

LESS: A Model-Based Classifier for Sparse Subspaces

Cor J. Veenman and David M.J. Tax

Abstract—In this paper, we specifically focus on high-dimensional data sets for which the number of dimensions is an order of magnitude higher than the number of objects. From a classifier design standpoint, such small sample size problems have some interesting challenges. The first challenge is to find, from all hyperplanes that separate the classes, a separating hyperplane which generalizes well for future data. A second important task is to determine which features are required to distinguish the classes. To attack these problems, we propose the LESS (Lowest Error in a Sparse Subspace) classifier that efficiently finds linear discriminants in a sparse subspace. In contrast with most classifiers for high-dimensional data sets, the LESS classifier incorporates a (simple) data model. Further, by means of a regularization parameter, the classifier establishes a suitable trade-off between subspace sparseness and classification accuracy. In the experiments, we show how LESS performs on several high-dimensional data sets and compare its performance to related state-of-the-art classifiers like, among others, linear ridge regression with the LASSO and the Support Vector Machine. It turns out that LESS performs competitively while using fewer dimensions.

Index Terms—Classification, support vector machine, high-dimensional, feature subset selection, mathematical programming.

1 INTRODUCTION

IN applications nowadays, data sets with hundreds or even thousands of features are no exception. For the representation of distributions in these high-dimensional feature spaces, there are hardly ever enough objects. According to a rule of thumb, for the estimation of data distributions, the number of objects should be an order of magnitude higher than the number of dimensions. In this paper, we consider data sets where the number of objects is even orders of magnitude lower than the number of features. Accordingly, the space is not just too sparsely sampled, almost any arbitrary hyperplane can separate the space into the desired classes.

A classifier that is known for being robust is the Nearest Mean Classifier. It has already been successfully applied to this type of data [22]. To be applied to such high-dimensional data, a suitable feature subset has to be selected for optimal performance. In [22], a naive filtering approach has been chosen. Alternative feature selection methods can be applied that are less greedy like forward selection or backward elimination, see, e.g., [19]. Ultimately, even a combinatorial optimization problem can be defined that aims at finding the feature subset with the highest classification performance in a wrapper framework [15]. The obvious drawback of the latter approach is that it is computationally intractable, though approximations through genetic algorithms, e.g., [14], simulated annealing [8], or tabu search [24] have been reported.

In this paper, we choose another approach. Instead of selecting a suitable feature subset, we introduce a weighting factor for each dimension. The feature subset selection problem then turns into a problem of finding the weight factors, where a zero weight effectively rules out the respective feature. We propose the LESS classifier which is a Weighted Nearest Mean Classifier that balances classification errors with model sparseness. The LESS classifier can be seen as a variant of the L_1 Support Vector Machine. The main

difference with related classifiers for high-dimensional data sets like the SVM is that the LESS classifier employs a model of the class distributions. Accordingly, in general, higher classification accuracy can be achieved in a lower-dimensional subspace.

In the next section, we motivate and propose the LESS classifier. In the following section, we describe a number of related classifiers that have proven to be adequate for high-dimensional data. Then, we evaluate the LESS classifier and compare its performance to the described related classifiers and we discuss the results in the final section.

2 THE LESS CLASSIFIER MODEL

We first formalize the problem. Let $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ be a data set with n objects, where $\mathbf{x}_i$ is a feature vector in a p -dimensional metric space. For the sake of simplicity, we consider only two-class problems. Then, each object has a class label y_i being either -1 or $+1$. Let μ_1 be the mean vector of the n_1 objects $X_1 \subset X$ with label -1 , μ_2 the mean vector of the n_2 objects $X_2 \subset X$ with label $+1$, and μ_0 the mean vector of the whole data set. Further, Σ_k is the covariance matrix of objects X_k and the variance vector Σ_k^2 is the diagonal of Σ_k . We denote the predicted label for object $\mathbf{x}$ with $f(\mathbf{x})$.

For the LESS classifier, we established two important design criteria. First, it should have high classification performance on high-dimensional data or small sample size problems. We consider data sets where the number of features is orders of magnitude higher than the number of objects, e.g., 10-100 objects with 1,000-10,000 dimensions. Second, the classifier should use as few dimensions as possible in achieving its performance.

These objectives are also the underlying design principles of a number of related classifiers that we describe in the *Related Work* Section, i.e., the Support Vector Machine [7], [23], Liknon [3], Ridge Regression with the LASSO [20], and the Nearest Shrunken Centroids [21]. These classifiers achieve these objectives by formulating a minimization problem consisting of a classification error term and a complexity term. Among these classifiers, only the Nearest Shrunken Centroids classifier incorporates a model of the class distributions. Especially, for the extreme low sample size problems that we study, such a model bias is expectedly beneficial.

Also, the proposed LESS classifier employs a data model. We base the LESS classifier on the Nearest Mean Classifier, which assumes statistically independent features with equal variances, i.e., $\Sigma_1 = \Sigma_2 = \sigma^2 I$, see [9, Section 2.6.1]. Accordingly, for that classifier, the classes can be modeled with their means only. An extension of the Nearest Mean Classifier that naturally allows for feature selection is the Weighted Nearest Mean Classifier. The Weighted Nearest Mean Classifier is defined as:

$$f(\mathbf{x}) = \begin{cases} -1, & \text{if } d_m^2(\mathbf{x}, \mu_1) - d_m^2(\mathbf{x}, \mu_2) < 0 \\ +1, & \text{otherwise,} \end{cases} \quad (1)$$

where $d_m^$

condition that all training objects must be classified correctly. Formally, this can be written as:

$$\forall i : y_i \left(d_m^2(\mathbf{x}_i, \boldsymbol{\mu}_1) - d_m^2(\mathbf{x}_i, \boldsymbol{\mu}_2) \right) = \quad (3)$$

$$\forall i : y_i : \sum_{j=1}^p m_j^2 \left((x_{ij} - \mu_{1j})^2 - (x_{ij} - \mu_{2j})^2 \right) = \quad (4)$$

$$\forall i : y_i : \sum_{j=1}^p m_j^2 \left(2x_{ij}(\mu_{2j} - \mu_{1j}) + (\mu_{1j}^2 - \mu_{2j}^2) \right) \geq 1, \quad (5)$$

which imposes a margin between the classes similarly to the margin defined for the Support Vector Machine. In case the classes are not linearly separable, we must allow for misclassifications. To this end, we introduce a slack variable ξ_i for each object constraint (5). These slack variables effectively release the constraints.

The objective of the LESS classifier is to find those weights that minimize the number of misclassifications ($\sum \xi_i$), while using as few features as possible ($\sum m_j^2$). The factors m_j that weigh the dimensions appear squared in the minimization term, so the optimization problem may seem quadratic. However, with some rewriting the problem turns into a linear optimization problem with linear constraints, also called a Linear Program (LP). We substitute $w_j = m_j^2$ and get the Lowest Error in a Sparse Subspace (LESS):

$$\min \sum_{j=1}^p w_j + C \sum_{i=1}^n \xi_i, \quad (6)$$

$$\text{subject to : } \forall i : y_i : \sum_{j=1}^p w_j \left(2x_{ij}(\mu_{2j} - \mu_{1j}) + (\mu_{1j}^2 - \mu_{2j}^2) \right) \geq 1 - \xi_i, \quad (7)$$

$$\forall i : \xi_i \geq 0, \quad (8)$$

$$\forall j : w_j \geq 0. \quad (9)$$

Fortunately, these Linear Programming problems can be solved efficiently even with thousands of variables (w_j s and ξ_i s in this case) and constraints. It can be seen that, in contrast with the Nearest Shrunk Centroids classifier [21], the LESS classifier finds the optimal feature weights in a combinatorial fashion. As such, the LESS classifier is more flexible, or, in other words, it has less model bias.

We now focus on the LESS constraints as formulated in (7). These constraints can be written as:

$$\forall i : y_i : \mathbf{w}' \Phi(\mathbf{x}_i) \geq 1 - \xi_i, \quad (10)$$

$$\text{with } \mathbf{w} = (w_1 \ w_2 \ \dots \ w_p)' \text{ and}$$

$$\Phi(\mathbf{x}_i) = (\phi(x_{i1}) \ \phi(x_{i2}) \ \dots \ \phi(x_{ip}))', \quad (11)$$

$$\text{where } \phi(x_{ij}) = 2x_{ij}(\mu_{2j} - \mu_{1j}) + (\mu_{1j}^2 - \mu_{2j}^2). \quad (12)$$

So, the function ϕ scales and translates the data vectors using the class means. Denoted as such, LESS shows similarities with SVM formulations. We have a closer look at this in the *Related Work* Section. We call the mapping (12) ϕ_μ and the corresponding LESS version LESS $_\mu$. Further, we propose a few other mappings leading to other LESS variants with interesting properties.

First, the choice for the Nearest Mean Classifier was motivated for its low complexity and, therefore, its stability. Nevertheless, the computation of the mean per dimension as we did so far is sensitive for outliers. An alternative, more robust estimate of the class prototype is the median $\tilde{\boldsymbol{\mu}}_k = \text{median}(X_k)$. The median can easily be substituted in (7) or (12) leading to a more robust LESS realization. We denote LESS with medians as class prototypes with LESS $_{\tilde{\mu}}$ and the mapping function with $\phi_{\tilde{\mu}}$.

Second, we propose a mapping that scales the distances to the class means with the variance in the respective dimension. This mean/variance mapping $\phi_{\mu\sigma}$ results in the nonlinear (quadratic) LESS $_{\mu\sigma}$ classifier and is defined as:

$$\phi_{\mu\sigma} = \frac{(x_{ij} - \mu_{1j})^2}{\sigma_{1j}^2} - \frac{(x_{ij} - \mu_{2j})^2}{\sigma_{2j}^2}. \quad (13)$$

Clearly, the previous extensions to LESS can also be combined. Especially, the robust class prototypes can be combined with variance scaling resulting in LESS $_{\tilde{\mu}\sigma}$. This means that in (13), $\boldsymbol{\mu}_k$ must be substituted with the median $\tilde{\boldsymbol{\mu}}_k$. It is then also more natural to replace the variances σ_k^2 with the median squared deviation $\tilde{\sigma}_k^2$ in the same equation. This results in the mapping $\phi_{\tilde{\mu}\sigma}$, where $\tilde{\sigma}_k^2$ is defined as:

data vectors are mapped, either linearly $(\phi_\mu, \phi_{\bar{\mu}})$ or nonlinearly $(\phi_{\mu\sigma}, \phi_{\bar{\mu}\sigma})$, and that a different weight vector norm is minimized. Further, LESS has an explicit bias term $(\mu_{1j}^2 - \mu_{2j}^2)$, while the bias term b in (17) is estimated. Moreover, for LESS the weights $w_j \geq 0$.

3.3 L_1 Support Vector Machine (Liknon)

The Liknon classifier was designed for class prediction with a low number of relevant features [3], which are the same design criteria as for LESS. The model is very similar to the linear SVM model (16)-(17). The only difference is that the L_1 -norm of the weight vector is minimized instead of the squared L_2 -norm. As such, Liknon is considered an L_1 -SVM. For more on L_1 -SVM classifiers, see, for instance, [5] and [11]. The motivation for applying the L_1 -norm is the same as for the LASSO (see the previous section). That is, the L_1 -norm forces the weight entries w_j to 0 so the classifier explicitly selects a feature subset. The minimization problem is stated as:

$$\min |\mathbf{w}| + C \sum_{i=1}^n \xi_i \quad (18)$$

$$\text{subject to: } \forall i: y_i(\mathbf{w}'\mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0. \quad (19)$$

This seemingly nonlinear problem can be rewritten into a linear optimization problem with linear constraints by substituting $\mathbf{w} = (\mathbf{u} - \mathbf{v})$ and $|\mathbf{w}| = (\mathbf{u} + \mathbf{v})$ with $u_j, v_j \geq 0$.

As an SVM variant also, Liknon resembles LESS to a certain extent. In addition to the L_2 SVM similarities, the L_1 -norm that is minimized in (18) is the same as the weight vector minimization term in (6) given that for LESS the weights $w_j \geq 0$.

3.4 Nearest Shrunk Centroids (NSC)

The last classifier that we consider is not formulated as a mathematical programming problem. This classifier assigns objects to the class to which shrunken centroid they are closest, therefore, the name Nearest Shrunk Centroids [21]. The shrunken centroids are the means of the classes, for which each of the feature components are reduced by a factor Δ until the feature component becomes 0. This classifier is defined as follows:

$$f(\mathbf{x}) = \begin{cases} -1, & \text{if } \sum_{j=1}^p \frac{(x_j - |\mu_{1j}|_\Delta)^2}{(s_1 + s_0)^2} - \frac{(x_j - |\mu_{2j}|_\Delta)^2}{(s_2 + s_0)^2} < 0, \\ +1, & \text{otherwise,} \end{cases} \quad (20)$$

where s_0 is a regularization term and $|\mu_{kj}|_\Delta = \mu_{0j} + m_k(s_j + s_0)|d_{kj}|_\Delta$ is the shrunken centroid with:

$$|d_{kj}|_\Delta = \text{sign}(d_{kj})(|d_{kj}| - \Delta) \text{ and } s_j^2 = \frac{1}{n-2}(\sigma_{1j}^2 + \sigma_{2j}^2). \quad (21)$$

In this expression, Δ is the shrinkage parameter. Without shrinkage ($\Delta = 0$) the means are not scaled or $|\mu_{kj}|_\Delta = \mu_{kj}$. Finally, d_{kj} is defined as:

$$d_{kj} = \frac{\mu_{kj} - \mu_{0j}}{m_k(s_j + s_0)} \text{ and } m_k = \sqrt{\frac{1}{n_k} + \frac{1}{n}} \text{ with } k \in \{1, 2\}. \quad (22)$$

By scaling the classes with the variances, the NSC results in a quadratic classifier similar to LESS $_{\mu\sigma}$.

4 EXPERIMENTS

We tested the classifiers on artificial and several real-life high-dimensional data sets. The real-life data sets range from biological tissue classification with microarrays to some well-known data sets from the Machine Learning Database Repository [4]. In Table 1, we list the characteristics of the real-life data sets.

For the optimization of the linear programs for LESS and Liknon, we used the GLPK solver [10]. The Support Vector Machine is implemented using the quadratic programming solver from [16]. Further, we used the efficient LASSO implementation as made available by the authors from [17].

TABLE 1
Overview of the Characteristics of the Data Sets Used in the Experiments

Dataset	Ref.	class I (n_1)	class II (n_2)	total (n)	features (p)
Colon	[1]	22	40	62	1908
Leukemia	[12]	25	47	72	3571
Metastasis	[22]	34	44	78	4919
Ionosphere	[18]	126	225	351	34
Sonar	[13]				

TABLE 2
The Mean and Standard Deviation of the Generalization Error Obtained with Cross-Validation Tests

Dataset	LESS $_{\mu}$	LESS $_{\mu}^{\pm}$	LESS $_{\mu}^{\pm b}$	LESS $_{\mu\sigma}$	LESS $_{\tilde{\mu}}$	LESS $_{\tilde{\mu}\tilde{\sigma}}$	SVM	LASSO	Liknon	NSC
colon	11.7 $\pm$ 2.6	12.8 $\pm$ 2.4	12.0 $\pm$ 2.3	12.7 $\pm$ 2.8	11.5 $\pm$ 1.5	13.5 $\pm$ 1.1	13.8 $\pm$ 1.8	16.2 $\pm$ 2.9	11.9 $\pm$ 2.8	12.7 $\pm$ 2.0
leukemia	10.3 $\pm$ 2.2	9.9 $\pm$ 2.4	9.5 $\pm$ 1.5	10.8 $\pm$ 1.7	10.3 $\pm$ 2.0	7.8 $\pm$ 2.8	2.7 $\pm$ 0.7	8.8 $\pm$ 2.9	11.7 $\pm$ 1.4	3.5 $\pm$ 0.7
metas	33.4 $\pm$ 3.4	33.2 $\pm$ 2.5	31.3 $\pm$ 2.2	34.0 $\pm$ 3.8	33.7 $\pm$ 2.9	31.7 $\pm$ 4.4	39.8 $\pm$ 2.7	31.6 $\pm$ 2.1	36.8 $\pm$ 3.5	31.5 $\pm$ 3.9
iono	18.0 $\pm$ 1.1	19.0 $\pm$ 0.6	16.0 $\pm$ 1.0	9.5 $\pm$ 0.7	21.3 $\pm$ 1.0	19.9 $\pm$ 1.4	16.4 $\pm$ 0.6	21.8 $\pm$ 1.0	16.6 $\pm$ 1.0	22.4 $\pm$ 0.8
sonar	24.5 $\pm$ 1.8	24.9 $\pm$ 2.4	24.3 $\pm$ 1.8	21.0 $\pm$ 1.0	24.5 $\pm$ 2.2	23.8 $\pm$ 1.2	24.6 $\pm$ 1.8	26.4 $\pm$ 1.4	26.8 $\pm$ 1.5	26.9 $\pm$ 1.5
Average	19.6	20.0	18.6	17.6	20.3	19.3	19.4	21.0	20.8	19.4

TABLE 3
The Mean and Standard Deviation of the Number of Dimensions Found in the Cross-Validation Tests

Dataset	LESS $_{\mu}$	LESS $_{\mu}^{\pm}$	LESS $_{\mu}^{\pm b}$	LESS $_{\mu\sigma}$	LESS $_{\tilde{\mu}}$	LESS $_{\tilde{\mu}\tilde{\sigma}}$	SVM	LASSO	Liknon	NSC
colon	13.3 $\pm$ 1.2	14.0 $\pm$ 0.9	13.7 $\pm$ 1.1	28.8 $\pm$ 2.2	16.7 $\pm$ 1.0	15.5 $\pm$ 1.5	1908	30.0 $\pm$ 3.1	20.7 $\pm$ 1.6	60.4 $\pm$ 31.8
leukemia	7.0 $\pm$ 0.8	7.8 $\pm$ 1.1	1.9 $\pm$ 0.4	9.4 $\pm$ 1.0	6.7 $\pm$ 0.7	9.3 $\pm$ 0.7	3571	26.2 $\pm$ 2.5	7.8 $\pm$ 1.8	2.7e3 $\pm$ 258
metas	5.0 $\pm$ 1.4	5.7 $\pm$ 2.4	6.1 $\pm$ 2.2	13.8 $\pm$ 1.8	4.1 $\pm$ 1.7	12.3 $\pm$ 1.8	4919	5.7 $\pm$ 2.0	17.8 $\pm$ 3.1	87.3 $\pm$ 153
iono	15.7 $\pm$ 1.5	15.0 $\pm$ 3.2	28.0 $\pm$ 1.8	11.1 $\pm$ 0.4	13.9 $\pm$ 1.0	6.1 $\pm$ 0.2	34	31.2 $\pm$ 1.5	27.7 $\pm$ 1.1	6.8 $\pm$ 1.2
sonar	12.9 $\pm$ 1.9	11.5 $\pm$ 3.7	12.3 $\pm$ 1.7	18.6 $\pm$ 2.1	18.1 $\pm$ 1.3	16.1 $\pm$ 1.9	60	25.6 $\pm$ 3.5	25.5 $\pm$ 3.8</	

incorporated in case the classes have different variances. The consequent quadratic classifier can still easily be optimized as a Linear Program. Further, we showed that robust estimations of the mean and variance can also be applied with the same profitable optimization properties.

The main difference between the LESS and other classifiers like the SVM, Liknon, and LASSO is that LESS utilizes a model of the data. Such a model can be beneficial in cases where the data is insufficient to reliably estimate a suitable discriminant based on training data only. This is especially true for small sample size problems. In the experiments, we compared these classifiers to the proposed LESS classifier alternatives. We compared both the resulting generalization error and the number of utilized features. It turns out that all LESS variants have some benefits. Overall, the LESS classifiers perform competitively while they utilize the lowest number of features.

We also showed that the LESS model inherently contains a data mapping, which is part of the difference between LESS, SVM, and Liknon. In the experiments, we incorporated this data mapping in the Liknon classifier. It turned out that the mapping is a fundamental part of the strength of the LESS classifier. Further research should be directed toward exploiting the data mapping in other classifiers.

ACKNOWLEDGMENTS

This research is supported by PAMGene within the BIT program ISAS-1. The authors thank Dr. Berwin Turlach for his help and for making the LASSO software available.

REFERENCES

- [1] U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, and A.J. Levine, "Broad Patterns of Gene Expression Revealed by Clustering of Tumor and Normal Colon Tissues Probed by Oligonucleotide Arrays," *Proc. Nat'l Academy of Sciences*, vol. 96, no. 12, pp. 6745-6750, 1999.
- [2] C.G. Atkeson, A.W. Moore, and S. Schaal, "Locally Weighted Learning," *Artificial Intelligence Rev.*, vol. 11, pp. 11-73, 1997.
- [3] C. Bhattacharyya, L.R. Grate, A. Rizki, D. Radisky, F.J. Molina, M.I. Jordan, M.J. Bissel, and I.S. Mian, "Simultaneous Classification and Relevant Feature Identification in High-Dimensional Spaces: Application to Molecular Profiling Data," *Signal Processing*, vol. 83, pp. 729-743, 2003.
- [4] C.L. Blake and C.J. Merz, "UCI Repository of Machine Learning Databases," 1998.
- [5] P.S. Bradley and O.L. Mangasarian, "Feature Selection via Concave Minimization and Support Vector Machines," *Proc. 15th Int'l Conf. Machine Learning*, pp. 82-90, 1998.
- [6] L. Breiman, "Better Subset Selection Using Non-Negative Garotte," technical report, Univ. of California, Berkeley, 1993.
- [7] C.J.C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121-167, 1998.
- [8] J.C.W. Debus and V.J. Rayward-Smith, "Feature Subset Selection within a Simulated Annealing Data Mining Algorithm," *J. Intelligent Information Systems*, vol. 9, no. 1, pp. 57-81, 1997.
- [9] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification*. New York: John Wiley and Sons, Inc., 2001.
- [10] "Free Software Foundation," *GNU Linear Programming Kit*, <http://www.gnu.org>, 2005.
- [11] G. Fung and O.L. Mangasarian, "Data Selection for Support Vector Machine Classifiers," *Proc. Sixth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining*, R. Ramakrishnan and S. Stolfo, eds., vol. 2094, pp. 64-70, Aug. 2000.
- [12] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield, and E.S. Lander, "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring," *Science*, vol. 286, no. 5439, pp. 531-537, 1999.
- [13] R.P. Gorman and T.J. Sejnowski, "Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets," *Neural Networks*, vol. 1, pp. 75-89, 1988.
- [14] M. Karzynski, A. Mateos, J. Herrero, and J. Dopazo, "Using a Genetic Algorithm and a Perceptron for Feature Selection and Supervised Class Learning in DNA Microarray Data," *Artificial Intelligence Rev.*, vol. 20, pp. 39-51, 2003.
- [15] R. Kohavi and G.H. John, "Wrappers for Feature Subset Selection," *Artificial Intelligence*, vol. 97, pp. 273-324, Dec. 1997.
- [16] "Mosek Optimization Toolbox," <http://www.mosek.com>, 2005.
- [17] M.R. Osborne, B. Presnell, and B.A. Turlach, "On the LASSO and Its Dual," *J. Computational and Graphical Statistics*, vol. 9, no. 2, pp. 319-337, 2000.
- [18] V.G. Sigillito, S.P. Wing, L.V. Hutton, and K.B. Baker, "Classification of Radar Returns from the Ionosphere Using Neural Networks," *Johns Hopkins APL Technical Digest*, vol. 10, pp. 262-266, 1989.
- [19] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*. London: Academic Press, 1999.
- [20] R. Tibshirani, "Regression Shrinkage and Selection via the LASSO," *J. Royal Statistical Soc. B*, vol. 58, no. 1, pp. 267-288, 1996.
- [21] R. Tibshirani, T. Hastie, B. Balasubramanian, and G. Chu, "Diagnosis of Multiple Cancer Types by Shrunken Centroids of Gene Expression," *Proc. Nat'l Academy of Sciences*, vol. 99, no. 10, pp. 6567-6572, 2002.
- [22] L.J. van 't Veer, H. Dai, M.J. van de Vijver, Y.D. He, A.A.M. Hart, M. Mao, H.L. Peterse, K. van der Kooy, M.J. Marton, A.T. Witteveen, G.J. Schreiber, R.M. Kerkhoven, C. Roberts, P.S. Linsley, R. Bernards, and S.H. Friend, "Gene Expression Profiling Predicts Clinical Outcome of Breast Cancer," *Nature*, vol. 415, pp. 530-536, Jan. 2002.
- [23] V. Vapnik, <