A Comparison of Classification Accuracy Achieved with Wrappers, Filters and PCA

Andreas G. K. Janecek and Wilfried N. Gansterer

University of Vienna, Research Lab Computational Technologies and Applications

Abstract. Dimensionality reduction and feature subset selection are two techniques for reducing the attribute space of a feature set, which is an important component of both supervised and unsupervised classification or regression problems. While in feature subset selection a subset of the original attributes is extracted, dimensionality reduction produces linear combinations of the original attribute set.

In this paper we investigate the relationship between attribute reduction techniques and the resulting classification accuracy for two very different application ares: On the one hand, we consider e-mail filtering, where various properties of e-mail messages are extracted, and on the other hand, we consider drug discovery problems, where quantitative representations of molecular structures are encoded in terms of information-preserving descriptor values.

In the present work, subsets of the original attributes constructed by filter and wrapper techniques as well as subsets of linear combinations of the original attributes constructed by three different variants of the principle component analysis (PCA) are compared in terms of the classification performance achieved with various machine learning algorithms. We successively reduce the size of the attribute sets and investigate the changes in the classification results. Moreover, we explore the relationship between the variance captured in the linear combinations within PCA and the classification accuracy.

First results show that the classification accuracy based on PCA are highly sensitive to the type of data and that the variance captured the principal components is not necessarily a vital indicator for the classification performance.

1 Introduction and Related Work

As the dimensionality of the data increases, many types of data analysis and classification become significantly harder, and sometimes the data becomes increasingly sparse in the space it occupies. This can lead to big problems for both supervised and unsupervised learning. In the literature, this phenomenon is referred to as the *curse of dimensionality* [20]. On the one hand, in the case of supervised learning or classification there might be too few data objects to allow the creation of a reliable model that assigns a class to all possible objects. On the other hand, for unsupervised learning methods or clustering algorithms various vitally important definitions (e.g., density or distance between points) may become less convincing. As a result, a high number of features can lead to lower classification accuracy and clusters of poor quality. High dimensional data is also a serious problem for many classification algorithms due to its huge requirements for computational cost and memory usage. Moreover, when dealing with a huge number of instances and attributes storage requirements may also become a problem. Besides this key factor, Tan et al. [22] also mention that a reduction of the attribute space leads to a better understandable model and simplifies the usage of different visualization techniques. Some extensive surveys of various feature selection and dimensionality reduction approaches can be found in the literature, for example, in Molina et al. [16] or Guyon et al [10].

Principle component analysis (PCA) is a well known data preprocessing technique to capture linear dependencies among attributes of a data set. It compresses the attribute space by identifying the strongest patterns in the data, i.e., the attribute space is reduced with loosing only the smallest possible amount of information about the original data.

Howley et al. [11] have investigated the effect of PCA on machine learning accuracy with high dimensional spectral data based on different pre-processing steps. They use the NIPALS [8] method to iteratively compute only the first n PCs of a data sample until the required number of PCs have been generated. Their results show that using this PCA method in combination with classification improves the classification accuracy when dealing with high dimensional data. Popelinsky [19] has analyzed the effect of PCA on three different machine learning methods. In one test-run, the PC scores (i.e., linear combinations of the original attributes) were added to the original attributes, in the second test-run, the PC scores replaced them. The results show that adding the PC scores increased the classification results for all machine learning algorithms, whereas replacing the original attributes only increased the accuracy for one algorithm.

Attributes resulting from various PCA variants may differ significantly in their coverage of the variability of the original attributes. To the best of our knowledge no systematic studies have been carried out to explore the relationship between the variability captured in the linear combinations of PCA and the accuracy of machine learning algorithms. One of the objectives of this paper is to summarize initial investigations of this issue. More generally, we investigate the variation of the classification accuracy depending on the choice of the feature set (that includes the choice of specific variants of for calculating the PCA) for two quite different data sets. Another important aspect motivating such investigations are questions relating to how classification accuracy based on PCA subsets compares to classification accuracy based on subsets of the original features of the same size, or how to identify the smallest subset of original features which yields a classification accuracy comparable to the one of a given PCA subset.

2 Feature Subset Selection and Dimensionality Reduction

The main idea of feature subset selection (FS) is to remove redundant or irrelevant features from the data set as they can lead to a reduction of the classification accuracy or clustering quality and to an unnecessary increase of computational cost [2], [14]. The advantage of FS is that no information about the importance of single features is lost. On the other hand, if a small set of features is required and the original features are diverse, information might be lost as some of the features must be omitted. With dimensionality reduction (DR) the size of the attribute space can often be strikingly decreased without loosing a lot of information of the original attribute space. The not neglectable disadvantage of DR is the fact that the linear combinations are not interpretable and the information about how much an attribute contributes to the linear combinations is often lost.

2.1 Feature (Subset) Selection

Generally speaking, there are three types of feature subset selection approaches, filters, wrappers, as well as embedded approaches which perform the feature selection process as an integral part of the machine learning (ML) algorithm.

Wrappers are feedback methods that incorporate the ML algorithm in the FS process, i.e., they rely on the performance of a specific classifier (which is used as a black box) to evaluate the quality of a set of features [13].

Filters are classifier agnostic pre-selection methods that are independent of the later applied machine learning algorithm. Beside some statistical filtering methods like Fisher score or Pearson correlation, *information gain*, originally used to compute splitting criteria for decision trees, is often applied to find out how well each single feature separates the given data set. The overall entropy I of a given dataset S is given as:

$$I(S) = -\sum_{i=1}^{C} p_i \log_2 p_i, \tag{1}$$

where C denotes the total number of classes and p_i the portion of instances that belong to class i. The reduction in entropy or the *gain in information* is computed for each attribute A according to

$$IG(S, A) = I(S) - \sum_{v \epsilon A} \frac{|S_{A,v}|}{|S|} I(S_{A,v}), \tag{2}$$

where v is a value of A and $S_{A,v}$ is the set of instances where A has value v.

2.2 Dimensionality Reduction

DR refers to algorithms and techniques that create new attributes which are combinations of the old attributes to reduce the dimensionality of data sets [15]. The most important DR technique is principal component analysis (PCA). PCA produces new attributes which are linear combinations of the original variables, whereas the goal of a factor analysis [9] is to express the original attributes as linear combinations of a small number of hidden or latent attributes.

PCA. The goal of PCA [12], [17] is to find a set of new attributes (these new attributes are called principal components, PCs) that meets the following criteria: The PCs are (i) linear combinations of the original attributes, (ii) orthogonal to each other, and (iii) capture the maximum amount of variation in the data. Often the variability of the data can be captured by a relatively small number of PCs, and, as a result, PCA can result in relatively low-dimensional data with usually lower noise than the original patterns. PCA depends on the scaling of the data, and therefore the results are sometimes non-conclusively, and, in addition to that, the principal components are not always easy to interpret.

Mathematical Background. The *covariance* of two attributes is a measure how strongly the attributes vary together. The covariance of two random variables x and y of a sample with size n and mean $\overline{x}$, $\overline{y}$ can be calculated as

$$Cov(x,y) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})(y - \overline{y}) \tag{3}$$

When x and y are normalized by their standard deviations σ_x and σ_y, then the covariance of x and y is equal to the correlation coefficient of x and y, which indicates the strength *and* direction of a linear relationship between x and y.

$$Corr(x,y) = \frac{Cov(x,y)}{\sigma_x \sigma_y} \tag{4}$$

Given an m by n matrix D, whose m rows are data objects and whose n columns are attributes, we can calculate the covariance matrix $Cov(D)$ which is constructed of the single covariances. If we shift the values of each attribute of D such that the mean of each attribute is 0, then $Cov(D) = D^T D$.

Tan et al. [22] summarizes four main properties of the principle component analysis: (*i*) Each pair of attributes has 0 covariance, (*ii*) the attributes are ordered descendingly with respect of their variance, (*iii*) the first attribute captures as much of the variance of the data as possible, and, (*iv*) each successive attribute captures as much of the remaining data as possible.

One way to obtain a transformation of the data that has these properties is based on the eigenvalue analysis of the covariance matrix. Let $\lambda_1, ..., \lambda_n$ be the non-negative descending ordered eigenvalues and $U = [\mathbf{u}_1, ..., \mathbf{u}_n]$ the matrix of eigenvectors of $Cov(D)$ (the i^{th} eigenvector corresponds to the the i^{th} largest eigenvalue). The matrix $X = DU$ is the transformed data that satisfies the conditions mentioned above, where each attribute is a linear combination of the original attributes, the variance of the i^{th} new attribute is λ_i, and the sum of the variance of all new attributes is equal to the sum of the variance of the original attributes. The eigenvectors of $Cov(D)$ define a new set of orthogonal axes that can be viewed as a rotation of the original axes. The total variability of the data is still preserved, but the new attributes are now uncorrelated.

3 Experimental Evaluation

For the experimental evaluation we used MATLAB to compute three different variants of PCA (cf. Section 3.2), and the WEKA toolkit [23] to measure the classification performance of the learning methods on each feature set.

3.1 Data Sets

The data sets used for the experiments come from two completely different application areas and differ strongly in the number of instances and features as well as in the character of the features.

E-mail data. The first data set consists of 10 000 e-mail messages (half spam, half not spam) taken from the TREC 2005 e-mail corpus [6]. The values of the features for each message were extracted using the state-of-the-art spam filtering system SpamAssassin [1] (SA), where different parts of each e-mail message are checked by various tests, each test is assigned a certain value (positive for spam messages, negative for non-spam messages). Although the number of the features within SA is rather large, only a relatively small number of these features provide useful information. For the data set used only 230 out of 800 tests triggered at least once. Hence, our e-mail data set is a $10\,000 \times 230$ matrix.

Drug discovery data. The second data set has a medicinal chemistry background. The goal is to identify potential safety risks in an early phase of the drug discovery process, in order to avoid costly and elaborate late stage failures in clinical studies. This data set consists of 249 structurally diverse compounds, 110 of them are known to be substrates of P-glycoprotein, a macromolecule that is notorious for its potential to decrease the efficacy of drugs ("antitarget"). The remaining 139 compounds are non-substrates. The chemical structures of these candidates are encoded in terms of 366 information preserving descriptor values (features). Hence, our drug discovery data set is a 249×366 matrix.

Diversity. While the e-mail data set is very sparse (97.5% of all entries are zero) the drug discovery data set contains only about 18% zero entries. Moreover, most of the e-mail features have the property, that they are either zero or have a fixed value (i.e., when a test triggers the same number is assigned for each message). This is completely different from the drug discovery data set where the attribute values and their variances are very diverse.

3.2 Feature Sets

In our experiments we compare two FS methods and three different DR methods based on PCA (cf. Section 2.1).

- Wrapper approach: For extracting the wrapper subsets we randomly select attribute subsets and use cross validation to estimate the accuracy of the learning scheme for these subsets. A paired t-test then computes the probability if other subsets may perform substantially better. The result is a *fixed* set of features, between 4 and 18 features for our evaluations.
- Filter approach: As a second FS method we ranked the attributes with respect to their information gain. This ranking is independent of a specific learning algorithm and contains – before selecting a subset – all attributes.

In the literature, several variants of PCA appear. Here, we investigate the differences of three variants in terms of resulting classification accuracy. For all subsets based on PCA we first performed a mean shift of all features such that the mean becomes 0. We denote the resulting feature-instance matrix as M. Based on this first preprocessing step we define three variants of the PCA computation. The PCA subsets contain $min(features, instances)$ linear combinations of the original attributes (before selecting a subset).

- PCA_1: The eigenvalues and eigenvectors are computed using the covariance matrix of M (cf. Section 2.2). The new attribute values are then computed by multiplying M with the eigenvectors of $Cov(M)$.
- PCA_2: The eigenvalues and eigenvectors are computed using the correlation matrix of M (cf. Section 2.2). The new attribute values are then computed by multiplying M with the eigenvectors of $Corr(M)$.
- PCA_3: Each feature of M is normalized by its standard deviation (i.e., z-scored). This normalized values are used for the computation of eigenvalues and eigenvectors (i.e., there is no difference between the covariance and the correlation coefficient) and also for the computation of the new attributes.

3.3 Machine Learning Methods

For evaluating the classification performance of the reduced feature sets we use six different machine learning methods. The methods are not explained in detail, but a reference is given for each of the models.

- A support vector machine (SVM) based on the sequential minimal optimization (SMO) [18] algorithm using a polynomial kernel (exponent of 1).
- A k-nearest neighbors (kNN) classifier using different values of k (1 to 9) [7].
- A bagging ensemble learner using a pruned decision tree as base learner [3].
- A single J.48 decision tree based on Quinlans C4.5 decision tree algorithm [21].
- A random forest classifier using a forest of random trees [4].
- A Java implementation (JRip) of a propositional rule learner, called RIPPER (Repeated incremental pruning to produce error reduction [5]).

4 Results

For all feature sets except the wrapper subsets we measured the classification performance for subsets containing the n “best ranked” features (n varies between 100 and 1). For the information gain method, these best ranked features are the top n information gain ranked original features, for the PCA subsets the best ranked features are the top n linear combinations of the original attributes capturing most of the variability of the original attributes, i.e., the linear combinations having the n biggest eigenvalues.

The classification results were measured using a 10-fold cross-validation, i.e. partitioning the data into 10 parts and iteratively using nine parts of the data objects for training and the remaining one for testing. The achieved results are shown separately for the two feature sets. As the kNN results with $k = 1$ achieved the best results amongst all values of k, only the kNN(1) results are shown.

4.1 E-mail Data

Table 1 shows the average classification accuracy for the IG subsets and the PCA subsets over all top n features (from 100 to 1). The best and the worst average results for each feature set are highlighted in bold and italic letters, respectively. The best overall result over all feature sets is marked with an asterisk.

Table 1. E-mail data – average overall classification accuracy (in %).

	SVM	kNN(1)	Bagging	J.48	RandF	JRip	AVG.
Infogain	99.20	99.18	*99.16*	*99.16*	**99.21**	99.18	99.18
PCA_1	*99.26*	99.70	99.68	99.70	* **99.77**	99.67	99.63
PCA_2	*99.55*	99.69	99.67	99.69	**99.75**	99.68	99.67
PCA_3	*99.54*	99.64	99.65	99.64	**99.65**	99.64	99.63

Table 2. E-mail data – best overall classification accuracy (in %).

	SVM	kNN(1)	Bagging	J.48	RandF	JRip
All features	**99.75** ***230 attr.***	99.70 *230 attr.*	99.71 *230 attr.*	*99.65* *230 attr.*	99.73 *230 attr.*	99.66 *230 attr.*
Wrapper *fixed set*	99.61 *7 attr.*	99.60 *5 attr.*	99.61 *4 attr.*	99.61 *7 attr.*	**99.67** ***11 attr.***	99.64 *7 attr.*
Infogain	99.76 *100 attr.*	99.71 *50 attr.*	*99.70* *50 attr.*	99.71 *100 attr.*	**99.78** ***80 attr.***	99.72 *90 attr.*
PCA_1	*99.65* *90 PCs*	99.74 *40 PCs*	99.69 *40 PCs*	99.75 *30 PCs*	* **99.82** ***70 PCs***	99.73 *5 PCs*
PCA_2	*99.67* *90 PCs*	99.75 *40 PCs*	99.72 *30 PCs*	99.78 *60 PCs*	**99.80** ***15 PCs***	99.76 *40 PCs*
PCA_3	*99.65* *100 PCs*	99.73 *5 PCs*	99.71 *20 PCs*	99.71 *4 PCs*	**99.79** ***15 PCs***	99.73 *50 PCs*

Table 2 shows the best classification results for all feature subsets (including the wrapper subsets) and for a classification using the complete feature set. Table 2 also contains the information how many original features (for FS) and how many linear combinations (for PCA) were needed to achieve these results.

Algorithms. Although the overall classification accuracy is very good in general, when comparing the different machine learning methods is can be seen that random forest achieves the best results for all of the reduced feature sets used, and that SVM (surprisingly!) often archives the lowest results. Especially for the average *PCA* results over all subsets (Table 1) SVM shows the lowest classification accuracy. When the complete feature set is used, the SVM results are slightly better than the random forest results (Table 2).

Feature Subset Selection. A comparison of the two FS methods shows that the best results achieved with IG subsets are better than the wrapper results (see Table 2). Nevertheless, when looking at the size of the subsets that could achieve the best results it can be seen that the wrapper subsets are very small. Figure 1 shows the degradation in the classification accuracy when the number of features in the IG subsets is reduced. Interestingly, all machine learning methods show the same curve without any significant differences. The results are very stable until the subsets are reduced to 30 or less features, then the overall classification accuracy tends to decrease proportional to the reduction of features. Comparing the wrapper results with the IG results using the same subset size (Figure 1 (6 to 8 features), and Table 2), it can be seen that wrapper clearly outperforms IG.

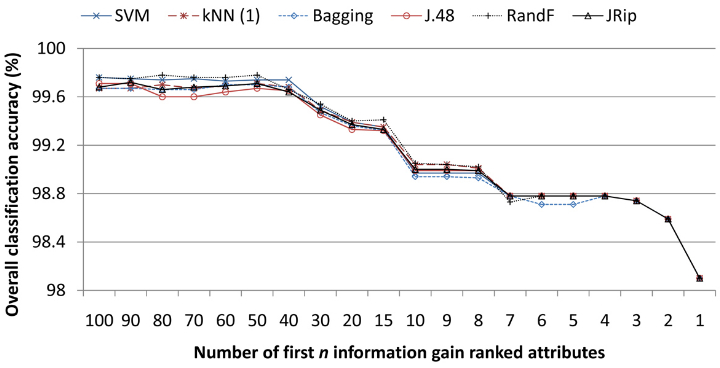

Fig. 1. E-mail data – information gain subsets

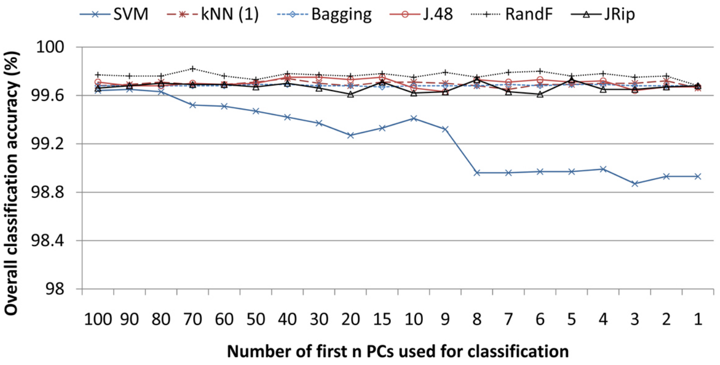

Fig. 2. E-mail data – PCA_1 subsets

PCA. Figure 2 shows the overall classification accuracy for PCA_1 (cf. Section 3.2). The classification performance is very stable regardless of the number of linear combinations used. Only the SVM method clearly decreases with a smaller number of PCs. The classification accuracy for the other two PCA variants PCA_2 and PCA_3 is very similar (see Tables 1 and 2).

Table 3. E-mail data – amount of variance captured by first n PCs (max. dim: 230)

PCs	100	80	60	40	20	10	8	6	5	4	3	2	1
Cov.	99.4%	98.9%	97.8%	95.3%	87.0%	75.7%	71.7%	66.0%	62.7%	58.9%	54.0%	48.0%	38.0%
Corr.	67.7%	58.9%	49.4%	38.3%	24.7%	15.8%	13.7%	11.5%	10.3%	9.0%	7.7%	6.3%	3.9%

Explaining the variance. Even though the classification results for all three PCA variants are similar, it is very interesting to notice that when the correlation matrix is used to compute the eigenvectors and eigenvalues (as it is the case for PCA_2 and PCA_3) – instead of the covariance matrix – the amount of variance that can be captured by the first n PCs (i.e., the cumulated percentage of the first n eigenvalues) decreases remarkably.

4.2 Drug Discovery Data

Tables 4 and 5 show the same information than Tables 1 and 2, the average and the best classification results, this time for the drug discover data set.

Table 4. Drug discovery data – average overall classification accuracy (in %).

	SVM	kNN(1)	Bagging	J.48	RandF	JRip	AVG.
Infogain	69.25	*67.55*	**70.63**	68.47	68.04	70.32	69.04
PCA_1	63.50	**65.87**	65.84	*61.81*	65.03	65.74	64.63
PCA_2	*61.05*	66.39	**69.27**	65.27	67.16	65.92	65.84
PCA_3	68.78	67.28	* **71.02**	*63.76*	69.66	67.06	67.93

Table 5. Drug discovery data – best overall classification accuracy (in %).

	SVM	kNN(1)	Bagging	J.48	RandF	JRip
All features	70.67 *367 attr.*	73.89 *367 attr.*	**74.30** ***367 attr.***	*64.24* *367 attr.*	73.52 *367 attr.*	69.09 *367 attr.*
Wrapper *fixed set*	77.48 *18 attr.*	79.91 *6 attr.*	79.51 *10 attr.*	79.53 *6 attr.*	* **79.93** ***6 attr.***	79.89 *6 attr.*
Infogain	72.70 *60 attr.*	73.08 *80 attr.*	**74.31** ***20 attr.***	*71.11* *1 attr.*	71.89 *7 attr.*	73.52 *2 attr.*
PCA_1	70.69 *60 PCs*	**73.07** ***15 PCs***	68.68 *15 PCs*	*65.87* *15 PCs*	69.48 *4 PCs*	69.09 *6 PCs*
PCA_2	*65.89* *60 PCs*	71.89 *60 PCs*	73.08 *15 PCs*	68.29 *60 PCs*	**73.89** ***40 PCs***	68.69 *4 PCs*
PCA_3	73.89 *6 PCs*	73.09 *10 PCs*	75.90 *6 PCs*	*69.09* *5 PCs*	**76.69** ***10 PCs***	71.48 *7 PCs*

Algorithms. Compared to the e-mail data set, the overall classification accuracy decreases strikingly for all machine learning methods used. Comparing the six different algorithms, it can be seen that bagging out-competes the other algorithms on the average results (Table 4). The best results over all feature subsets (Table 5) are either achieved with kNN, bagging or random forests.

Feature Subset Selection. A very interesting observation is that the (again very small) wrapper subsets clearly outperform the IG subsets, in contrast to the results for the e-mail data also for IG subsets with 30-100 features. The classification results using all features are similar to the best IG results, and therefore also much lower than the wrapper results. The three best wrapper results (kNN, random forest and JRip) were all achieved with a feature set containing only 6 features. Interestingly, only two features appear in all three subsets, the other features are diverse. Figure 3 shows the classification performance for the IG subsets with different sizes. There is a sharp distinction between this curve and the curve shown in Figure 1 (IG subsets for the e-mail data). Although there is a small decline in the results when the number of attributes is decreased, even with very small IG gain subsets the classification performance remains acceptable (compared to big IG subsets).

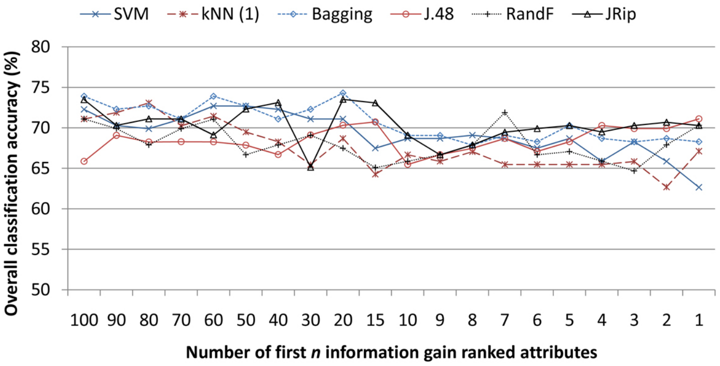

Fig. 3. Drug discovery data – information gain subsets

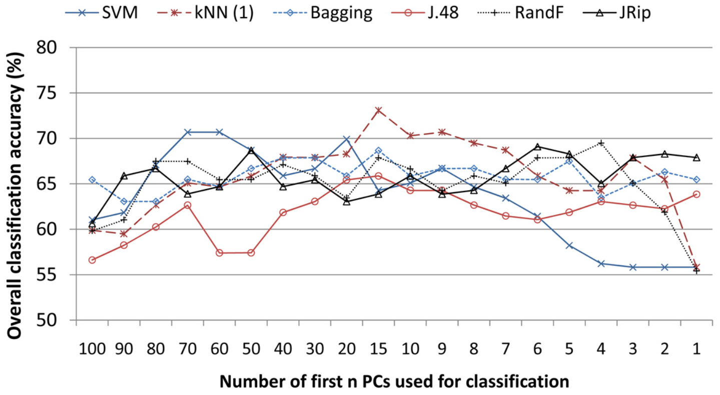

Fig. 4. Drug discovery data – PCA (A) subsets

PCA. Figure 4 shows the overall classification accuracy for PCA_1. The results are again very different to the results from the e-mail data set. Surprisingly, the results using between the first 90 to 100 PCs, as well as the results using only one PC are much lower than the results for other PC numbers. For subsets between 3 and 80 PCs, the average classification result combining all machine learning methods are always between 64.1% and 67.3%. When comparing PCA_1 (using the covariance matrix) and PCA_2 (using the correlation matrix), it can be seen that the results for some classifiers change significantly. The best result for SVM (see Table 5) decreases by 5% even though both subsets achieved the best result with the same number of PCs (60). On the other hand, for bagging and random forest the results increased by about 4%. The best bagging results were again achieved with the same number of PCs (15). The best PCA subsets for this data set is PCA_3, the standardized data set (see Tables 4 and 5).

Explaining the variance. The amount of variance captured by the first n PCs is much higher than for the e-mail data set (cf. Table 6). This is due to the smaller number of features (249 compared to 10 000) and the fact that the e-mail data set is very sparse (which is not the case for the DD data set).

Table 6. DD data – amount of variance captured by first n PCs (max. dim: 249).

PCs	100	80	60	40	20	10	8	6	5	4	3	2	1
Cov	100%	99.9%	99.9%	99.9%	99.9%	99.8%	99.6%	99.3%	98.7%	98.1%	97.0%	94.6%	87.9%
Corr	99.6%	99.1%	97.9%	95.2%	88.7%	79.9%	76.5%	71.5%	68.2%	63.6%	58.1%	51.4%	40.6%

5 Conclusion

We investigated the relationship between various feature subset selection (FS) and dimensionality reduction (DR) methods and the resulting classification performance based on data sets from two distinct application contexts, e-mail classification and drug discovery (DD). We compared the classification results for subsets containing some of the original attributes (extracted with a wrapper method and information gain (IG)) with subsets containing linear combinations of the original attributes, computed with three variants of the principle component analysis (PCA). Moreover, we investigated the relationship between the variability captured in the linear combinations of PCA and the classification performance of machine learning (ML) algorithms. The observations made lead to the following conclusions.

1. Comparison of the FS methods reveals that wrapper methods clearly outperform IG on the DD data, and also show very good performance on the e-mail data set. Although for the e-mail data the best overall IG subset achieves better results than the wrapper subsets, the wrapper subsets lead on average to better results than the IG subsets with comparable sizes.
2. Looking at the ML algorithms, SVMs show surprisingly low classification results on PCA subsets. Although SVMs perform very well when all features or subsets of the original features are used, SVMs achieve only the lowest results for all three PCA subsets of the e-mail data. On the DD data, only the PCA_3 results are better than the average results over all six classifiers.
3. A comparison of the PCA subsets with IG and wrapper shows that PCA combined with machine learning is highly sensitive to the characteristics of the data. For the sparse e-mail data set with much more instances than attributes the PCA results are better and much more stable with respect to the size of the feature set (see Figures 1 and 2). For the DD data set, which contains more attributes than instances and only few null entries, the PCA results are much lower than the results achieved with wrappers or IG.
4. The amount of variability captured in the first n PCs does not have much influence on the classification performance. Looking at Table 3 it can be seen that the first n PCs capture much more of the variability of the original data when the covariance is used for the PCA (PCA_1) than when the correlation is used (PCA_2). Nevertheless, the classification performances of both variants are very similar (cf. Table 1). Moreover, normalizing the e-mail features by their standard deviation (PCA_3) does not change the classification performance. This normalization step seems to slightly increase the classification performance on the DD data whereas there covariance and correlation results do not vary much.

Overall, very significant variations in the classification accuracy and in the influence of different feature subsets on this accuracy can be observed, obviously also depending on the type of data. It is clear that more investigation is needed in the complex connections between feature selection and classification accuracy.

References

1. Apache Software Foundation, *SpamAssassin open-source spam filter*, 2006. http://spamassassin.apache.org/.
2. A. L. Blum and P. Langley, *Selection of relevant features and examples in machine learning*, J. Artif. Intell., 97 (1997), pp. 245–271.
3. L. Breiman, *Bagging predictors*, J. of Machine Learning, 24 (1996), pp. 123–140.
4. ——, *Random forests*, J. of Machine Learning, 45 (2004), pp. 5–32.
5. W. W. Cohen, *Fast effective rule induction*, in Proc. of the 12th International Conference on Machine Learning, A. Prieditis and S. Russell, eds., Morgan Kaufmann, 1995, pp. 115–123.
6. G. V. Cormack and T. R. Lynam, *Trec 2005 spam public corpora*, 2005. http://plg.uwaterloo.ca/cgi-bin/cgiwrap/gvcormac/foo.
7. T. M. Cover and P. E. Hart, *Nearest neighbor pattern classification*, IEEE Transactions on Information Theory, 8 (1995), pp. 373–389.
8. P. Geladi and B. Kowalski, *Partial least-squares regression: A tutorial*, Analytica Chimica Acta, (1986), pp. 1–17.
9. R. L. Gorsuch, *Factor Analysis*, Lawrence Erlbaum, 2nd ed., 1983.
10. I. Guyon and A. Elisseeff, *An introduction to variable and feature selection*, J. Mach Learn Res, 3 (2003), pp. 1157–1182.
11. T. Howley, M. G. Madden, M.-L. O'Connell, and A. G. Ryder, *The effect of principal component analysis on machine learning accuracy with high-dimensional spectral data*, Knowl.-Based Syst., 19 (2006), pp. 363–370.
12. I. T. Jolliffe, *Principal Component Analysis*, Springer, 2nd ed., 2002.
13. R. Kohavi and G. H. John, *Wrappers for feature subset selection*, Artificial Intelligence, 97 (1997), pp. 273–324.
14. D. Koller and M. Sahami, *Toward optimal feature selection*, 1996, pp. 284–292.
15. H. Liu and H. Motoda, *Feature Selection for Knowledge Discovery and Data Mining*, Kluwer Academic Publishers, 1998.
16. L. C. Molina, L. Belanche, and A. Nebot, *Feature selection algorithms: A survey and experimental evaluation*, 2002.
17. B. C. Moore, *Principal component analysis in linear systems: Controllability, observability and model reduction*, IEEE Transactions on Automatic Control, 26 (1981), pp. 17–32.
18. J. Platt, *Machines using sequential minimal optimization*, in Advances in Kernel Methods - Support Vector Learning, B. Schoelkopf, C. Burges, and A. Smola, eds., MIT Press, 1998.
19. L. Popelinsky, *Combining the principal components method with different learning algorithms*, in Proc. of ECML/PKDD2001 IDDM Workshop, 2001.
20. W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, Wiley-Interscience, 1st ed., 2007.
21. R. J. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann, 1993.
22. P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, Addison Wesley, 1st ed., may 2005.
23. I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, 2 ed., 2005.