large errors, atypical of the process, such as labelling errors in a classification setting or contamination of a sample in a series of chemical or biological experiments are regarded as *outliers* with respect to the process that generates data and with respect to the data driven model we chose to apply.

In the comparison of machine learning algorithms, it is customary to evaluate performances on several benchmark datasets which are publicly available within the research community. In most of these cases we only have access to the data and can neither refer back to the originators of the data nor repeat physically the experiments that generated the measurements. In such a situation it is important to be confident that the datasets do not include outliers so that comparisons of algorithm performance are meaningful.

Tarassenko [1] has successfully considered the problem of outlier detection in engineering and health monitoring applications, working with respect to an assumed probability density model. Novelty detection is achieved by fitting a Gaussian mixture model through data corresponding to the normal cases, or, in the case of engines, normal operating conditions. Once the mixture model is trained, a test data is said to be abnormal if its likelihood is lower than a particular threshold with respect to the model.

Density estimation alone is not sufficient to define and detect outliers in classification problems. A Gaussian mixture model of the entire data would ignore class labels, occasional errors in which may well be the outliers. For classification problems, the main effect of outliers is observed as variability in classification accuracy evaluated on a validation set, when we randomly partition the data into training and validation sets. In a given partition, the presence of outlier data in the validation set has small effect on accuracy, but if the outlier data were to be in the training set, validation set accuracy will be affected a lot more because the resulting estimate of the classifier parameters will be poor.

Detecting and dealing with outliers has been considered in the statistical literature [2], mainly in the context of regression problems. The classic way of dealing with outliers is to make the inference process robust to their presence (*e.g.* by the use of L_1 norm in linear regression, which attenuates large errors contributed by outliers). Robustness properties of Bayesian methods to the presence of outliers have been considered in [3]. Derivation of robust regression models can usually be associated with an appropriate assumption about a noise model (*e.g.* the use of L_1 norm assumes additive noise of Laplacian distribution).

In classification problems, however, the case for detecting outliers is more compelling because outliers may arise due to labelling errors. In most data gathering environments errors on inputs, or features, tend to be instrument noise which can be modelled as a probability density. Errors in outputs, or labels, tend to be occasional human errors in interpreting the data to associate a label. Literature on detecting outliers in classification problems is not vast, making this a topic which has not been given the seriousness it deserves. That the problem exists is obvious since most publications that report performance of a new machine learning algorithm on a dataset

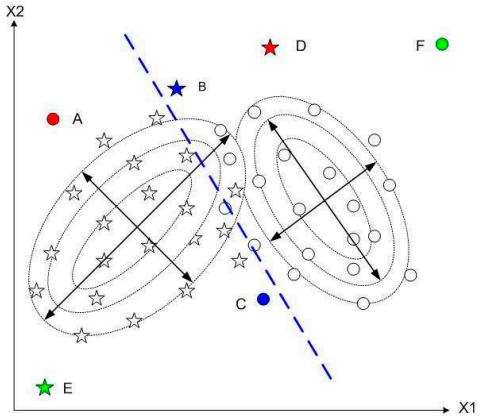

Fig. 1. Illustration of outliers in a two-class classification problem in which stars and circles are used to denote the data. Consider the filled patterns A, B, C, D, E & F. Of these, A, D, E & F are examples of low likelihood with respect to a probability density of all the data. A Gaussian mixture model with two components, for example, will detect these. A, B, C and D are examples that can cause large errors if they are in a bootstrap sample of the training set as they appear mislabelled. These will be detected by the sequential backward elimination procedure discussed in this paper. Our definition of outliers, with respect to the classification problem, combines the two and labels A & D as outliers.

quantify the variability in performance when the data is randomly partitioned into different training and validation sets.

2. METHODOLOGY

Our working definition of outliers is a stringent one, illustrated in Fig.1, examples that have a low likelihood with respect to the entire dataset and at the same time contribute to large generalization errors. We use Gaussian mixture models with two components and full covariance matrices fitted to the entire dataset to identify examples of low likelihood. Examples are ranked in ascending order of likelihood with respect to the GMM model. The model is trained using the expectation maximization (EM) algorithm and examples that fall in the lowest 10% of all the likelihoods are considered for further processing. Fig. 3 (a) shows the ordered likelihoods for the `sonar` dataset from the machine learning repository at University of California, Irvine [4].

The sequential backward elimination procedure is shown in pseudo code in Fig.2. It employs Fisher linear discriminant analysis as base classifier and goes through a series of estimation of a discriminant direction on a training set and subsequent evaluation of its performance on a validation set. Each of the N_{runs} runs of the algorithm will pick out an ordered set of N_{remove} data points as candidate outliers, removal of which produced the best validation set performance, quantified by the F_1 measure, striking a balance between the two types of classification errors in two class problems. Similar to the GMM, we set N_{remove} as 10% of total number of patterns. We compute the frequency of trials of each pattern in the matrix D and select the most frequent ones as candidate outliers. Where there is a tie in frequencies, we use position in ranking to break it (i.e., the higher ranked patterns are picked first.)

Finally, patterns which appear on the low likelihood list and occur with higher frequency in the elimination process are declared as outliers. Results of such ranks for the `sonar` dataset are shown in Table 2. Pattern 3 (first row) is picked as the first outlier as it has the 3^{rd} lowest likelihood and it ranked the 4^{th} in terms of frequency of appearance across the 50 runs of the backward elimination algorithm.

Input: Number of candidate outliers N_{remove};
Number of ranking iterations N_{runs};
Output: A matrix $D_{\{N_{\text{runs}} \times N_{\text{remove}}\}}$ of data IDs the columns of the matrix are ordered

for $p = 1 : N_{\text{runs}}$
  Randomly partition the data set into $\mathcal{S}^{\text{tr}}$ $\mathcal{S}^{\text{va}}$;
  Empty Elimination set $\mathcal{E} = \{\text{empty}\}$;
  for $k = 1 : N_{\text{remove}}$
    for $i = 1 : |S^{\text{tr}}|$
      $\underline{w}$ =train Fisher LDA on $\mathcal{S}^{\text{tr}}_{\setminus i}$;
      $e(i)$ = performance of $\underline{w}$ on $\mathcal{S}^{\text{va}}$;
    end
    Find pattern j for which $e(j)$ is minimal;
    Move j from $\mathcal{S}^{\text{tr}}$ to $\mathcal{E}$;
  end
end

Fig. 2. Pseudocode for the sequential ranking algorithm. $|\mathcal{S}^{\text{tr}}|$ denotes the size of the set S^{tr} and $S^{\text{tr}}_{\setminus i}$ denotes the set S^{tr} with the i^{th} element removed.

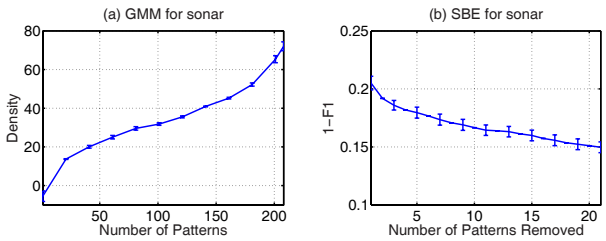

Fig. 3. Data ranked by log likelihood of a GMM, (a), and validation set error during backward elimination, (b), for the `sonar` dataset. The small vertical bars represent standard deviations across 50 different runs.

Pattern ID	GMM RANKING	SBE RANKING
3	3	4
101	9	39
23	11	15
172	13	40
134	19	18
131	20	21
19	-	1
91	1	-
22	-	2
166	2	-
196	-	3

Table 1. Outliers Selection on the `sonar` Dataset. Patterns above the separation line are outliers detected by two approaches; Patterns below the line are only detected by one approach. The first column is an identifier for the pattern picked out as outlier. The second and third columns indicate the position in the two ranking procedures.

Dataset	No. of Patterns	No. of Attributes	No. of Outliers
Soybean-small	47	35	0
Sonar	208	60	6
Glass	214	9	4
Ionosphere	351	34	11
Pima Indians	768	8	13
Vehicle	846	18	10
SPECT heart	267	22	0
Breast Cancer	569	30	8
Iris plant	150	4	0
Letter-Recogn(M N)	1575	16	6

Table 2. Selected datasets from UCI repository and the number of examples detected as outliers.

3. EXPERIMENTAL ILLUSTRATIONS

We applied the outliers detection method to ten datasets from the UCI repository and a spam email filtering problem to which we artificially injected outliers (*i.e.* emails from outside the original domain).

3.1. UCI Datasets

Some aspects and the number of examples identified as outliers by applying our methodology for ten of the benchmark UCI problems are shown in Table 2.

In order to confirm that the detected examples can indeed be regarded as outliers for the problem, we compare classification performance when these are removed from the training sets using different classifiers not used in the detection process. Again we use the `sonar` dataset and consider Naive Bayes (NB), SVM, RandomForest (RF), AdaBoost (AB), RBF Network (RBF) and C4.5, all implemented in the Weka software package [5], with default settings for all tunable parameters. Ten fold cross validation was used to measure improvements in generalization and the results are shown in Table 3. It is found that good performances are achieved after removing six outliers. There is significant improvement on the performance by the NB, AB, RBF and C4.5. Note we do not get an improvement for the higher performing classifiers (SVM & RF). This is probably due to the robustness of these classifiers and the fact that we never reduce empirical error on training data to zero.

	NB	SVM	RF	AB	RBF	C4.5
Ran	0.651	0.815	0.823	0.743	0.719	0.732
Out	0.684	0.824	0.845	0.761	0.777	0.758
Base	0.652	0.825	0.838	0.739	0.711	0.739

Table 3. Performance of Outlier Detection Method measured by F_1 for `sonar` dataset. "Ran" denotes the performance of the dataset after randomly removing six patterns; "Out" denotes the performance of the dataset after removing six outliers; "Base" is the performance on the original dataset.

We illustrate the data items identified as outliers using projections onto a linear discriminant plane, following [6]. The discriminant direction that maximizes the Fisher ratio is computed by $\underline{d}_1 = \alpha_1 A^{-1}\Delta$, where A is within-class scatter matrix, $\Delta = \{\underline{m}_1 - \underline{m}_2\}$ is the difference in the estimated means. It can also shown that the second discriminant directions, projections upon which will give the best separation for the corresponding subspaces is given by $\underline{d}_2 = \alpha_2\{A^{-1} - b_1[A^{-1}]^2\}\Delta$, where, $b_1 = \frac{\Delta^T(A^{-1})^2\Delta}{\Delta^T(A^{-1})^3\Delta}$. $\underline{d}_2$ is obtained by maximizing the Fisher ratio subject to the constraint that d_2 should be orthogonal to $\underline{d}_1$. Thus the two directions define a plane in the space containing the data, linear projections onto which will be maximally separated, forming a convenient mechanism to visualize the data in two dimensions. In Fig 4, we show these discriminant plane projection for the sonar dataset.

3.2. Spam Filtering

We consider a spam email filtering problem defined by the `Ling-Spam` corpus[1] in which emails are pre-processed as 505 dimensional Term Frequency-Inverse Document Frequency (TF-IDF) vectors, usual in text categorization tasks. We induced outliers into this dataset by taking 40 emails from our mailboxes, 20 of which were spam. When including these with the dataset, we deliberately mislabelled 10 of the 20 spam and 20 non-spam emails. The resulting dataset has labels shown in Table 4. To reduce the size of the problem, we took a random subset of 500 examples form `Ling-Spam` and

[1]The corpus is publicly available at http://www.aueb.gr/users/ion/data/lingspam_public.tar.gz.

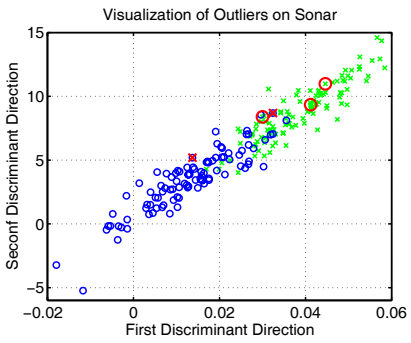

Fig. 4. Projections of the data and detected outliers for the `sonar` dataset on the Fisher discriminant plane. Circles and crosses denote patterns of two classes; the red circles and crosses are outliers.

E-Mail ID	E-Mail
1-10	correctly labelled non-spam email from CS
11-20	mislabelled non-spam email from CS
21-30	correctly labelled spam email from CS
31-40	mislabelled spam email from CS
41-540	correctly labelled email from `Ling-Spam`

Table 4. The Pattern Identifiers (ID) of outlier detection experiments. Emails whose ID range from 1 to 40 are emails from our mailboxes at the University of Sheffield; These IDs are used to track the rankings shown in Table 5.

used 20 of the 505 features to represent the data. The 20 features were selected by a sequential feature selection (wrapper) process as the most discriminant features in the set.

4. CONCLUSION

We have shown that outliers in data corresponding to a classification task can be reliably detected by a method that combines estimating the overall probability density and sequential ranking of the data according to observed changes in performance on validation sets. When outliers detected in this way are excluded, generalization ability of a range of classifiers not used in the ranking procedure increases by a greater amount than when the size of the dataset is reduced by the removal of random patterns. We believe what we have demonstrated here will have significant bearing on empirical studies that use benchmark datasets (with no reference to the data collection process) to reach conclusions on the relative merits of different families of classifiers. A stringent criterion has been used because we assume that there isn't the possibility of returning to the data gathering process, hence false identifications have to be minimized and as much of the available data can be used. This criterion can be relaxed in various ways in a different scenario.

Pattern ID	GMM RANKING	SBE RANKING
39	13	1
35	31	4
13	39	10
24	14	78
32	52	15
16	41	21
21	23	60
34	29	36
19	55	35
348	35	87
20	37	94
11	40	63
12	42	98
14	43	75
216	48	57
4	51	54
36	56	90
383	1	-
30	-	2
197	2	-
33	-	3
145	3	-

Table 5. Outliers Selection on the `Ling-Spam` experiments. 17 messages are detected from 540 messages, messages (ID: 11, 12, 13, 14, 16, 19 and 20) are detected as false positive; messages (ID: 32, 34, 35, 36 and 39) are detected as false negative; one message (ID: 4) from true positive and two message (ID: 21, 24) from true negative are wrongly selected by this method. Thus, of the 20 artificially mislabelled examples 12 are correctly picked up by the algorithm. Three of the 20 are incorrectly identified as outliers and two examples from the original dataset are also identified.

5. REFERENCES

[1] L. Tarassenko, A. Nairac, N. Townsend, I. Buxton, and P. Cowley, "Novelty detection for the identification of abnormalities," *International Journal of Systems Science*, vol. 31, no. 11, pp. 1427–1439, 2000.

[2] V. Barnett and T. Lewis, *Outliers in statistical data, 3rd Edition*, John Wiley and Sons, 1994.

[3] A. O'Hagan, "On outlier rejection phenomena in Bayes inference," *J. R. Statist. Soc. B*, vol. 41, no. 3, pp. 358–367, 1979.

[4] C.L. Blake and C.J. Merz, "UCI repository of machine learning databases, http://www.ics.uci.edu/∼mlearn/mlrepository.html, Department of Information and Computer Science, University of California, Irvine," 1998.

[5] I.H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques (Second Edition)*, Morgan Kaufmann, 2005.

[6] D.H. Foley and J.W. Sammon, "An optimal set of discriminant vectors," *IEEE Transaction on Computers*, vol. c-24, no. 3, pp. 281–289, 1975.