A sparse version of the ridge logistic regression for large-scale text categorization

Sujeevan Aseervatham, Anestis Antoniadis, Éric Gaussier, Michel Burlet, Yves Denneulin

A Sparse Version of the Ridge Logistic Regression for Large-Scale Text Categorization

Sujeevan Aseervatham[a,*], Anestis Antoniadis[b], Eric Gaussier[a], Michel Burlet[c], Yves Denneulin[d]

[a] *LIG - Université Joseph Fourier, 385, rue de la Bibliothèque, BP 53, F-38041 Grenoble Cedex 9, France*
[b] *LJK - Université Joseph Fourier, BP 53, F-38041 Grenoble Cedex 9, France*
[c] *Lab. Leibniz - Université Joseph Fourier, 46 Avenue Félix Viallet, F-38031 Grenoble Cedex 1, France*
[d] *LIG - ENSIMAG, 51 avenue Jean Kuntzmann, F-38330 Montbonnot Saint Martin, France*

Abstract

The ridge logistic regression has successfully been used in text categorization problems and it has been shown to reach the same performance as the Support Vector Machine but with the main advantage of computing a probability value rather than a score. However, the dense solution of the ridge makes its use unpractical for large scale categorization. On the other side, LASSO regularization is able to produce sparse solutions but its performance is dominated by the ridge when the number of features is larger than the number of observations and/or when the features are highly correlated. In this paper, we propose a new model selection method which tries to approach the ridge solution by a sparse solution. The method first computes the ridge solution and then performs feature selection. The experimental evaluations show that our method gives a solution which is a good trade-off between the ridge and LASSO solutions.

Keywords: Logistic Regression, Model Selection, Text Categorization, Large Scale Categorization

*Corresponding author, Tel.: +33(0)476514515; Fax: +33(0)476446675
Email addresses: `Sujeevan.Aseervatham@imag.fr` (Sujeevan Aseervatham), `Anestis.Antoniadis@imag.fr` (Anestis Antoniadis), `Eric.Gaussier@imag.fr` (Eric Gaussier), `Michel.Burlet@imag.fr` (Michel Burlet), `Yves.Denneulin@imag.fr` (Yves Denneulin)

1. Introduction

The automatic text categorization problem consists in assigning, according to its content, a textual document to one or more relevant predefined categories. Given a training dataset, where the documents have been manually labeled, the problem lies in inducing a function f, from the training data, which can then be used to classify documents. Machine learning algorithms are used to find the optimal f by solving a minimization problem which can be stated as the minimization of the cost of misclassification over the training dataset (Empirical Risk Minimization).

In order to use numerical machine learning algorithm, the Vector Space Model is commonly used to represent a textual documents by a simple term-frequency vector (Salton et al., 1975). This representation produces datasets in which 1) the number of features is often larger than the number of documents, 2) the vectors are very sparse, i.e., a lot of features are set to zero and 3) the features are highly correlated (due to the nature of natural languages). Moreover, real-life datasets tend to be larger and larger which makes the automatic categorization process complicated and leads to scalability problems. As long as the datasets only grow in terms of the number of observations, the problem can be tackled by distributing the computation over a network of processors (Chu et al., 2006). However, when the number of features becomes larger than the number of observations, machine learning techniques tend to perform poorly due to overfitting, i.e., the model performs well on the training set but poorly on any other set. To prevent overfitting, the complexity of the model must be controlled during the training process, through model selection techniques. In the Support Vector Machine (SVM) algorithm (Vapnik, 1995), the model complexity is given by the VC-dimension, which is the maximum number of vectors, for any combination of labels, that can be shattered by the model. SVMs rely on the Structural Risk Minimization (SRM) principle, which not only aims at minimizing the empirical risk (Empirical Risk Minimization - ERM) but also the VC-dimension. SVMs have been used for text categorization and their performance is among the best ones obtained so far (Joachims, 1998).

The VC-dimension remaining unknown for many functions, the SRM is difficult to implement. Another model selection, widely used, is to minimize both the ERM and a regularization term: $\lambda\Omega[f]$ where λ is a penalty factor, $\Omega[f]$ a convex non-negative regularization term and f the model. For linear functions: $f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b$, the regularization term is often defined as $\Omega[f] = \|\mathbf{w}\|_p$ where $\|\cdot\|_p$ is the L_p-norm (Hoerl and Kennard, 1970; Tibshirani, 1994; Zou and Hastie, 2005). This has the effect of smoothing f and reducing its generalization error. The use of the L_2-norm is known as the ridge penalization, whereas the use of the L_1-norm as the LASSO penalization, which has the property of simultaneously doing shrinkage and feature selection.

In this paper, we focus on penalized logistic regression. Logistic regression has the main advantage of computing a probability value rather than a score, as for the SVM. Furthermore, the ridge logistic regression has been shown to reach the same performance as the SVM on standard text categorization problems (Zhang and Oles, 2001). Nevertheless, it produces a dense solution which cannot be used for large scale categorization. In (Genkin et al., 2007), the LASSO logistic regression was used to obtain a sparse solution. However, when the number of features is larger than the number of observations and/or when the features are correlated, the ridge penalization performance dominates the LASSO one (Zou and Hastie, 2005). Taking into account these observations, we propose a new model selection which produces a sparse solution by approaching the ridge solution.

The rest of the paper is organized as follows: in the next section we discuss related works; we then describe, in section 3, our model selection approach before reporting, in section 4, our experimental results; section 5 concludes the paper.

2. Related work

In (le Cessie and van Houwelingen, 1992), the authors have shown how ridge penalization can be used to improve the logistic regression parameter estimates in the cases where the number of features is larger than the number of observations or when the variables are highly correlated. They have applied ridge logistic regression on DNA data and have obtained good results with stable parameters. More recently, the ridge logistic regression was used in (Zhang and Oles, 2001) on the text categorization problem where the data are sparse and the number of features is larger than the number of observations. The authors have proposed several algorithms, which take advantage of the sparsity of the data, to solve efficiently the ridge optimization problem. The experimental results show that the L_2 logistic regression reaches the same performance as the SVM. Although the ridge method allows to select a more stable model by doing continuous shrinkage, the produced solution is dense and thus not appropriate for large and sparse data such as textual data.

The LASSO regularization (L_1-norm) has been introduced in (Tibshirani, 1994). The author shows, for linear regression, that the L_1 penalization can not only do continuous shrinkage but has also the property of doing automatic variable selection simultaneously which means that the L_1 solution is sparse. In (Genkin et al., 2007), an optimization algorithm based on (Zhang and Oles, 2001) is presented for Ridge and LASSO logistic regressions in the context of text categorization. According to their experiments, the lasso penalization gives slightly better results than the ridge penalization in terms of the macro-averaged-F_1 measure (the micro-averaged results are not given). It has been shown in (Efron et al., 2004; Tibshirani, 1994; Zou and Hastie, 2005) that the performance of the LASSO is dominated by the ridge in the following cases (we denote by p the number of features and by n the number of observations):

- $p > n$: the LASSO will only select at most n features,
- the features are highly correlated: the LASSO will select only one feature among the correlated features.

To tackle the limitations of the LASSO, the Elastic net method has been proposed in (Zou and Hastie, 2005) which tries to capture the best of both L_1 and L_2 penalizations. The Elastic net uses both L_1 and L_2 regularization in the linear regression problem. The authors show that the L_2 regularization term can be reformulated by adding p artificial input data such that each artificial data i has only the i^{th} component non-null set to $\sqrt{\lambda_2}$ where λ_2 is the L_2 regularization hyperparameter. This reformulation, which leads to a LASSO problem, relies on the particular form of the least square term, and cannot be extended to the logistic regression problem. Furthermore, as the L_1 and L_2 regularizations are done simultaneously, it is unclear how the solution of the Elastic net approaches the L_2 solution. In (Zhao and Yu, 2006), the model consistency of LASSO is studied for linear regression and it is shown that the consistency of LASSO depends on the regularization parameter. In (Bach, 2008), the author proves that for a regularization parameter decay factor of $\frac{1}{\sqrt{n}}$, a consistent model can be obtained by applying LASSO on bootstrap samples and by selecting only the intersecting features. Nevertheless, using LASSO on bootstrap samples is a time consuming process. Moreover, since this method is based on LASSO, it also fails to induce a good model when the variables are correlated.

3. Selected Ridge Logistic Regression

The logistic regression model is part of the Generalized Linear Model (GLM) family (Hastie and Tibshirani, 1990; Mccullagh and Nelder, 1989). The GLM is a family of models, parametrized by β, which associate a target variable y to an input data $\mathbf{x}$ ($x \in \mathbb{R}^p$) according to the relation $\beta \cdot \mathbf{x} = g(y)$ where g is a link function and $\beta \in \mathbb{R}^p$. For simplicity, we represent any linear function $\beta' \cdot \mathbf{x}' + \beta'_0$ by $\beta \cdot \mathbf{x}$, where $\mathbf{x}$ is $\mathbf{x}'$ with an extra dimension set to 1, and β is β' with an extra dimension set to β'_0. The logistic regression model is obtained by using a logit function $g(y) = \frac{P(y|\beta,\mathbf{x})}{1-P(y|\beta,\mathbf{x})}$. When $y \in \{-1, 1\}$, the logistic regression model can be written as:

$$P(y = 1|\beta, \mathbf{x}) = \frac{1}{1 + \exp(-\beta \cdot \mathbf{x})} \tag{1}$$

β can be obtained by maximizing the log-likelihood over the training set $\mathcal{D} = \{(\mathbf{x_1}, y_1), \dots, (\mathbf{x_n}, y_n)\}$. However, in order to obtain a strictly convex optimization problem and to avoid overfitting, a Tikhonov regularization term (Hoerl and Kennard, 1970) is added, leading to the following ridge logistic regression problem:

$$\beta^* = \underset{\beta}{\operatorname{argmin}} \underbrace{\sum_{i=1}^{n} \log(1 + \exp(-y_i \beta \cdot \mathbf{x_i})) + \lambda \|\beta\|_2^2}_{l(\beta)} \tag{2}$$

where λ is a strictly positive scalar. Adding a ridge regularization term is equivalent, in a Bayesian framework, to using a Gaussian prior on each component of β, under the assumption that the components are independent, i.e. $P(\beta) = \prod_j P(\beta_j)$ with $P(\beta_j) \sim \mathcal{N}(0, \frac{1}{2\lambda})$.

Several algorithms have been proposed in the literature to solve the optimization problem in 2 (Friedman et al., 2008; Minka, 2003). In (Genkin et al., 2007), an efficient algorithm, based on the one presented in (Zhang and Oles, 2001), is proposed to solve problems with sparse data, such as text documents. However, the ridge regression solution is a dense vector which can hardly be used in large scale categorization where hundreds of thousand features are used. The problem we face is thus the one of finding $\hat{\beta}$ such that:

1. $\hat{\beta}$ is close to β^* and thus behaves well, ie $l(\hat{\beta}) \simeq l(\beta^*)$;
2. $\hat{\beta}$ is a sparse solution and thus can be used on large datasets.

The second order Taylor series expansion on $l(\beta)$ around β^* leads to:

$$\begin{aligned} l(\beta) &\simeq l(\beta^*) + (\beta - \beta^*)^T \nabla l(\beta^*) \\ &\quad + \frac{1}{2}(\beta - \beta^*)^T H_l(\beta^*)(\beta - \beta^*) \\ &= l(\beta^*) + \frac{1}{2}(\beta - \beta^*)^T H_l(\beta^*)(\beta - \beta^*) \end{aligned} \tag{3}$$

where $\nabla l(\beta^*)$ and $H_l(\beta^*)$ are respectively the gradient and the Hessian of $l(\beta)$ at β^* and where the equality derives from the fact that for β^*, the ridge solution, the gradient vanishes. Hence, obtaining a $\hat{\beta}$ yielding a value for l close to the one of β^* while being sparse can be achieved by solving the following strictly convex optimization problem:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} (\beta - \beta^*)^T H_l(\beta^*)(\beta - \beta^*) + \alpha \|\beta\|_1 \tag{4}$$

The L_1 regularization term, used to ensure sparsity, corresponds, in the Bayesian framework, to the Laplace distribution prior on the components of β: $P(\beta_i) \sim \text{Laplace}(0, \frac{1}{\alpha})$ with α a strictly positive scalar. We refer to the above approach as the Selected Ridge Logistic Regression method.

The so-called bag-of-words representation used in most text classification methods assumes independence between words in documents[1]. Such an independence assumption naturally leads to assuming that the components of β are independent of one another, and thus that the Hessian $H_l(\beta^*)$ is diagonal. We make this assumption in the remainder of the paper. In this case, an analytical solution to equation 4 can be obtained. Indeed, equation 4 can be rewritten as:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{p} (\beta_i - \beta_i^*)^2 H_i(\beta^*) + \alpha \|\beta\|_1 \tag{5}$$

with

$$H_i(\beta) = \frac{\partial^2 l(\beta)}{\partial \beta_i^2} = \sum_{j=1}^{n} \frac{{x_{ji}}^2 \exp(-y_j \beta \cdot \mathbf{x_j})}{(1 + \exp(-y_j \beta \cdot \mathbf{x_j}))^2} + 2\lambda \tag{6}$$

Thus, the overall optimization problem can be solved component by component:

$$\hat{\beta}_i = \underset{\beta_i}{\operatorname{argmin}} (\beta_i - \beta_i^*)^2 H_i(\beta^*) + \alpha |\beta_i| \tag{7}$$

and its solution is given by theorem 3.1.

[1](Joachims, 2002) for example recommends to use linear kernels, and not polynomial or Gaussian ones, for text classification.

Theorem 3.1. *The solution, $\hat{\beta}_i$, of the minimization problem in 7 is given by:*

$$\hat{\beta}_i = \begin{cases} \beta_i^* - \frac{\alpha}{2H_i(\beta^*)} & \text{if } \beta_i^* > \frac{\alpha}{2H_i(\beta^*)} \\ \beta_i^* + \frac{\alpha}{2H_i(\beta^*)} & \text{if } \beta_i^* < -\frac{\alpha}{2H_i(\beta^*)} \\ 0 & \text{otherwise} \end{cases}$$

(note that $\hat{\beta}_i = 0$ if $H_i(\beta^*) = 0$).

Proof Let us assume that $\beta_i^* \geq 0$ and let $g(\beta_i) = (\beta_i - \beta_i^*)^2 H_i(\beta^*) + \alpha|\beta_i|$. We have: $\forall \beta_i \geq 0, g(\beta_i) \leq g(-\beta_i)$, so that $\hat{\beta}_i \geq 0$. Setting the derivative of the strictly convex function g wrt β_i to 0, one gets:

$$\begin{aligned} \beta_i^+ &= \underset{\beta_i > 0}{\operatorname{argmin}}\, g(\beta_i) \\ &= \begin{cases} \beta_i^* - \frac{\alpha}{2H_i(\beta^*)} & \text{if } \beta_i^* > \frac{\alpha}{2H_i(\beta^*)} \\ 0 & \text{otherwise.} \end{cases} \end{aligned}$$

In the case where

$$\beta_i^* > \frac{\alpha}{2H_i(\beta^*)}$$

let

$$\beta_i^* = \frac{\alpha}{2H_i(\beta^*)} + \epsilon$$

Then, we have:

$$g(0) = g(\beta_i^* - \frac{\alpha}{2H_i(\beta^*)}) + \epsilon^2 H_i(\beta^*) > g(\beta_i^* - \frac{\alpha}{2H_i(\beta^*)})$$

This shows that $\hat{\beta}_i = \beta_i^+$ when $\beta_i^* \geq 0$. The case $\beta_i^* \leq 0$ is treated in exactly the same way and completes the proof of theorem 3.1.

Automatic Setting of the penalty hyperparameter

In order to reduce the number of hyperparameters to estimate, one can set the LASSO penalty α to the universal penalty (or universal thresholding). Indeed, the function to be minimized in 4 can also be interpreted as the penalized loss of a Gaussian vector β with mean β^* and covariance matrix $H_l^{-1}(\beta^*)$. For $H_l^{-1}(\beta^*)$ bounded in the vicinity of β^*, theorem 4 in (Antoniadis and Fan, 2001) applies and defines the universal penalty (or universal thresholding) to be $\sqrt{\frac{2\log(p)}{p}}$, a value which guarantees that the risk function of $\hat{\beta}$ (solution of 4) is within a factor of logarithmic order. This leads to the following property:

Property 3.1. *The universal penalty α for minimizing 4 w.r.t. β, for $H_l^{-1}(\beta^*)$ bounded in the vicinity of β^*, is $\sqrt{\frac{2\log(p)}{p}}$, with p the dimension of β.*

The algorithm associated with the above, overall approach can be described as follows.

Summary of the approach

The Selected Ridge Logistic Regression method is summarized in algorithm 1.

Algorithm 1 Selected Ridge Logistic Regression

Input: $\mathcal{D}$ - the training dataset

Input: λ - the ridge penalization factor

Input: *optionally* α - the lasso penalization factor

Output: $\hat{\beta}$ - the parameter of the model as defined in eq. 1

```
Compute β* by solving eq. 2
if α is not given as an input argument then
    Use property 3.1 to set α
end if
for all β̂_i of β̂ do
    Use theorem 3.1 to compute β̂_i
end for
```

Despite the fact that the Selected Ridge method involves the computation of a ridge solution, it is important to note, as we will see in the experimental section, that the training time of the Selected Ridge method is usually shorter than that of the Ridge method. This is due to the fact that the optimal λ for both methods are different and, especially in text categorization, the optimal

λ for the Selected Ridge method is larger than the optimal λ for the Ridge method. For a small λ close to zero, the training time of the Ridge method will be important as more iterations will be needed to reach convergence.

Relation to the Fisher Information Matrix

The fisher information matrix $I(\beta)$ is given, for each entry (i, j), by the following equation:

$$I_{i,j}(\beta) = -\mathbb{E}(\frac{\partial^2 \log P(y|\mathbf{x}, \beta)}{\partial \beta_i \partial \beta_j}) \tag{8}$$

Thus, using the empirical Fisher information matrix $\hat{I}(\beta^*)$, we have:

$$H_i(\beta^*) = \hat{I}_{i,i}(\beta^*) + 2\lambda \tag{9}$$

The Fisher information matrix summarizes the average amount of information brought by the data on β. Hence according to theorem 3.1 and formula 9, the more information the data brings on β_i^* (ie the higher $\hat{I}_{i,i}(\beta^*)$), the higher $H_i(\beta^*)$ will be and the closer $\hat{\beta}_i$ will be to β_i^*. In other words, the value obtained through the original ridge regression problem is almost not modified. On the contrary, if the data brings little information on β_i^* (ie $\hat{I}_{i,i}(\beta^*)$ is small), then $H_i(\beta^*)$ will be small and $\hat{\beta}_i$ will be set to zero for a large range of values of β_i^*. Thus, *sparsity is obtained in the Selected Ridge Regression method by setting to 0 the dimensions of the ridge solution β^* which have small values and which are not supported by the data, ie for which $\hat{I}_{i,i}(\beta^*)$ is small.* This result reflects the intuition that, in many text categorization problems, only a few words are crucial and usually correspond to the dimensions for which the ridge values are sufficiently large. The development provided here, in particular theorem 3.1, shows that one should discard dimensions for which the ridge value is not larger than, roughly, the inverse of the Fisher information. Thus, the ridge value is not the only parameter one should consider. The information brought by the data on this value plays indeed a crucial role: dimensions with small values strongly supported by the data should be kept in the final solution.

4. Experimental Results

The proposed model selection method was evaluated over a set of three well-known datasets and one large dataset. The first three datasets are Reuters 21578, Ohsumed and 20-NewsGroups (Hersh et al., 1994; Joachims, 1998, 2002). All of these datasets have been widely studied in the text categorization literature. Reuters 21578[2] is a collection of news on different domains. Ohsumed[3] is a collection of medical abstracts originally designed for content-based information retrieval, and 20-NewsGroup[4] a collection of documents written in the context of 20 different news groups. These collections are thus varied in terms of their production and content. The last dataset is a subset of documents taken from the DMOZ website[5]. This DMOZ dataset was collected in order to perform Large-Scale text categorization experiments. The characteristics of the datasets are reported in table 1. This last collection is a collection of web pages, and contains documents of various types (scientific articles, business descriptions, ...) on several domains.

Table 1: Datasets used for the experiments

Name	**Train. size (n)**	**Test size**	**#Features (p)**	**#Categories**	**Case**
Reuters-21578	7770	3019	6760	90	$p < n$ ($\frac{p}{n} \approx 1$)
Ohsumed	6286	7643	20520	23	$p > n$ ($\frac{p}{n} \approx 3$)
20-NewsGroups	12492	6246	51666	20	$p > n$ ($\frac{p}{n} \approx 4$)
DMOZ	20249	7257	133348	3503	$p > n$ ($\frac{p}{n} \approx 6$)

All the datasets have been pre-processed according to the setting defined in (Joachims, 2002), which we briefly describe here. The Vector Space Model (VSM) (Salton et al., 1975) is used to represent the textual documents in a vector

[2]http://www.daviddlewis.com/resources/testcollections/reuters21578/
[3]http://ir.ohsu.edu/ohsumed/ohsumed.html
[4]http://people.csail.mit.edu/jrennie/20Newsgroups/
[5]http://www.dmoz.org

space. The VSM is also known as the *Bag-of-Words* (BOW) representation in which a list of terms is used to define a vector space, each term defining an axis of the space. A textual document can then be represented as a vector, using for each axis, the corresponding term frequency value. In order to obtain an efficient vector representation, each document is pre-processed using the following steps:

1. Cleaning by removing non-Latin characters, numerical symbols and punctuation marks,
2. Segmenting terms separated by a white space into a list of words,
3. Removing stopwords (using a stopword list),
4. Stemming each word using the Porter Stemming algorithm (Porter, 1980).

We also used the TF-IDF weighting scheme (Jones, 1988) to give more importance to terms that are frequent in a document (the TF part) and specific to a small number of documents (IDF part). Furthermore, we normalized all document vectors.

For multi-class categorization, we use the one-vs-the-rest strategy based on binary logistic regression models. To assign a document to a unique category in mono-label problems, we use the following decision function: $\text{argmax}_c\, P(y_i = +1|\beta_c, \mathbf{x_i})$; if a document can be assigned to several categories (multi-label problems), we assign it to each category c such that $P(y_i = +1|\beta_c, \mathbf{x_i}) \geq 0.5$.

In the experiments, the F_1 measure (van Rijsbergen, 1979) is used to evaluate the performance of the classifiers. It is defined as:

$$F_1 = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FP} + \text{FN}} \tag{10}$$

where TP stands for true positive, FP for false positive and FN for false negative. For multi-class datasets, we used the micro-F_1 and macro-F_1 measures. In the micro-F_1 measure, TP, FP and FN are summed over each category giving thus an equal weight to each document. In this case, this measure corresponds to the overall precision of the system and provides a measure of the accuracy of the classifier. The macro-F_1 is the arithmetic mean of F_1 across the categories, giving an equal weight to each category. If F_1^c denotes the F_1 measure for category c, then the micro-F_1 is defined by:

$$\text{micro-F}_1 = \sum_{c=1}^{K} \frac{N_c}{N} F_1^c \tag{11}$$

where K denotes the number of categories, N_c corresponds to the number of documents in category c, and N is the total number of documents ($N = \sum_c N_c$). The macro-F_1 is defined by:

$$\text{macro-F}_1 = \sum_{c=1}^{K} \frac{1}{K} F_1^c \tag{12}$$

We also report the degree of sparsity for each model. The sparsity is given by :

$$s = 1 - \frac{\text{avg \#features used in the model}}{\text{\#features in the dataset}} \tag{13}$$

A solution based on all the features will thus have a degree of sparsity of 0.

Moreover, it is important to note that the penalization parameter was fixed for each algorithm by cross-validation except for the DMOZ subset where the parameter was tuned using a validation set composed of 7256 documents. For the Selected Ridge method, the hyperparameter α in equation 4 was automatically set using property 3.1.

To solve the LASSO and Ridge regression problems, we used the algorithm described in (Genkin et al., 2007). The training and prediction times are given as indications. Since, the calculations were distributed over a set of computers, the given times are the times spent on calculation plus the times consumed by the system (thread swapping, network transfer time, etc.).

It is also important to note that in all the results below, the training time of the Selected Ridge method is always shorter than that of the Ridge method. This can be confusing since the Selected Ridge involves the computation of a Ridge solution and, thus, one can expect its training time to be at least equal to that of the Ridge method. Actually, as we said above, this is due to the fact that the ridge's training time depends on the Ridge regularization parameter λ. If λ is very close to 0 then the training time will be important as more iterations will be needed to reach convergence. In all our experiments, the optimal λ for the Ridge method was always smaller than that for the Selected Ridge method. For example, for the DMOZ dataset in subsection 4.4, the optimal λ for Ridge is 0.0001 (training time: 13299.43s) but the optimal λ for the Selected Ridge is 0.001 (training time: 10996.80s). This difference can also be seen in table 6: when the L1-parameter (α) is zero (Selected Ridge=ridge) then the optimal λ for the Selected Ridge method is 0.0001, but when α is greater than 0 then the optimal λ is always 0.001.

4.1. Experiments on Reuters-21587

The Reuters-21587 dataset is a collection of newswire articles. Each document was manually assigned to one or more categories, according to its subject. In this collection, we used the standard "ModApte" split, which provides training and test sets. The results are reported in Table 2. The LASSO and the ridge reach the same level of performance; however, the ridge method yields a dense model whereas the LASSO one only selects 0.0043% of the features. The feature selection method used on the ridge model (Selected Ridge Regression method) allows to reach the same micro-F_1 performance than the ridge method, but with a number of features reduced by 95%.

Table 2: Categorization Result on Reuters-21587(ModApte)

Algorithm	**Micro F_1**	**Macro F_1**	**Sparsity**	**Training time (sec)**	**Pred. time (sec)**
LASSO	0.8711	0.5167	0.9957	164.07	0.44
Ridge	0.8690	0.5099	0.0	257.96	13.20
Selected Ridge	0.8645	0.4563	0.9447	180.24	1.55

4.2. Experiments on Ohsumed

The Ohsumed corpus (Hersh et al., 1994) is a subset of the medical bibliographic database MEDLINE. Each document is a reference of a medical article published in a medical journal. Following the settings defined in (Joachims, 1998, 2002), we only kept the first 20,000 references which had abstracts and were published in 1991. This set is split into a training set composed of the first 10,000 documents and a test set composed of the rest. Only abstracts are used for the categorization task. After the pre-processing, the training set is reduced to 6,286 unique documents and the test set to 7,643. Each document belongs to one or more cardiovascular categories.

As shown in table 3, the LASSO method performs well on this dataset ; not only it has the best performance in terms of micro and macro-F_1 but it also gives a very sparse solution. The Selected Ridge method slightly improves the micro-F_1 performance of the ridge method while removing 88% of its features.

Table 3: Categorization Result on Ohsumed

Algorithm	**Micro F_1**	**Macro F_1**	**Sparsity**	**Training time (sec)**	**Pred. time (sec)**
LASSO	0.6533	0.6053	0.9800	81.16	1.83
Ridge	0.6387	0.5897	0.0	144.06	31.20
Selected Ridge	0.6409	0.5802	0.8827	107.08	5.32

4.3. Experiments on 20-Newsgroups

The 20-NewsGroups is a collection of emails taken from the Usenet newsgroups. Each email is assigned to a unique category according to its topic. The experiment on 20-newsgroups, reported in table 4, clearly shows that the ridge penalization outperforms the LASSO method. In fact, the variable selection of the LASSO is too aggressive and eliminates interesting features. However, our variable selection method (Selected Ridge) achieves micro-F1 and macro-F1 scores similar to those obtained by the ridge, while relying on only 10% of the features used in the ridge solution.

Table 4: Categorization Result on 20-NewsGroups

Algorithm	Micro F_1	Macro F_1	Sparsity	Training time (sec)	Pred. time (sec)
LASSO	0.8663	0.8644	0.9861	384.16	1.72
Ridge	0.9038	0.9018	0.0	157.96	71.25
Selected Ridge	0.8966	0.8939	0.9050	136.01	7.51

4.4. Experiments on DMOZ

In order to assess the behavior of the different methods in a large scale categorization setting, we have collected 34,762 html documents from the DMOZ website. DMOZ (www.dmoz.org) is an open directory project that aims to classify the whole web into categories. In the collected dataset, we only used 3,503 categories and we split the corpus into 3 parts: a training set composed of 20,249 documents, a validation set composed of 7,256 documents and a test set composed of 7,257 documents. The validation set is used to tune the hyperparameters. For the pre-processing of the documents, we removed the html tags and the script parts to keep only the text and we applied the standard pre-processing steps described above. For illustration, figure 1 shows a part of a document from the corpus before and after the pre-processing. In this dataset, each document belongs to a unique category.

As expected in the case where the number of features is largely greater than the number of documents, the ridge method clearly outperforms LASSO as shown in table 5. However, the ridge solution being dense, the categorization of large sets is a time consuming process which makes the ridge solution inappropriate. The LASSO and the Selected Ridge methods both produce a sparse

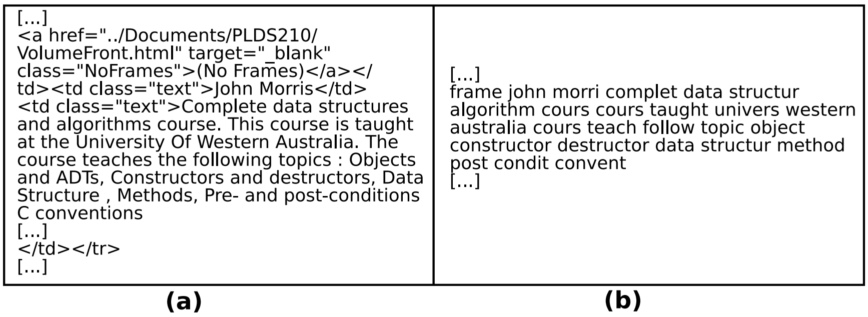

Figure 1: A part of an html document taken from the DMOZ corpus (a) before and (b) after the pre-processing. The address of the document is *http://www.oopweb.com/Algorithms/Files/Algorithms.html* and it is referenced in DMOZ as "OOPWeb Algorithms Directory" at *http://www.dmoz.org/Computers/Algorithms*.

solution with a degree of sparsity of 99%. The selected ridge performs better than the LASSO in terms of the micro-F_1 measure, but however has a macro-F_1 value slightly lower than the value obtained by LASSO.

Table 5: Categorization Result on DMOZ

Algorithm	**Micro F_1**	**Macro F_1**	**Sparsity**	**Training time (sec)**	**Pred. time (sec)**
LASSO	0.2936	0.1661	0.9999	9805.78	41.51
Ridge	0.3434	0.2020	0.0	13299.40	31084.90
Selected Ridge	0.3124	0.1586	0.9993	10996.80	42.52

In table 6, we report the performance of the Selected Ridge method according to the value of the L_1 penalty term in equation 4. The results show that property 3.1 provides a good penalty value in terms of trade-off between micro-F_1 performance and sparsity. It is also interesting to note that with α set to 10^{-7}, one obtains a method yielding results on a par with the ones obtained by the ridge (which provides the best results in terms of both micro- and macro-F_1) while being twice sparser and almost four times faster.

Table 6: Performance of the Selected Ridge Method on DMOZ according to the penalty value in equation 4. The results corresponding to the optimal universal penalty value (property 3.1) are indicated in bold.

Penalty (α)	**Micro-F_1**	**Macro-F_1**	**Sparsity**	**Training time (sec)**	**Prediction time (sec)**
1	0.1188	0.0353	0.9999	11103.9	8.33
0.1	0.2604	0.1209	0.9999	11065.8	32.81
0.05	0.2835	0.1343	0.9998	11325.1	39.82
0.03	0.2953	0.1436	0.9997	11013.4	39.34
0.02	0.3040	0.1524	0.9996	11164.6	49.07
0.0133	**0.3124**	**0.1586**	**0.9993**	**10996.8**	**42.52**
0.01	0.3156	0.1604	0.9992	10965.5	52.97
10^{-7}	0.3434	0.1949	0.5423	11090.2	8858.31
0	0.3434	0.2020	0.0	13299.40	31084.90

5. Conclusion

As pointed in (Zhao and Yu, 2006): *Sparsity or parsimony of statistical models is crucial for their proper interpretations.* In this paper, we have proposed a model selection method to "sparsify" the ridge logistic regression solution. This method first solves the classic ridge logistic regression, then sets less informative features with low values to zero, while ensuring that the resulting sparse solution remains in the vicinity of the ridge solution. This latter property is obtained by using a Taylor expansion of the likelihood function around the solution of the ridge, penalized with the L_1 norm. The experimental text categorization results obtained on well-studied datasets and on a large-scale dataset collected from *www.dmoz.org* show that our method produces a solution which offers a good trade-off between the performance of the ridge solution and the sparsity of the LASSO solution. In particular, when $p > n$ (the number of features is greater than the number of observations), our method leads to a sparse version of the ridge which is both accurate (in terms of both micro- and macro-F_1) and fast.

Acknowledgements

This work was partly supported by the LASCAR project, Univ. J. Fourier, Grenoble.

Antoniadis, A., Fan, J., 2001. Regularization of Wavelet Approximations. Journal of the American Statistical Association 96, 939–967.

Bach, F. R., 2008. Bolasso: model consistent lasso estimation through the bootstrap. In: ICML '08: Proceedings of the 25th international conference on Machine learning. ACM, New York, NY, USA, pp. 33–40.

Chu, C.-T., Kim, S. K., Lin, Y.-A., Yu, Y., Ng, G. B. A. Y., Olukotun, K., 2006. Map-Reduce for Machine Learning on Multicore. In: Schölkopf, B., Platt, J. C., Hoffman, T. (Eds.), NIPS. MIT Press, pp. 281–288.

Efron, B., Hastie, T., Johnstone, L., Tibshirani, R., 2004. Least angle regression. Annals of Statistics 32, 407–499.

Friedman, J., Hastie, T., Tibshirani, R., 2008. Regularization paths for generalized linear models via coordinate descent. Tech. rep., Dept. of Statistics, Stanford University.

Genkin, A., Lewis, D. D., Madigan, D., 2007. Large-Scale Bayesian Logistic Regression for Text Categorization. Technometrics 49, 291–304(14).

Hastie, T. J., Tibshirani, R. J., 1990. Generalized Additive Models. Chapman & Hall.

Hersh, W., Buckley, C., Leone, T. J., Hickam, D., 1994. OHSUMED: an interactive retrieval evaluation and new large test collection for research. In:

Proceedings of the 17th annual international ACM SIGIR. Springer-Verlag New York, Inc., New York, NY, USA, pp. 192–201.

Hoerl, A. E., Kennard, R. W., 1970. Ridge Regression: Biased Estimation for Nonorthogonal Problems. Technometrics 12 (1), 55–67.

Joachims, T., 1998. Text categorization with support vector machines: learning with many relevant features. In: Proceedings of the 10th European Conference on Machine Learning (ECML). Springer Verlag, Heidelberg, DE, pp. 137–142.

Joachims, T., 2002. Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms. Kluwer Academic Publishers, Norwell, MA, USA.

Jones, K. S., 1988. A statistical interpretation of term specificity and its application in retrieval. Document retrieval systems, 132–142.

le Cessie, S., van Houwelingen, J. C., 1992. Ridge Estimators in Logistic Regression. Applied Statistics 41 (1), 191–201.

Mccullagh, P., Nelder, J. A., 1989. Generalized Linear Models. Chapman & Hall, London, UK.

Minka, T. P., 2003. A Comparison of Numerical Optimizers for Logistic Regression. Tech. rep., Dept. of Statistics, Carnegie Mellon University.

Porter, M. F., July 1980. An algorithm for suffix stripping. Program 14 (3).

Salton, G., Wong, A., Yang, C. S., 1975. A Vector Space Model for automatic indexing. Communications of the ACM 18 (11), 613–620.

Tibshirani, R., 1994. Regression Shrinkage and Selection Via the Lasso. Journal of the Royal Statistical Society, Series B 58, 267–288.

van Rijsbergen, C. J., 1979. Information retrieval, 2nd Edition. Butterworths, London, UK.

Vapnik, V. N., 1995. The nature of statistical learning theory. Springer-Verlag New York, Inc.

Zhang, T., Oles, F. J., 2001. Text Categorization Based on Regularized Linear Classification Methods. Information Retrieval 4, 5–31.

Zhao, P., Yu, B., 2006. On Model Selection Consistency of Lasso. The Journal of Machine Learning Research 7, 2541–2563.

Zou, H., Hastie, T., 2005. Regularization and variable selection via the elastic net. Journal Of The Royal Statistical Society Series B 67 (2), 301–320.