Variable Selection using Correlation and SVC Methods: Applications

Amir Reza Saffari Azar Alamdari

Electrical Engineering Department, Sahand University of Technology, Mellat Blvd., Tabriz, Iran amir@ymer.org

Correlation and single variable classifier (SVC) methods are very simple algorithms to select a subset of variables in a dimension reduction problem, which utilize some measures to detect relevancy of a single variable to the target classes without considering the predictor properties to be used. In this paper, along with the description of correlation and single variable classifier ranking methods, the application of these algorithms to the NIPS 2003 Feature Selection Challenge problems is also presented. The results show that these methods can be used as one of primary, computational cost efficient, and easy to implement techniques which have good performance especially when variable space is very large. Also, it has been shown that in all cases using an ensemble averaging predictor would result in a better performance, compared to a single stand-alone predictor.

1 Introduction

Variable and feature selection have become one of the most important topics in machine learning field, especially for those applications with very large variable spaces. Examples vary from image processing, internet texts processing to gene expression array analysis, and in all of these cases handling the large amount of datasets is the major problem.

Any method used to select some of variables in a dataset, resulting in a dimension reduction, is called variable selection method which is the main theme of this book. These methods vary from filter methods to more complex wrappers and embedded algorithms. Filter methods are one of the simplest techniques for variable selection problem, and they can be used as an independent or primary dimension reduction tool before applying more complex methods. Most of filter methods utilize a measure of how a single variable could be useful independently from other variables and from the classifier which is to be used. So the main step is to apply this measure to each individual variable and then select those with the highest values as the best

variables, assuming that this measure provides higher values for better variables. Correlation and single variable classifier (SVC) are two examples of filter algorithms.

In Sect. 2, there is a brief introduction to correlation and single variable classifier methods. Details about the mathematical description and concepts of these methods are not included in this section and unfamiliar readers can refer to Chap. 3 in this book for more details. Sect 3 is an introduction to ensemble averaging methods used as the main predictors in this work, and in Sect. 4, the results and comparisons of applied methods on 5 different datasets of NIPS 2003 Feature Selection Challenge are shown. There is also a conclusion section discussing the results.

2 Introduction to Correlation and SVC Methods

Since Chap. 3 covers filter methods in details, this section contains only a short introduction to the correlation and SVC feature ranking algorithms. Consider a classification problem with two classes, λ_1 and λ_2 represented by $+1$ and -1 respectively. Let $X = \{\mathbf{x}_k | \mathbf{x}_k = (x_{k1}, x_{k2}, \ldots, x_{kn})^T \in \mathbf{R}^n, k = 1, 2, \ldots, m\}$ be the set of m input examples and $Y = \{y_k | y_k \in \{+1, -1\}, k = 1, 2, \ldots, m\}$ be the set of corresponding output labels. If $\mathbf{x}^i = (x_{1i}, x_{2i}, \ldots, x_{mi})^T$ denotes the ith variable vector for $i = 1, 2, \ldots, n$ and $\mathbf{y} = (y_1, y_2, \ldots, y_m)^T$ represents the output vector, then the correlation scoring function is given bellow (**?**):

$$C(i) = \frac{(\mathbf{x}^i - \mu_i)^T(\mathbf{y} - \mu_y)}{\|\mathbf{x}^i - \mu_i\| \times \|\mathbf{y} - \mu_y\|} = \frac{\sum_{k=1}^{m}(x_{ki} - \mu_i)(y_k - \mu_y)}{\sqrt{\sum_{k=1}^{m}(x_{ki} - \mu_i)^2 \sum_{k=1}^{m}(y_k - \mu_y)^2}} \quad (1)$$

where μ_i and μ_y are the expectation values for the variable vector $\mathbf{x}^i$ and the output labels vector $\mathbf{y}$, respectively and $\|.\|$ denotes Euclidean norm. It is clear that this function calculates cosine of the angle between the variable and target vector for each variable. In other words, higher absolute value of correlation indicates higher linear correlation between that variable and target.

Single variable classifier (SVC) (**?**) is a measure of how a single variable can predicts output labels without using other variables. In other words, SVC method constructs a predictor using only the given variable and then measures its correct prediction rate (the number of correct predictions over the total number of examples) on the set of given examples as the corresponding SVC value. The crossfold validation technique can be used to estimate the prediction rate, if there is no validation set. Because this method needs a predictor and a validation algorithm, there exists no explicit equation indicating the SVC values.

There is a very simple way to calculate the SVC quantities. This method is used for all experiments in the application section. First of all, for each variable i, class dependent variable set is constructed: $X^{i,1} = \{x_{ki} | y_k = 1\}$

and $X^{i,-1} = \{x_{ki} | y_k = -1\}$. Let μ_i^1 and μ_i^{-1} be the expectation values of the $X^{i,1}$ and $X^{i,-1}$ sets, respectively. These values are the concetration point of each class on the ith variable axis. The following equation provides a simple predictor based on only ith variable:

$$y = sign((x_i - \frac{\mu_i^1 + \mu_i^{-1}}{2})(\mu_i^1 - \mu_i^{-1})), \quad x_i \in \mathbf{R} \tag{2}$$

where y is the estimated output label, x_i is the input value from ith variable, and the $sign(x)$ gives the sign of its input as $+1$ for $x \geq 0$ and -1 for $x < 0$. The first term inside the sign function, determines the distance and the direction of the input variable from the threshold point, $\frac{\mu_i^1+\mu_i^{-1}}{2}$, and the second term determines the corresponding output class label due to the direction.

Because there is no training session, the correct prediction rate of this predictor on the training set can be used to determine the SVC value for each of the variables, and there is no need to do crossfold validation operations.

2.1 Characteristics of the Correlation and SVC

There are some characteristics of these methods which should be pointed out before proceeding to the applications. The main advantage of using these methods are their simplicity and hence, computational time efficiency. Other methods, which use search methods in possible subsets of variable space, need much more computation time when compared to filter methods. So, if there is a time or computation constraint, one can use these methods. In addition to the simplicity, these methods can also suggest how much class distributions are nonlinear or subjected to noise. In most cases, those variables with nonlinear correlation to the output labels, result in a low value and this can be used to identify them easily. Very noisy variables also can be thought as a highly nonlinear variable. As a result, the scoring functions described above gives lower values for both of the noisy and noninear variables and it is not possible to distinguish between them using only these methods.

To gain more insight, consider a classification problem with two input variables, shown in Fig. 1. Both variables are drawn from a normal distribution with different mean values set to (0,0) and (0,3) for class 1 and class 2, respectively. The standard deviations for both classes are equal to 1. The plot of dataset is shown in upper right section together with axes interchanged in the lower left to simplify the understanding of images. Also, the histograms of each class distribution are shown in upper left for vertical axis and in the lower right for horizontal axis. The total number of examples is 500 for each class. On each axis, the correlation and SVC values are printed. These values are calculated using the methods described in previous section. As shown in Fig.1, regardless of the class labels, first variable is a pure noisy one. The correlation value for noisy variable is very low and the SVC value is about

0.5, indicating that the prediction using this variable is the same as randomly choosing target labels. The second variable is a linearly correlated variable with the target labels, resulting in high values.

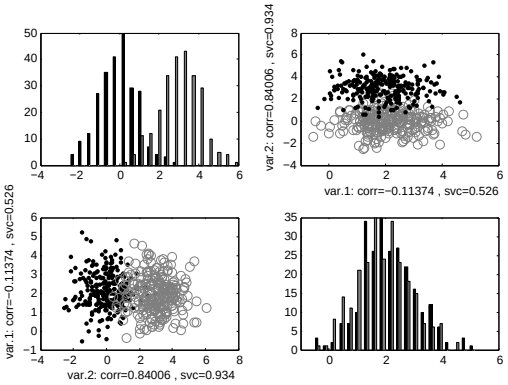

Fig. 1. A simple two variable classification problem: var.1 is a pure noise variable, var.2 is a linearly correlated one.

For a nonlinear problem, consider Fig. 2 which is the famous XOR classification problem. This time each variable has no prediction power when used individually, but can classify the classes when used with other one. As shown in Fig. 2, class distribution on each axis is the same, similar to the situation in noisy variables, and both correlation and SVC values are very low.

Summarizing the examples, correlation and SVC methods can distinguish clearly between a highly noisy variable and one with linear correlation to target values, and they can be used to filter out highly noisy variables. But in nonlinear problems these methods are less applicable and would conflict between noisy and nonlinear variables. Another disadvantage of these methods is the lack of redundancy check in the selected variable subset. In other words, if there were some correlated or similar variables, which carry the same information, these methods would select all of them. Because there is no check to exclude the similar variables.

3 Ensemble Averaging

Ensemble averaging is a simple method to obtain a powerful predictor using a committee of weaker predictors (**?**). The general configuration is shown in Fig. 3 which illustrates some different experts or predictors sharing the same input, in which the individual outputs are combined to produce an overall output. The main hypothesis is that a combination of differently trained predictors can help to improve the prediction performance with increasing the accuracy and confidence of any decision. This is useful especially when the performance of any individual predictor is not satisfactory whether because of variable space complexity, overfitting, or insufficient number of training examples comparing to the input space dimensionality.

There are several ways to combine outputs of individual predictors. First is to vote over different decisions of experts about a given input. This is called the voting system: each expert provides its final decision as a class label, and then the class label with the higher number of votes is selected as final output of the system.

If the output of each predictor before applying decision is named as decision confidence , then another way to combine outputs is to compute average confidence of predictors for a given input, and then select the class with a higher confidence value as final decision. This scheme is a bit different from the previous system, because in voting each predictor shares the same right to select a class label, but in confidence averaging those with less confidence values have lower effect on the final decision than those with higher confidence values.

For example, consider a classification problem that both of its class examples are drawn from a normal distribution with mean values of (0,0) for class 1 and (1,1) for class 2 with standard deviations equal to 1, as shown in

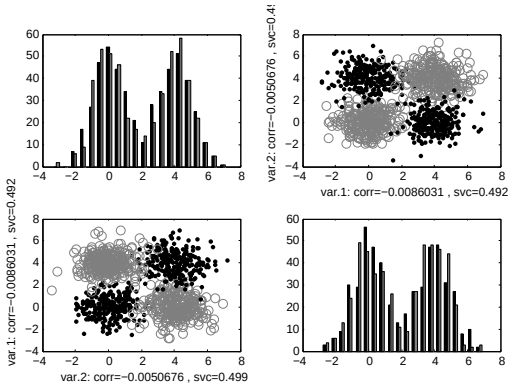

Fig. 2. Nonlinear XOR problem: both variables have very low values.

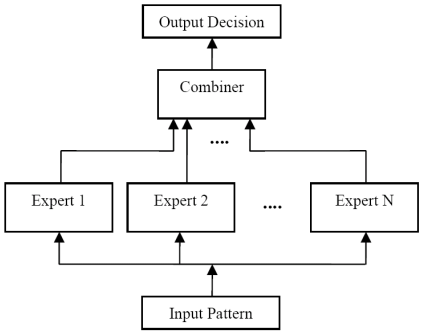

Fig. 3. General structure of an ensemble averaging predictor using N experts.

Fig. 4. The individual prediction error of 9 MLP neural networks with different initial weights is shown in Table. 1. Here values are prediction errors on 20000 unseen test examples. All networks have 2 tangent sigmoid neurons in hidden layer and are trained using scaled conjugate gradient (SCG) algorithm on 4000 examples.

The average of prediction error of each network is about 0.4795 which is a bit better than a random guess, and this is due to the high overlap and conflict of class distributions. Using a voting method, the overall prediction error turns to be 0.2395 which shows a 0.2400 improvement. This is why in most cases ensemble averaging can convert a group of weak predictors to a stronger one easily. Even using 2 MLPs in the committee would result in a 0.2303 improvement. Additional MLPs is presented here to show that the bad performance of each MLP is not due to the learning processes. The effect of more committee memebers on the overall improvement is not so much here, but might be important in more difficult problems. Note that the second row of the Table. 1 shows the ensemble prediction error due to addition of each MLP to the committee.

Since the original distribution is normal, the Bayesian optimum estimation of the class labels can be carried out easily. For each test example, its distance from the mean points of each class can be used to predict the output label. Using this method, the test prediction error is 0.2396. Again this shows that ensemble averaging method can improve the prediction performance of a set of weak learners to a near Bayesian optimum predictor. The cost of this process is just training more weak predictors, which in most of cases is not so much high (according to computation time).

Table 1. Prediction error of individual neural networks, the first row, and the prediction error of the committee according to the number of members in the ensemble, the second row.

Network No.	1	2	3	4	5	6	7	8	9
	0.4799	0.4799	0.4784	0.4806	0.4796	0.4783	0.4805	0.4790	0.4794
Member Num.	1	2	3	4	5	6	7	8	9
	0.4799	0.2492	0.2392	0.2425	0.2401	0.2424	0.2392	0.2403	0.2395

For more information about ensemble methods and other committee machines, refer to Chaps. 1 and 7 in this book and also (**????**). Note that this section is not related explicitly to the variable selection issues.

4 Applications to NIPS 2003 Feature Selection Challenge

This section contains applications of discussed methods to the NIPS 2003 Feature Selection Challenge. The main goal in this challenge was to reduce the

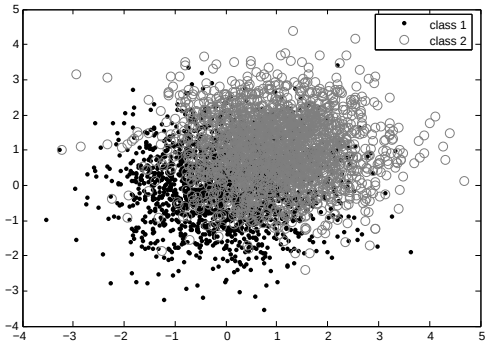

Fig. 4. Dataset used to train neural networks in ensemble averaging example.

variable space as much as possible while improving the performance of predictors as higher as possible. There were five different datasets with different size of variable spaces ranging from 500 to 100,000. The number of training examples was also different and in some cases was very low with respect to the space dimensionality. In addition, some pure noisy variables were included in the datasets as random probes to measure the quality of variable selection methods.

The results of the correlation and SVC analysis for each dataset are shown in Fig. 5. Values are sorted in descending manner according to the correlation values. Since the descending order of variables for the correlation and SVC values are not the same, there are some irregularities in the SVC plots. Note that the logarithmic scale is used for the horizontal axis for more clarity on first parts of the plot.

Before proceeding to the applications sections, it is useful to explain the common overall procedures applied to the challenge datasets in this work. The dataset specific information will be given in next subsections. There are three different but not independent processes to solve the problem of each dataset: variable selection, preprocessing, and classification. The followings are the summarized steps for these three basic tasks:

1. First of all, constant variables, which their values do not change over the training set, are detected and removed from the dataset.
2. The variables are normalized to have zero mean values and also to fit in the $[-1, 1]$ range, except the Dorothea (see Dorothea subsection).
3. For each dataset, using a k-fold cross-validation (k depends on the dataset), a MLP neural network with one hidden layer is trained to estimate the number of neurons in the hidden layer.
4. The correlation and SVC values are calculated and sorted for each variable in the dataset, as shown in Fig. 5.
5. The first estimation for the number of good variables in each dataset is computed using a simple crossfold validation method for the MLP predictor in step 2. Since an online validation test was provided through the challenge website, these numbers were optimized in next steps to be consistent with the actual preprocessings and also predictors.
6. 25 MLP networks with different randomly chosen initial weights are trained on the selected subset using SCG algorithm. The transfer function of each neuron is selected to be tangent sigmoid for all predictors. The number of neurons in the hidden layer is selected on the basis of the experimental results of the variable selection step, but are tuned manually according to the online validation tests.
7. After the training, those networks with acceptable training error performances are selected as committee members (because in some cases the networks are stuck to the local minima during the training sessions). This selection procedure is carried out by filtering out low performance networks using a threshold on the training error.

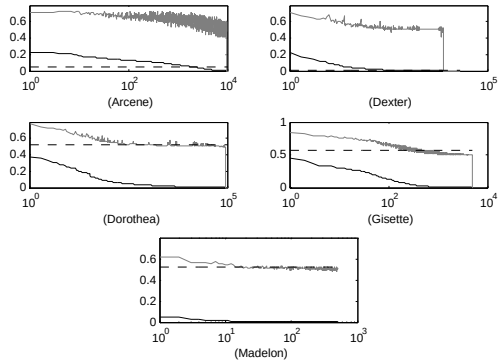

Fig. 5. Correlation and SVC values plot for 5 challenge datasets. Correlation values are plotted with black lines while SVC values are in grey. The dashed horizontal line indicates threshold.

8. For validation/test class prediction, the output values of the committee networks are averaged to give the overall confidence about the class labels. The sign of this confidence value gives the final predicted class label.
9. The necessity of a linear PCA (**?**) preprocessing method usage is also determined for each dataset by applying the PCA to the selected subset of variables and then comparing the validation classification results to the non-preprocessing system.
10. These procedures are applied for both correlation and SVC ranking methods in each dataset, and then one with higher validation performance (lower classification error) and also lower number of variables is selected as the basic algorithm for the variable selection in that dataset.
11. Using online validation utility, the number of variables and also the number of neurons in the hidden layer of MLPs are tuned manually to give the best result.

Next subsections have the detailed information about the application of the described methods on each dataset specificly. More information about this competition, the results, and the descriptions of the datasets can be found in the following website:
`http://clopinet.com/isabelle/Projects/NIPS2003/#challenge`.

4.1 Arcene

This is the first dataset with a high number of variables (10000) and relatively low number of examples (100). The correlation values are sorted and those with higher values than 0.05 are selected which is about 20% of overall variables, see Fig. 5.a.

The correlation analysis shows that in comparison to other datasets discussed below, the numbers of variables with relatively good correlation values are high in Arcene. As a result, it seems that this dataset consists of many linearly correlated parts with less contributed noise. The fraction of random probes included in the selected variables is 2.92% which again shows that correlation analysis is good for noisy variables detection and removal.

A linear PCA is applied to the selected subset and the components with low contribution to overall variance are removed. Then 25 MLP networks with 5 hidden neurons are trained on the resulting dataset, as discussed in previous section. It is useful to note that because of very low number of examples, all networks are subject to overfitting. Average prediction error for single networks on unseen validation set is 0.2199. Using a committee prediction error turns to be 0.1437 which shows a 0.0762 improvement. This result is expected for the cases with low number of examples and hence low generalization. The prediction error of ensemble on unseen test examples is 0.1924.

4.2 Dexter

The second dataset is Dexter with again unbalanced number of variables (20000) and examples (300). The correlation values are sorted and those with higher values than 0.0065 are selected which is about 5% of overall variables, see Fig. 5.b.

Note that there are many variables with fixed values in this and others datasets. Since using these variables gains no power in prediction algorithm, they can be easily filtered out. These variables consist about 60% of overall variables in this dataset. There are also many variables with low correlation values. This indicates a highly nonlinear or a noisy problem compared to the previous dataset. Another fact that suggests this issue, is seen from the number of selected variables (5%) with very low threshold value of 0.0065 which is very close to the correlation values of pure noisy variables. As a result, the fraction of random probes included in the selected variables is 36.86% which is very high.

There is no preprocessing for this dataset, except the normalization applied in first steps. 25 MLP networks with 2 hidden neurons are trained on the resulting dataset. Prediction error average for single networks on unseen validation set is 0.0821, where using a committee improves prediction error to 0.0700. The prediction error of ensemble on unseen test examples is 0.0495.

4.3 Dorothea

Dorothea is the third dataset which its variable are all binary values with very high dimensional input space (100000) and relatively low number of examples (800). Also this dataset is highly unbalanced according to the number of positive and negative examples, where positive examples consist only 10% of overall examples. The SVC values are sorted and those with higher values than 0.52 are selected which they consist about 1.25% of variables, see Fig. 5.c.

Fig.5.c together with the number of selected variables (1.25%) with low threshold value of 0.52 for SVC shows that this problem has again many nonlinear or noisy parts. The fraction of random probes included in the selected variables is 13.22%, indicating that lowering the threshold value results in a higher number of noise variables to be included in the selected set.

In preprocessing step, every binary value of zero in dataset is converted to -1. 25 MLP networks with 2 hidden neurons are trained on the resulting dataset. Since the number of negative examples is much higher than positive ones, each network tends to predict more negative. The main performance measure of this competition was balanced error rate (BER), which calculates the average of the false detections according to the number of positive and negative examples by:

$$BER = 0.5(\frac{F_p}{N_p} + \frac{F_n}{N_n}) \tag{3}$$

where N_p and N_n are the total number of positive and negative examples, respectively, and F_p and F_n are the number of false detections of the positive and negative examples, respectively. As a result, the risk of an erroneous prediction for both classes is not equal and a risk minimization (**?**) scenario must be used. In this way, decision boundary which is zero for other datasets, is shifted toward -0.7. This results in the prediction of negative label if the confidence were higher than -0.7. So, only the examples which predictor is more confident about them are detected as negative. The -0.7 bias value is calculated first with a crossfold validation method and then optimized with online validation tests manually.

The prediction error average for single networks on unseen validation set is 0.1643. The committee has prediction error of 0.1020 and shows a 0.0623 improvement, which is again expected because of low number of examples, especially positive ones. The prediction error of ensemble on unseen test set is 0.1393.

4.4 Gisette

The fourth dataset is Gisette with a balanced number of variables (5000) and examples (6000). The SVC values are sorted and those with higher values than 0.56 are selected which is about 10% of the overall variables, see Fig. 5.d.

SVC analysis shows that this example is not much nonlinear or subjected to noise, because the number of variables with good values is high. The fraction of random probes included in the selected variables is zero, indicating very good performance in noisy variables removal.

A linear PCA is applied and the components with low contribution to overall variance are removed. Then 25 MLP networks with 3 hidden neurons are trained on the resulting dataset. Because of relatively high number of examples according to difficulty of problem, it is expected that the performance of a committee and individual members would be close. Prediction error average for single networks on unseen validation set is 0.0309. Using a committee, prediction error only improves with 0.0019 and becomes 0.0290. The prediction error of ensemble on unseen test set is 0.0258.

4.5 Madelon

The last dataset is Madelon with (2000) number of examples and (500) variables. The SVC values are sorted and those with higher values than 0.55 are selected which is about 2% of variables, see Fig. 5.e. This dataset is a highly nonlinear classification problem as seen from SVC values. The fraction of random probes included in the selected variables is zero. Since this dataset is a high dimensional XOR problem, it is a matter of chance to get none of the random probes in the selected subset and this is not an indication of the powers of this method.

Table 2. NIPS 2003 challenge results for Collection2.

Dec. 1^{st}	Our best challenge entry					The winning challenge entry					
Dataset	Score	BER	AUC	Feat	Probe	Score	BER	AUC	Feat	Probe	Test
Overall	28.00	10.03	89.97	7.71	10.60	88.00	6.84	97.22	80.3	47.8	1
Arcene	25.45	19.24	80.76	20.18	2.92	98.18	13.30	93.48	100.0	30.0	1
Dexter	63.64	4.95	95.05	5.01	36.86	96.36	3.90	99.01	1.5	12.9	1
Dorothea	32.73	13.93	86.07	1.25	13.22	98.18	8.54	95.92	100.0	50.0	1
Gisette	-23.64	2.58	97.42	10.10	0	98.18	1.37	98.63	18.3	0.0	1
Madelon	41.82	9.44	90.56	2	0	100.00	7.17	96.95	1.6	0.0	1

There is no preprocessing for this dataset, except the primary normalization. 25 MLP networks with 25 hidden neurons are trained on resulting dataset. The number of neurons in hidden layer is more than other cases because of nonlinearity of class distributions. Prediction error average for single networks on unseen validation set is 0.1309 and combining them into a committee, prediction error improves with 0.0292 and reaches 0.1017. The prediction error of ensemble on unseen test set is 0.0944.

5 Conclusion

In this paper, the correlation and SVC based variable selection was introduced and applied to NIPS 2003 Feature Selection Challenge. There was also a brief introduction to ensemble averaging methods and it was shown that how a committee of weak predictors could be converted to a stronger one.

The overall performance of applied methods to 5 different datasets of challenge is shown in Table. 2 together with the best winning entry of the challenge. Table. 3 shows the improvements obtained by using a committee instead of a single MLP network for the validation sets of the challenge datasets.

Table 3. Improvements obtained by using a committee instead of a single MLP network on the validation set.

Overall	Arcene	Dexter	Dorothea	Gisette	Madelon
3.63	7.62	1.21	6.23	0.19	2.29

Summarizing the results, the correlation and SVC are very simple, easy to implement, and computational time efficient algorithms which have relatively good performance compared to other complex methods. These methods are very useful when the variable space dimension is large and other methods using exhaustive search in subset of possible variables need much more computations. On a Pentium IV, 2.4GHz PC with 512MB RAM running Microsoft Windows 2000 Professional, all computations for variable selection

using MATLAB 6.5 finished in less than 15 minutes for both correlation and SVC values of all 5 datasets. This is quite great performance if one considers very large challenge datasets.

A simple comparision between the correlation and SVC ranking methods is given in Fig. 6. Let S^N_{COR} and S^N_{SVC} be the subsets of the original dataset with N selected variables according to their rankings using the correlation and SVC, respectively. In this case the vertical axis of Fig.6 shows the fraction of the total number of common elements in these two sets per set sizes, i.e. $N_c = \frac{F(S^N_{COR} \cap S^N_{SVC})}{N}$, where $F(.)$ returns the number of elements of the input set argument. In other words, this figure shows the similarity in the selected variable subsets according to the correlation and SVC methods.

Table 4. The average rate of the common variables using correlation and SVC for the challenge datasets, first row. Second row presents this rate for the number of selected variables in the application section.

	Overall	Arcene	Dexter	Dorothea	Gisette	Madelon
Average	0.7950	0.8013	0.7796	0.7959	0.8712	0.7269
Application	0.7698	0.7661	0.7193	0.6042	0.8594	0.9000

As it is obvious from this figure, the correlation and SVC shares most of the best variables (first parts of the plots) in all of the datasets, except the Arcene. In other words, linear correlation might result in a good SVC score and vice versa. Table. 4 shows the average of these plots in first row, together with the rate of the common variable in the selected subset of variables for the application section discussed earlier. Another interesting issue is the relation between the rate of the random probes and the rate of the common variables in the subsets of datasets used in the applications. For Gisette and Madelon the total number of selected random probes was zero. Table. 4 shows that the common variable rate for these two datasets are also higher, comparing to other datasets. The Dexter and Dorothea had the worst performance in filtering out the random probes, and the rate of common variables for these two sets are also lower than others. In other words, as the filtering system starts to select the random probes, the difference between the correlation and SVC grows higher. Note that these results and analysis are only experimental and have no theoritical basis and the relation between these ranking and filtering methods might be a case of future study.

It is obvious that these simple ranking methods are not the best ones to choose a subset of variables, especially in nonlinear classification problems which one have to consider a couple of variables together to understand underlying distribution. But it is useful to note that these methods can be used to guess nonlinearity degree of the problem and on the other hand filter out very noisy variables. As a result, these can be used as a primary analysis and

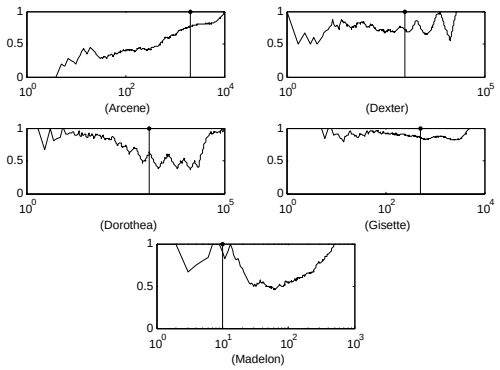

Fig. 6. The similarity plots of the variable selection using correlation and SVC methods on challenge datasets. Note that the vertical solid line indicates the number of selected variables for each dataset in the competition.

selection tools in very large variable spaces, comparing to methods and results obtained by other challenge participants.

Another point is the benefits of using a simple ensemble averaging method over single predictors, especially in situations where generalization is not satisfactory, due to the complexity of the problem, or low number of training examples. Results show a 3.63% improvement in overall performance using an ensemble averaging scenario over single predictors. Training 25 neural networks for each dataset take 5-30 minutes on average depending on the size of the dataset. This is fast enough to be implemented in order to improve prediction performance especially when the numbers of training examples are low.

The overall results can be found in challenge results website under Collection2 method name. Also, you can visit the following link for some MATLAB programs used by author and additional information for this challenge: `http://www.ymer.org/research/variable.htm`.