POLITECNICO DI TORINO
Repository ISTITUZIONALE

Sparse 1 and 2 center classifiers

Original
Sparse 1 and 2 center classifiers / Calafiore, G. C.; Fracastoro, G.. - ELETTRONICO. - 53:(2020), pp. 518-523. (Intervento presentato al convegno 21st IFAC World Congress 2020 tenutosi a Berlin, Germany nel 11-17 July, 2020) [10.1016/j.ifacol.2020.12.322].

Availability:
This version is available at: 11583/2957265 since: 2022-03-09T12:02:24Z

Publisher:
Elsevier B.V.

Published
DOI:10.1016/j.ifacol.2020.12.322

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

25 April 2024

Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 53-2 (2020) 518–523

Sparse ℓ_1 and ℓ_2 Center Classifiers [1]

Giuseppe C. Calafiore * **Giulia Fracastoro** **

* *DET Politecnico di Torino and IEIIT CNR*

** *DET Politecnico di Torino*

Abstract: The nearest-centroid classifier is a simple linear-time classifier based on computing the centroids of the data classes in the training phase, and then assigning a new datum to the class corresponding to its nearest centroid. Thanks to its very low computational cost, the nearest-centroid classifier is still widely used in machine learning, despite the development of many other more sophisticated classification methods. In this paper, we propose two sparse variants of the nearest-centroid classifier, based respectively on ℓ_1 and ℓ_2 distance criteria. The proposed sparse classifiers perform simultaneous classification and feature selection, by detecting the features that are most relevant for the classification purpose. We show that training of the proposed sparse models, with both distance criteria, can be performed exactly (i.e., the globally optimal set of features is selected) and at a quasi-linear computational cost. The experimental results show that the proposed methods are competitive in accuracy with state-of-the-art feature selection techniques, while having a significantly lower computational cost.

Copyright © 2020 The Authors. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0)

Keywords: Nearest-centroid Classifier, Machine learning, Sparse optimization, Feature selection, Text classification.

1. INTRODUCTION

In the last years the technological development has led to a massive proliferation of large-scale datasets. The processing of these large amounts of data poses many new challenges and there is a strong need of algorithms that scale mildly (e.g., linearly or quasi-linearly) with the dataset size. For this reason, classification methods with a very low computational cost, such as Naive Bayes (McCallum et al., 1998; Jiang et al., 2007) and the nearest centroid classifier (Manning et al., 2008; Tibshirani et al., 2002), are an appealing choice in this endeavour. In many cases, these methods are the only feasible approaches, since more sophisticated techniques would be too demanding from a computational point of view.

When the number of features in a datasets is very high, feature selection is a necessary step of any machine learning algorithm. Feature selection consists in detecting the most relevant features of the dataset. Besides reducing the dataset size, feature selection has some other important advantages. First, it eliminates noisy or irrelevant features, reducing the risk of overfitting. Second, by selecting only the most significant features, it improves the interpretability of the model. State-of-the-art feature selection methods are usually based on some heuristics without any guarantee of optimality. Some of them, such as Lasso (Tibshirani, 1996) or ℓ_1-regularized logistic regression (Ng, 2004), are based on a convex optimization problem with a ℓ_1-norm penalty on the regression coefficients to promote sparsity. The main drawback of these techniques is that they are usually computationally expensive. Other methods, such as Odds Ratio (Mladenic and Grobelnik, 1999), propose a different approach that employs a feature ranking based on their inherent characteristics. These methods are usually very fast, but often their performance in terms of accuracy is poor. Recently, Askari et al. (2019) have presented a feature selection method targeted for a Naive Bayes classifier. This method can provide an optimal solution in the case of binary data, and an approximate upper bound for general data.

In this paper, we propose sparse nearest-center classifiers that guarantee both global optimality and numerical efficiency. The proposed methods simultaneously perform feature selection and classification. We discuss two variants of the approach, namely an ℓ_1-sparse center classifiers and an ℓ_2-sparse center classifier, in which we consider the ℓ_1 and the ℓ_2 distance criteria, respectively. The ℓ_2 case is a sparse variant of the nearest centroid classifier (Manning et al., 2008; Tibshirani et al., 2002), which is a widely used classifier, especially in text classification (Han and Karypis, 2000). Instead, the ℓ_1 case is related to the median classifier (Hall et al., 2009; Jörnsten, 2004), that has shown to be more robust to outliers than the ℓ_2 version. We prove that both the proposed methods select the optimal subset of features for the corresponding classifier, in quasi-linear time. The experimental results show that the proposed techniques achieve similar performance as state-of-the-art feature selection methods, but at a substantially lower computational cost.

2. PRELIMINARIES ON CENTER-BASED CLASSIFIERS

Let

$$X = \begin{bmatrix} x^{(1)} & \cdots & x^{(n)} \end{bmatrix} \in \mathbb{R}^{m,n}, \tag{1}$$

be a given data matrix whose columns $x^{(j)} \in \mathbb{R}^m$, $j = 1, \ldots, n$, contain feature vectors from n observations, and let $\mathbf{y} \in \mathbb{R}^n$ be a given vector such that $y_j \in \{-1, +1\}$ is the class label corresponding to the j-th observation. We

[1] This research was funded in part by sumup.ai.

2405-8963 Copyright © 2020 The Authors. This is an open access article under the CC BY-NC-ND license.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2020.12.322

consider a binary classification problem, in which a new observation vector $x \in \mathbb{R}^m$ is to be assigned to the positive class C_+ (corresponding to $y = +1$) or to the negative class C_- (corresponding to $y = -1$). To this purpose, the *nearest centroid classifier* is a well-known classification model, which works by assigning the class label based on the least Euclidean distance from x to the centroids of the classes. The centroids are computed on the basis of the training data as

$$\bar{x}^+ = \frac{1}{n_+} \sum_{j \in \mathcal{J}^+} x^{(j)}, \quad \bar{x}^- = \frac{1}{n_-} \sum_{j \in \mathcal{J}^-} x^{(j)}, \tag{2}$$

where $\mathcal{J}^+ \doteq \{j \in \{1, \dots, n\} : y_j = +1\}$ contains the indices of the observations in the positive class, $\mathcal{J}^- \doteq \{j \in \{1, \dots, n\} : y_j = -1\}$ contains the indices of the observations in the negative class, and n_+, n_- are the cardinalities of $\mathcal{J}^+$ and $\mathcal{J}^-$, respectively. A new observation vector x is classified as positive or negative according to the sign of

$$\Delta_2(x) = \|x - \bar{x}^-\|_2^2 - \|x - \bar{x}^+\|_2^2,$$

that is, x is classified in the positive class if its Euclidean distance from the positive centroid is smaller than its distance from the negative centroid, and viceversa for the negative class. The discrimination surface for the centroid classifier is linear with respect to x, since

$$\begin{aligned} \Delta_2(x) &= \|x\|_2^2 + \|\bar{x}^-\|_2^2 - 2x^\top \bar{x}^- - \|x\|_2^2 - \|\bar{x}^+\|_2^2 + 2x^\top \bar{x}^+ \\ &= (\|\bar{x}^-\|_2^2 - \|\bar{x}^+\|_2^2) + 2x^\top (\bar{x}^+ - \bar{x}^-), \end{aligned} \tag{3}$$

where the coefficient in the linear term of the classifier is given by vector $w \doteq \bar{x}^+ - \bar{x}^-$. Notice that, whenever $\bar{x}_i^+ = \bar{x}_i^-$ for some component i (i.e., $w_i = 0$), the corresponding feature x_i in x is irrelevant for the purpose of classification.

Remark 1. We observe that the centroids in (2) can be seen as the optimal solutions to the following optimization problem:

$$\min_{\theta^+, \theta^- \in \mathbb{R}^m} \frac{1}{n_+} \sum_{j \in \mathcal{J}^+} \|x^{(j)} - \theta^+\|_2^2 + \frac{1}{n_-} \sum_{j \in \mathcal{J}^-} \|x^{(j)} - \theta^-\|_2^2. \tag{4}$$

That is, the centroids are the points that minimize the average squared distance to the samples within each class. A proof of this fact is immediate, by taking the gradient of the objective in (4) with respect to θ^+ and equating it to zero, and then doing the same thing for θ^-. The two problems are actually decoupled, so the two coefficients $1/n_+$ and $1/n_-$ play no role in terms of the optimal solution. However, they have been introduced for balancing the contribution of the residuals of the two classes. ⋆

We shall call (4) the (plain) ℓ_2-center classifier training problem, and Δ_2 in (3) the corresponding discrimination function. The usual centroids in (2) are thus the points that minimize the average ℓ_2 distance from the respective class representatives. This interpretation opens the way to considering different types of metrics for computing centers. In particular, there exist an extensive literature on the favorable properties of the ℓ_1 norm criterion, which is well known to provide estimates that are robust to outliers, see, e.g., Huber (1981); Rousseeuw and Leroy (1987). The natural ℓ_1 version of problem (4) is

$$\min_{\theta^+, \theta^- \in \mathbb{R}^m} \frac{1}{n_+} \sum_{j \in \mathcal{J}^+} \|x^{(j)} - \theta^+\|_1 + \frac{1}{n_-} \sum_{j \in \mathcal{J}^-} \|x^{(j)} - \theta^-\|_1, \tag{5}$$

which we shall call the (plain) ℓ_1-center classifier training problem. It is known that an optimal solution to problem (5) is obtained, for each $i = 1, \dots, m$, by taking θ_i^+ to be the *median* of the values $x_i^{(j)}$ in the positive class, and θ_i^- to be the median of the values $x_i^{(j)}$ in the negative class, see also the more general result given in Proposition 2. We let

$$\mu^+ \doteq \text{med}(\{x^{(j)}\}_{j \in \mathcal{J}^+}), \quad \mu^- \doteq \text{med}(\{x^{(j)}\}_{j \in \mathcal{J}^-}), \tag{6}$$

where med computes the median of its input vector sequence along each component, i.e., for each $i = 1, \dots, m$, μ_i^+ is the median of $\{x_i^{(j)}\}_{j \in \mathcal{J}^+}$, and μ_i^- is the median of $\{x_i^{(j)}\}_{j \in \mathcal{J}^-}$. The classification in the ℓ_1-center classifier is made by computing the distances from the new datum x and the ℓ_1 centers of the classes, and assigning x to the closest center, that is, we compute

$$\Delta_1(x) \doteq \|x - \mu^-\|_1 - \|x - \mu^+\|_1,$$

and assign x to the positive or negative class depending on the sign of $\Delta_1(x)$. We observe that, contrary to the ℓ_2 case, the discrimination criterion based on the sign of $\Delta_1(x)$ is not linear in x. However, expressed more explicitly in its components, $\Delta_1(x)$ is written as

$$\Delta_1(x) = \sum_{i=1}^{m} \left(|x_i - \mu_i^-| - |x_i - \mu_i^+| \right),$$

and we observe again, like in the ℓ_2 case, that the contribution to $\Delta_1(x)$ from the ith feature x_i is identically zero when $\mu_i^- = \mu_i^+$.

3. SPARSE ℓ_1 AND ℓ_2 CENTER CLASSIFIERS

In Section 2 we observed that, for both the ℓ_2 and the ℓ_1 distance criteria, the discrimination is insensitive to the ith feature whenever $\theta_i^+ - \theta_i^- = 0$, where θ^+, θ^- are the two class centers. The *sparse* classifiers that we introduce in this section are aimed precisely at computing optimal class centers such that the center difference $\theta^+ - \theta^-$ is k-sparse, meaning that $\|\theta^+ - \theta^-\|_0 \leq k$, where $\|\cdot\|_0$ denotes the number of nonzero entries (i.e., the cardinality) of its argument, and $k \leq m$ is a given cardinality bound. Such type of sparse classifiers will thus perform simultaneous classification and feature selection, by detecting which k out of the total m features are relevant for the classification purposes. We next formally define the sparse ℓ_2 and ℓ_1 center classifier training problems.

Definition 1. (Sparse ℓ_2-center classifier). A sparse ℓ_2-center classifier is a model which classifies an input feature vector $x \in \mathbb{R}^m$ into a positive or a negative class, according to the sign of the discrimination function

$$\begin{aligned} \Delta_2(x) &= \|x - \theta^-\|_2^2 - \|x - \theta^+\|_2^2 \\ &= (\|\theta^-\|_2^2 - \|\theta^+\|_2^2) + 2x^\top (\theta^+ - \theta^-), \end{aligned}$$

where the sparse ℓ_2-centers θ^+, θ^- are learned from a data batch (1) as the optimal solutions of the problem

$$\min_{\theta^+,\theta^-\in\mathbb{R}^m} \frac{1}{n_+}\sum_{j\in\mathcal{J}^+} \|x^{(j)}-\theta^+\|_2^2 + \frac{1}{n_-}\sum_{j\in\mathcal{J}^-} \|x^{(j)}-\theta^-\|_2^2 \quad (7)$$
$$\text{s.t.:} \qquad \|\theta^+-\theta^-\|_0 \le k,$$

where $k \le m$ is a given upper bound on the cardinality of $\theta^+ - \theta^-$.

Definition 2. (Sparse ℓ_1-center classifier). A sparse ℓ_1-center classifier is a model which classifies an input feature vector $x \in \mathbb{R}^m$ into a positive or a negative class, according to the sign of the discrimination function

$$\Delta_1(x) \doteq \|x-\theta^-\|_1 - \|x-\theta^+\|_1,$$

where the sparse ℓ_1-centers θ^+, θ^- are learned from a data batch (1) as the optimal solutions of the problem

$$\min_{\theta^+,\theta^-\in\mathbb{R}^m} \frac{1}{n_+}\sum_{j\in\mathcal{J}^+} \|x^{(j)}-\theta^+\|_1 + \frac{1}{n_-}\sum_{j\in\mathcal{J}^-} \|x^{(j)}-\theta^-\|_1 \quad (8)$$
$$\text{s.t.:} \qquad \|\theta^+-\theta^-\|_0 \le k,$$

where $k \le m$ is a given upper bound on the cardinality of $\theta^+ - \theta^-$.

A perhaps notable fact is that both the sparse ℓ_2 and the sparse ℓ_1 classifier training problems can be solved exactly and with almost-linear-time complexity (this fact is discussed in the next sections), which also makes them good candidates for efficient feature selection methods in two-phase (feature selection + actual classifier training) classifier training procedures.

4. TRAINING THE SPARSE ℓ_2-CENTER CLASSIFIER

Let $\mathcal{E}$ denote a fixed set of indices of cardinality $m-k$, and $\mathcal{D}$ denote the complementary set, that is, $\mathcal{D} = \{1,\dots,m\}\setminus\mathcal{E}$. For any vector $x \in \mathbb{R}^m$ we next use the notation $x_\mathcal{D}$ to denote a vector of the same dimension as x which coincides with x at the locations in $\mathcal{D}$ and it is zero elsewhere. We define analogously $x_\mathcal{E}$, so that $x = x_\mathcal{D} + x_\mathcal{E}$.

The following result characterizes an optimal solution to problem (7); for space reasons a proof of this result is reported in the full version of this paper (Calafiore and Fracastoro, 2019).

Proposition 1. The optimal solution of problem (7) is obtained as follows:

(1) Compute the standard class centroids $\bar{x}^+$, $\bar{x}^-$;
(2) Compute the centroids midpoint $\tilde{x}$
$$\tilde{x} \doteq \frac{\bar{x}^+ + \bar{x}^-}{2}$$
and the centroids difference $\delta \doteq \bar{x}^+ - \bar{x}^-$;
(3) Let $\mathcal{D}$ be the set of the indices of the k largest absolute value elements in vector δ, and let $\mathcal{E}$ be the complementary index set;
(4) The optimal parameters θ^+, θ^- are given by
$$\theta^+ = \bar{x}^+_\mathcal{D} + \tilde{x}_\mathcal{E}$$
$$\theta^- = \bar{x}^-_\mathcal{D} + \tilde{x}_\mathcal{E}.$$

Remark 2. Steps 1-2 in Proposition 1 essentially require computing mn sums. Finding the k largest elements in Step 3 takes $O(m\log k)$ operations (using, e.g., min-heap sorting), whence the whole procedure takes $O(mn) + O(m\log k)$ operations. Thus, while training a plain centroid classifier takes $O(mn)$ operations (which, incidentally, is also the complexity figure for training a classical Naive Bayes classifier), adding exact sparsity comes at the quite moderate extra cost of $O(m\log k)$ operations. ⋆

Remark 3. The sparse ℓ_2 center classifier training procedure is amenable to efficient online implementation, since the class centers are easily updatable as soon as new data comes in. Denote by $\bar{x}(\nu)$ the centroid of one of the two classes when ν observations $\xi^{(1)},\dots,\xi^{(\nu)}$ in that class are present: $\bar{x}(\nu) = \frac{1}{\nu}\sum_{j=1}^{\nu}\xi^{(j)}$. If a new observation $\xi^{(\nu+1)}$ in the same class becomes available, the new centroid will be

$$\bar{x}(\nu+1) = \frac{1}{\nu+1}\sum_{j=1}^{\nu+1}\xi^{(j)} = \frac{1}{\nu+1}\left(\sum_{j=1}^{\nu}\xi^{(j)} + \xi^{(\nu+1)}\right)$$
$$= \frac{\nu}{\nu+1}\bar{x}(\nu) + \frac{1}{\nu+1}\xi^{(\nu+1)}.$$

This latter formula gives the new centroid as a weighted linear combination of the previous centroid and of the new observation. An online version of the procedure in Proposition 1 is thus readily obtained, in which only the current centroids are kept into memory and, as soon as a new datum is available, the corresponding centroid is updated (this takes $O(m)$ operations, or less if the datum is sparse) and the feature ranking is recomputed (this takes $O(m\log k)$ operations). A sparse ℓ_2 center classifier can therefore be trained online with $O(m)$ memory storage and $O(m\log k)$ operations per update. ⋆

Remark 4. (Sparsity-accuracy tradeoff). As it is customary with sparse methods, in practice a whole sequence of training problems is solved at different levels of sparsity, say from $k = 1$ (only one feature selected) to $k = m$ (all features selected), accuracy is evaluated for each model via cross validation, and then the resulting sparsity-accuracy tradeoff curve is examined for the purpose of selection of the most suitable k level. Most feature selection methods, including sparse SVM (Fan et al., 2008), the Lasso (Tibshirani, 1996), and the sparse Naive Bayes method (Askari et al., 2019), require repeatedly solving the training problem for each k, albeit typically warm-starting the optimization procedure with the solution from the previous k value. In the sparse ℓ_2 classifier, instead, one can fully order the vector $|\bar{x}^+ - \bar{x}^-|$ only once, at a computational cost of $O(m\log m)$, and then the optimal solutions are obtained, for any k, by simply selecting in Step 3 of Proposition 1 the first k elements of the ordered vector. ⋆

5. TRAINING THE SPARSE ℓ_1-CENTER CLASSIFIER

We next present a result for efficient and exact solution of the sparse ℓ_1-center classifier training problem. We start by stating a preliminary instrumental result, whose proof is reported in (Calafiore and Fracastoro, 2019) for space resons, and an ensuing definition.

Proposition 2. (Weighted ℓ_1 center). Given a real vector $z = (z_1, z_2, \dots, z_p)$ and a nonnegative vector $w = (w_1,\dots,w_p)$, consider the weighted ℓ_1 centering problem

$$d_w(z) \doteq \min_{\vartheta \in \mathbb{R}} \sum_{i=1}^{p} w_i |z_i - \vartheta|. \tag{9}$$

Let

$$W(\zeta) \doteq \sum_{\{i:\, z_i \le \zeta\}} w_i, \quad \bar{W} \doteq \sum_{i=1}^{p} w_i,$$

and

$$\bar{\zeta} \doteq \inf\{\zeta :\ W(\zeta) \ge \bar{W}/2\}.$$

Then, an optimal solution for problem (9) is given by

$$\vartheta^* = \mathrm{med}_w(z) \doteq \begin{cases} \bar{\zeta} & \text{if } W(\bar{\zeta}) > \dfrac{\bar{W}}{2} \\ \\ \dfrac{1}{2}(\bar{\zeta} + \bar{\zeta}_+) & \text{if } W(\bar{\zeta}) = \dfrac{\bar{W}}{2}, \end{cases} \tag{10}$$

where $\bar{\zeta}_+ \doteq \min\{z_i,\, i = 1, \dots, p : z_i > \bar{\zeta}\}$ is the smallest element in z that is strictly larger than $\bar{\zeta}$. ⋆

Definition 3. (Weighted median and dispersion). Given a row vector z and a nonnegative vector w of the same size, we define as the *weighted median* of z the optimal solution of problem (9) given in (10), and we denote it by $\mathrm{med}_w(z)$. We define as the *weighted median dispersion* the optimal value $d_w(z)$ of problem (9). We extend this notation to matrices, so that for a matrix $X \in \mathbb{R}^{m,n}$ we denote by $\mathrm{med}_w(X) \in \mathbb{R}^m$ a vector whose ith component is $\mathrm{med}_w(X_{i,:})$, where $X_{i,:}$ is the ith row of X, and we denote by $d_w(X) \in \mathbb{R}^m$ the vector of corresponding dispersions. ⋆

We now let $\mathcal{E}$ and $\mathcal{D}$ be defined as in Section 4, and we use the same notation as before for $\theta^{\pm}_{\mathcal{D}}$, $\theta^{\pm}_{\mathcal{E}}$, $x_{\mathcal{D}}$, $x_{\mathcal{E}}$. The following result characterizes an optimal solution to problem (8); for space reasons a proof of this result is reported in the full version of this paper (Calafiore and Fracastoro, 2019).

Proposition 3. The optimal solution of problem (8) is obtained as follows:

(1) Compute the plain class medians

$$\mu^+ \doteq \mathrm{med}(\{x^{(j)}\}_{j \in \mathcal{J}^+})$$
$$\mu^- \doteq \mathrm{med}(\{x^{(j)}\}_{j \in \mathcal{J}^-})$$

and the weighted median of all observations

$$\mu \doteq \mathrm{med}_w(\{x_i^{(j)}\}_{j=1,\dots,n}),$$

where the weight vector w is such that, for $j = 1, \dots, n$, $w_j = 1/n_+$ if $j \in \mathcal{J}^+$, and $w_j = 1/n_-$ if $j \in \mathcal{J}^-$.

(2) Compute the median dispersion vectors d^+, d^-, whose entries, for $i = 1, \dots, m$, are given by

$$d_i^+ \doteq \frac{1}{n_+} \sum_{j \in \mathcal{J}^+} |x_i^{(j)} - \mu_i^+|$$
$$d_i^- \doteq \frac{1}{n_-} \sum_{j \in \mathcal{J}^-} |x_i^{(j)} - \mu_i^-|.$$

Also, compute the weighted median dispersion vector d, whose components are, for $i = 1, \dots, m$,

$$d_i \doteq \sum_{j=1}^{n} w_j |x_i^{(j)} - \mu_i|$$
$$= \frac{1}{n_+} \sum_{j \in \mathcal{J}^+} |x_i^{(j)} - \mu_i| + \frac{1}{n_-} \sum_{j \in \mathcal{J}^-} |x_i^{(j)} - \mu_i|,$$

Table 1. Text dataset sizes

	TWTR	MPQA	SST
Number of features	273779	6208	16599
Number of samples	1600000	10606	79654

and compute the difference vector

$$e \doteq (d^+ + d^-) - d.$$

(3) Let $\mathcal{D}$ be the set of the indices of the k smallest elements in vector e, and let $\mathcal{E}$ be the complementary index set.

(4) The optimal parameters θ^+, θ^- are given by

$$\theta^+ = \mu^+_{\mathcal{D}} + \mu_{\mathcal{E}}$$
$$\theta^- = \mu^-_{\mathcal{D}} + \mu_{\mathcal{E}}.$$

Remark 5. Computation of the medians in Step 1 of Proposition 3 can be performed with in $O(m)$ operations, see, e.g., Blum et al. (1973). Computation of the median dispersions requires $O(mn)$ operations, and finding the k smallest elements in vector e can be performed in $O(m \log k)$ operations, hence the whole procedure in Proposition 3 is performed in $O(mn) + O(m \log k)$ operations. Similar to the case discussed in Remark 4, also in the sparse ℓ_1 center classifier one need to do a full ordering of an m-vector only once in order to obtain all the sparse classifiers for any sparsity level k. ⋆

6. EXPERIMENTS

In this section, we perform an experimental evaluation of the proposed methods, comparing their performance with other feature selection techniques. The sparse ℓ_2-center classifier is tested in the context of sentiment classification on text datasets. This is one of the most common application fields of the nearest centroid classifier. Instead, the sparse ℓ_1-center classifier is evaluated on gene expression datasets. Since this type of data is usually affected by the presence of many outliers, the classifier with the ℓ_1 distance criteria can be preferred over the ℓ_2 version (Hall et al., 2009; Jörnsten, 2004).

6.1 Sparse ℓ_2-center classifier

We compared the proposed sparse ℓ_2-center classifier with other feature selection methods for sentiment classification on text datasets. We considered three different datasets: the TwitterSentiment140 (TWTR) dataset, the MPQA Opinion Corpus Dataset, and the Stanford Sentiment Treebank (SST). Table 1 gives some details on the dataset sizes. Before classification, the dataset are preprocessed rescaling each feature by the inverse of its variance. Each dataset was randomly split in a training (80% of the dataset) and test (20% of the dataset) set. The results reported in this section are an average of 50 different random splits of the dataset.

For each dataset, we performed a two-stage classification procedure. In the first stage, we applied a feature selection method in order to reduce the number of features. Then, in the second stage we trained a classifier model, by employing only the selected features. In order to have a fair comparison, we used the same classifier for all the feature selection methods, namely a linear support vector machine classifier. We compared different feature selection

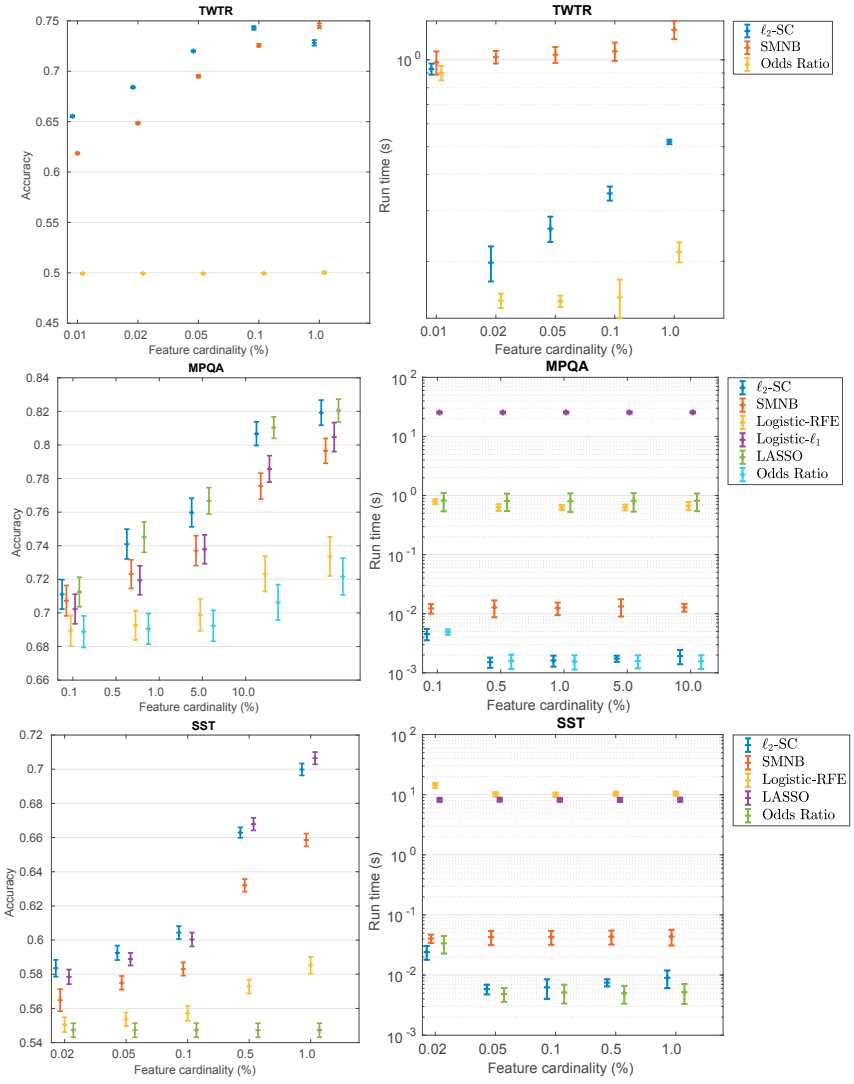

Fig. 1. Classification accuracy and average run time on text datasets.

methods: sparse ℓ_2-centers (ℓ_2-SC), sparse multinomial naive Bayes (SMNB), logistic regression with recursive feature selection (Logistic-RFE), ℓ_1-regularized logistic regression (Logistic-ℓ_1), Lasso, and Odds Ratio. Logistic-RFE, Logistic-ℓ_1 and Lasso are not considered on some datasets, due to their high computational cost that makes them not viable when the dataset size is very large. Fig. 1 shows the accuracy performance and the average run time of the different feature selection methods. These plots show that the sparse ℓ_2-centers is competitive with other feature selection methods in terms of accuracy performance, while its run time is significantly lower than most of the other feature selection methods. The only method that has a comparable computational time is Odds Ratio, but its performance is poor in terms of accuracy.

6.2 Sparse ℓ_1-center classifiers

We compared the proposed sparse ℓ_1-center classifier with other feature selection methods for RNA gene expression classification. We considered three datasets: Chin dataset (Chin et al., 2006), Chowdary dataset (Chowdary et al., 2006), and Singh dataset (Singh et al., 2002). The details of the datasets are summarized in Table 2. As done in the

Table 2. RNA gene expression dataset sizes

	Chowdary (Breast Cancer)	Chin (Breast Cancer)	Singh (Prostate Cancer)
N. features	22283	22215	12600
N. samples	104	118	102

ℓ_2 case, we subdivided each dataset in a training (80% of the dataset) and test (20% of the dataset) set, and we tested 50 random splits.

For each dataset, we performed a two-stage procedure, as explained in the previous section. In the first stage, we compared five feature selection methods: sparse ℓ_1-centers (ℓ_1-SC), ℓ_1-regularized logistic regression (Logistic-ℓ_1), logistic regression with recursive feature elimination (Logistic-RFE), Lasso, and Odds Ratio. Sparse Multinomial Naive Bayes (SMNB) is not taken into account in this experiment since the gene expression datasets can have negative features and SMNB can only be applied to datasets with positive features. In the second stage, we used a linear SVM classifier, as in the previous section. Figure 2 shows the balanced accuracy and average run time of the feature selection methods. Also in this experiment we observe that the proposed method provides an accuracy

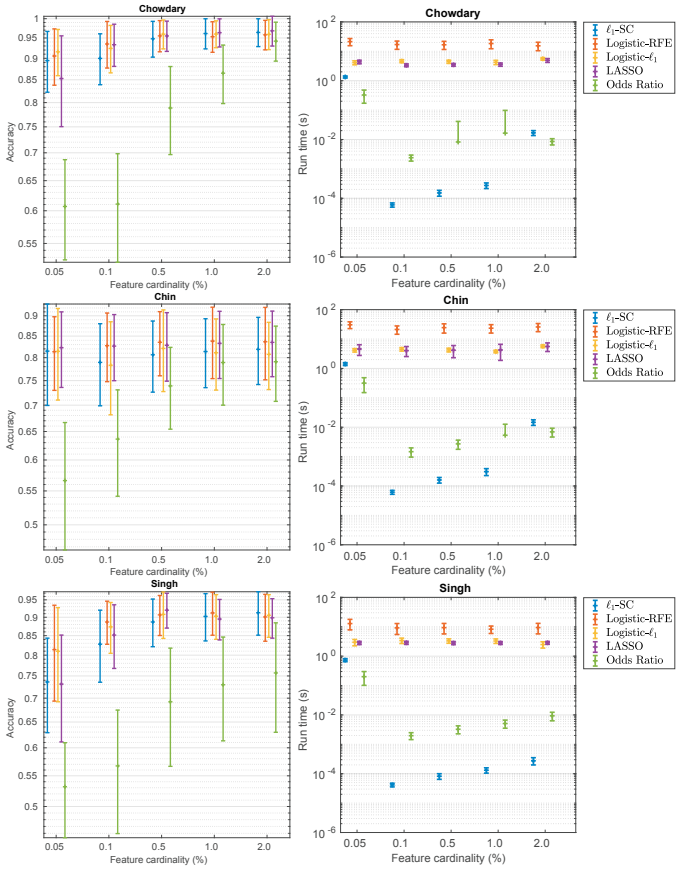

Fig. 2. Classification accuracy and average run time on RNA gene expression datasets.

performance which is similar to that of state-of-the-art techniques, but with a significantly lower computational time.

7. CONCLUSION

In this paper we proposed two types of sparse center classifiers, based respectively on ℓ_1 and the ℓ_2 distance metrics. The proposed methods perform simultaneous classification and feature selection, and in both cases the proposed training method selects the optimal set of features in a quasi-linear computing time. The experimental results also show that the proposed methods achieve accuracy levels that are on par with state-of-the-art feature selection methods, while being substantially faster.

REFERENCES

Askari, A., d'Aspremont, A., and El Ghaoui, L. (2019). Naive feature selection: Sparsity in naive bayes. *arXiv preprint arXiv:1905.09884*.

Blum, M., Floyd, R.W., Pratt, V.R., Rivest, R.L., and Tarjan, R.E. (1973). Time bounds for selection. *J. Comput. Syst. Sci.*, 7(4), 448–461.

Calafiore, G.C. and Fracastoro, G. (2019). Sparse ℓ_1 and ℓ_2 center classifiers. *arXiv preprint arXiv:1911.07320*.

Chin, K., DeVries, S., Fridlyand, J., Spellman, P.T., Roydasgupta, R., Kuo, W.L., Lapuk, A., Neve, R.M., Qian, Z., Ryder, T., et al. (2006). Genomic and transcriptional aberrations linked to breast cancer pathophysiologies. *Cancer cell*, 10(6), 529–541.

Chowdary, D., Lathrop, J., Skelton, J., Curtin, K., Briggs, T., Zhang, Y., Yu, J., Wang, Y., and Mazumder, A. (2006). Prognostic gene expression signatures can be measured in tissues collected in rnalater preservative. *The journal of molecular diagnostics*, 8(1), 31–39.

Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., and Lin, C.J. (2008). Liblinear: A library for large linear classification. *Journal of machine learning research*, 9(Aug), 1871–1874.

Hall, P., Titterington, D., and Xue, J.H. (2009). Median-based classifiers for high-dimensional data. *Journal of the American Statistical Association*, 104(488), 1597–1608.

Han, E.H.S. and Karypis, G. (2000). Centroid-based document classification: Analysis and experimental results. In *European conference on principles of data mining and knowledge discovery*, 424–431. Springer.

Huber, P. (1981). *Robust statistics*. John Wiley & Sons.

Jiang, L., Wang, D., Cai, Z., and Yan, X. (2007). Survey of improving Naive Bayes for classification. In *International Conference on Advanced Data Mining and Applications*, 134–145. Springer.

Jörnsten, R. (2004). Clustering and classification based on the ℓ_1 data depth. *Journal of Multivariate Analysis*, 90(1), 67–89.

Manning, C., Raghavan, P., and Schütze, H. (2008). Vector space classification. *Introduction to Information Retrieval*.

McCallum, A., Nigam, K., et al. (1998). A comparison of event models for Naive Bayes text classification. In *AAAI-98 workshop on learning for text categorization*, volume 752, 41–48. Citeseer.

Mladenic, D. and Grobelnik, M. (1999). Feature selection for unbalanced class distribution and Naive Bayes. In *ICML*, volume 99, 258–267.

Ng, A.Y. (2004). Feature selection, ℓ_1 vs. ℓ_2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning*, 78. ACM.

Rousseeuw, P. and Leroy, A. (1987). *Robust Regression and Outlier Detection*. John Wiley & Sons.

Singh, D., Febbo, P.G., Ross, K., Jackson, D.G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A.A., D'Amico, A.V., Richie, J.P., et al. (2002). Gene expression correlates of clinical prostate cancer behavior. *Cancer cell*, 1(2), 203–209.

Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1), 267–288.

Tibshirani, R., Hastie, T., Narasimhan, B., and Chu, G. (2002). Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proceedings of the National Academy of Sciences*, 99(10), 6567–6572.