



Universidad Autónoma  
de Madrid

**Biblos-e Archivo**  
Repositorio Institucional UAM

**Repositorio Institucional de la Universidad Autónoma de Madrid**

<https://repositorio.uam.es>

Esta es la **versión de autor** del artículo publicado en:  
This is an **author produced version** of a paper published in:

The International Joint Conference on Neural Networks  
(IJCNN), Brisbane, 2012

**DOI:** <https://doi.org/10.1109/IJCNN.2012.6252843>

**Copyright:** © 2012 IEEE.

El acceso a la versión del editor puede requerir la suscripción del recurso  
Access to the published version may require subscription

# Sparse Methods for Wind Energy Prediction

Carlos M. Alaíz, Álvaro Barbero, José R. Dorronsoro

Departamento de Ingeniería Informática & Instituto de Ingeniería del Conocimiento,

Universidad Autónoma de Madrid, 28049 Madrid, Spain

{carlos.alaiz, alvaro.barbero, jose.dorronsoro}@uam.es

**Abstract**—In this work we will analyze and apply to the prediction of wind energy some of the best known regularized linear regression algorithms, such as Ordinary Least Squares, Ridge Regression and, particularly, Lasso, Group Lasso and Elastic-Net that also seek to impose a certain degree of sparseness on the final models. To achieve this goal, some of them introduce a non-differentiable regularization term that requires special techniques to solve the corresponding optimization problem that will yield the final model. Proximal Algorithms have been recently introduced precisely to handle this kind of optimization problems, and so we will briefly review how to apply them in regularized linear regression. Moreover, the proximal method FISTA will be used when applying the non-differentiable models to the problem of predicting the global wind energy production in Spain, using as inputs numerical weather forecasts for the entire Iberian peninsula. Our results show how some of the studied sparsity-inducing models are able to produce a coherent selection of features, attaining similar performance to a baseline model using expert information, while making use of less data features.

## I. INTRODUCTION

Weather related research is receiving nowadays a growing degree of attention. Climate change is an obvious topic of research (and concern); another important issue is the management of renewable energies, particularly those such as wind and solar energy that are not easily stored. This difficulty can only be compensated by adequate planning which, in turn, requires accurate enough forecasting methods. The main forecasting tool are numerical weather prediction (NWP) systems, such as those provided by the European Center for Medium-Range Weather Forecast (ECMWF; [1]) or the Global Forecasting System (GFS; [2]). But it is also known that machine learning (ML) methods can be very useful to improve on the base forecasts given by such systems. One area of interest is the application of ML models to transform NWP forecasts into actual energy production forecasts [3]. However, the large dimensionality of NWP predictions makes mandatory to precede ML model building with either dimensionality reduction techniques or, alternatively, the use of sparsity-inducing models.

One such problem is addressed in this work: globally predicting the wind energy production over a whole country (namely, Spain) which is among the world leaders in both absolute and relative wind energy penetration. This high penetration level makes it critical to provide accurate predictions of wind energy, both to meet the daily energy market requirements and also to enable the electricity system operator to efficiently and reliably manage the system.

A first approach to this problem would be to predict the individual output of each individual wind farm and then aggregate the estimations. This is a quite sensible approach, which allows to obtain an aggregate global prediction more accurate than that of individual farms, particularly if the individual farm outputs are fairly uncorrelated (something that is not always true, as wind farms are clustered about those regions with larger wind readings). However this approach implies building a model for each of the wind farms in the country (above 600 in Spain), which in turn would require very specific and detailed energy production information from each of them, as well as a high degree of information synchronization.

The alternative considered in this work is the prediction of a single global wind energy value. The usual approach for this is to use historical wind energy production data and NWP to build models that will be able to predict wind energy from NWP forecasts for days to follow. One important issue is then the handling of the very large dimensionality of NWP forecasts. In the case of the ECMWF forecasts for Spain, data points are provided in the form of a lattice with a spatial resolution of 0.25 degrees, which results in a grid of  $35 \times 57 = 1995$  nodes for the Iberian peninsula. Furthermore, a number of predicted meteorological variables are provided for each of those points, also possibly including information at different pressure layers. In this paper we shall work with a coarser  $0.5^\circ$  spatial resolution over a  $18 \times 29 = 522$  node grid, including five meteorological variables per node and just a single layer of surface forecasts. Still, this reduced data set results in NWP patterns with 2,610 features. This has to be compared to the number of samples available: ECMWF forecasts are given at three-hour intervals and a whole year of data is used as training set for this study. The number of available patterns is then  $8 \times 365 = 2,920$ , that is, of the same order of magnitude than pattern dimension, and well below the standard rule of thumb for linear regression of having at least 10 patterns per free parameter.

One effective way to alleviate this problem is by using some of the prior knowledge about the problem, i.e. that the actual wind productions are mainly influenced by those NWP features corresponding to data points close to farms themselves. However, the spatial location of all of the wind farms within a country is not always readily available, and surprisingly this information can vary rather fast: Spain has essentially doubled its wind energy output in the last 5 years. A more practical alternative is the use of automated methods to pick those grid nodes and/or features most useful for global wind

power prediction, discarding the rest. Therefore, a sensible approach is to use NWP data over all the considered area (in our case, an enlarged region around the Iberian peninsula) but to work with sparsity-inducing models. We shall consider here regression models including an sparsity-enforcing term, such as Lasso [4], Group Lasso [5] and Elastic-Net [6], that were developed by the Stanford school of Breiman, Friedman, Hastie, Tibshirani *et al.* These methods have behind them both a strong theoretical foundation as well as a suite of algorithms for solving their associated convex optimization problems. We shall however consider them from an alternative algorithmic viewpoint, that of proximal optimization [7], available in principle for a wider range of problems, and for which there is a growing number of algorithms and publicly available code.

The paper is organized as follows. Section II contains a revision of some of the state-of-the-art sparse and regularized linear models of interest for this work. We shall first review what we could call the standard approach to them and then reconsider some of these models under a proximal optimization viewpoint. In Section III all these models are applied to the particular problem of wind energy prediction described before and finally, in Section IV, a discussion is made and the conclusions of the work are presented.

## II. SPARSE LINEAR MODELS

Let  $x^n \in \mathbb{R}^M$ , with  $n = 1, \dots, N$ , be the set of input patterns, and let  $y^n \in \mathbb{R}$ ,  $n = 1, \dots, N$ , be the corresponding desired outputs. The goal of any linear regression method is to determine a vector of weights  $w$  such that  $x^n \cdot w \simeq y^n$ ,  $n = 1, \dots, N$ ; or, equivalently with matrix notation,  $Xw \simeq y$ , where  $X$  is the  $N \times M$  matrix collecting the inputs  $x_n$  as its rows, and  $y$  is the vector formed by the desired outputs. In what follows some widely-known techniques for solving this problem are considered.

### A. Sparse Linear Regression

The simplest approach to this problem is to compute  $w$  directly by minimizing the quadratic error

$$\min_w \frac{1}{2N} \|Xw - y\|_2^2.$$

Computing the gradient and making it equal to 0, we obtain

$$w^* = (X^T X)^{-1} X^T y.$$

This is just the Ordinary Least Squares (OLS) algorithm that relies on the pseudo-inverse of the sample matrix  $X$ . It is well known that just by itself, OLS is very prone to overfitting, particularly in problems where sample size  $N$  and feature dimension  $M$  have similar magnitude. For this reason, some kind of regularization is needed to guarantee a good generalization.

In this line, Regularized Least Squares algorithm (RLS, also known as Ridge Regression [8]) is a natural first option. RLS penalizes also the complexity of the model by adding a quadratic term  $\|w\|_2^2$ , so that the error function turns out to be

$$\min_w \frac{1}{2N} \|Xw - y\|_2^2 + \frac{\lambda_2}{2M} \|w\|_2^2.$$

The parameter  $\lambda_2$  determines the importance of the error versus the complexity of the model. We normalize the regularization term for RLS and all the following algorithms to automatically compensate for variations of the number of patterns  $N$  or of feature dimension  $M$ . The analytical solution to this problem is now

$$w^* = (X^T X + \frac{\lambda_2 N}{M} I)^{-1} X^T y,$$

where  $I \in \mathbb{R}^{M \times M}$  is the identity matrix.

Although RLS usually handles well the problem of overfitting, it does not produce a sparse solution, something which can be desirable when working with a large number of features. This is the context in which the Lasso algorithm (LA) arises [4]. This algorithm penalizes the complexity using an  $\ell_1$  norm, instead of the  $\ell_2$  norm of RLS, and therefore it minimizes the expression

$$\min_w \frac{1}{2N} \|Xw - y\|_2^2 + \frac{\lambda_1}{M} \|w\|_1.$$

Due to the non-differentiability of the term  $\|w\|_1$ , this problem cannot be solved analytically. The standard method for solving LA is the Least Angle Regression method (LARS) [9].

It should be noticed that LA promotes sparsity over all features indistinctly. Many problems of interest present some kind of group structure in their features. As it will be described in detail in Section III, wind power prediction is such a problem, as features are naturally grouped by their values at each individual grid point. In this case, a desirable property is that all the variables corresponding to the same point are active or inactive at the same time. This can be applied, for instance, to analyze which points are actually relevant for the particular task, something of obvious interest in wind power prediction. Trying to solve this, the Group Lasso (GL) algorithm [5] uses the following error function:

$$\min_w \frac{1}{2N} \|Xw - y\|_2^2 + \frac{\lambda_1}{M} \|w\|_{2,1},$$

where the mixed  $\ell_{2,1}$  norm is defined as

$$\|x\|_{2,1} = \sum_{i=1}^{\frac{M}{V}} \sqrt{\sum_{v=1}^V x_{i,v}^2},$$

with  $M$  is the total number of features, divided in  $\frac{M}{V}$  groups of  $V$  variables each group. The first subscript  $i$  denotes the index group, and the second subscript  $v$  the individual variables inside a group. In other words, the mixed norm is the  $\ell_1$  norm of the  $\ell_2$  norms of the variable groups. This means that it produces sparsity over the norms of the groups, i.e., only some groups are active in the sense that for a particular group either all variables or none are taken. Because the norm used inside the groups is the  $\ell_2$  norm, there will not be a second level of intra-group sparsity for the active groups. Again, the objective function of Group Lasso is non-differentiable, and a particular algorithm to solve it is proposed in [5].

Additionally, depending on the nature of the particular regression problem at hand, the performance of either LA or

TABLE I  
LINEAR REGRESSION PROBLEMS.

Problem	Criterion Function
OLS	$\min_w \frac{1}{2N} \ Xw - y\ _2^2$
RLS	$\min_w \frac{1}{2N} \ Xw - y\ _2^2 + \frac{\lambda_2}{2M} \ w\ _2^2$
LA	$\min_w \frac{1}{2N} \ Xw - y\ _2^2 + \frac{\lambda_1}{M} \ w\ _1$
GL	$\min_w \frac{1}{2N} \ Xw - y\ _2^2 + \frac{\lambda_1}{M} \ w\ _{2,1}$
ENet	$\min_w \frac{1}{2N} \ Xw - y\ _2^2 + \frac{\lambda_1}{M} \ w\ _1 + \frac{\lambda_2}{2M} \ w\ _2^2$

RLS will dominate. Also, both models feature their own pros and cons, and so, as a way to join their strengths, both  $\ell_1$  and  $\ell_2$  regularizers can be employed to get the optimization problem:

$$\min_w \frac{1}{2N} \|Xw - y\|_2^2 + \frac{\lambda_1}{M} \|w\|_1 + \frac{\lambda_2}{2M} \|w\|_2^2.$$

This becomes then the Elastic-Net (ENet) approach [6]. To solve it, observe that just as RLS involves in its solution a regularized covariance matrix, in [6] the features are expanded so that the ENet problem can be transformed into a standard Lasso one. All the previous algorithms are summarized in Table I.

Individual algorithms have been introduced to solve all the previous non-differentiable regularization problems. However, we shall briefly discuss next in Subsection II-B how to place all of them under a general proximal optimization setting and to solve them under the common framework of the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [10]. While we shall not discuss it here, this approach also opens the way to enforce sparsity working with more general and possibly stronger models than an  $\ell_1$  penalty term that, on the other hand, may still result in corresponding convex optimization problems solvable by FISTA-like methods.

### B. Proximal Algorithms

As just mentioned, a framework that unifies a variety of non-differentiable optimization problems are the Proximal Methods (PM) [7], [11], which are briefly introduced next.

First of all, we recall the concept of proximity operator. Let  $\Gamma_0(\mathbb{R}^M)$  be the set of lower semi-continuous convex functions from  $\mathbb{R}^M$  to  $\mathbb{R}$  with non-empty domains. Note that functions in  $\Gamma_0(\mathbb{R}^M)$  are not required to be differentiable. For a function  $f \in \Gamma_0(\mathbb{R}^M)$ , its proximity operator at a point  $x \in \mathbb{R}^M$  with step  $\lambda$  is defined as

$$\text{prox}_{\lambda;f}(x) = \arg \min_y \left\{ \lambda f(y) + \frac{1}{2} \|x - y\|_2^2 \right\}.$$

As explained below, this operator can be seen as somehow reminiscent of gradient descent step of standard differentiable optimization in the sense that it “moves” the point so as to minimize the function  $f$  while trying to remain as close as possible to the initial point. Another, more precise, interpretation of this operator is as a generalization of the projection onto a convex set. In fact, if  $f$  is the indicator function of such

a set, then  $\text{prox}_{\lambda;f}(\cdot)$  becomes just the euclidean projection for any  $\lambda \in \mathbb{R}$ .

The PM algorithms are used to minimize sums of certain functions  $f_i$ ,  $i = 1, \dots, m$ . The underlying idea is to minimize alternatively over each of the functions until convergence to the global optimal solution. This splitting of the problem allows to easily include non-differentiable functions in the summation.

The problem considered here is the special case of the sum of a differentiable function and another in  $\Gamma_0(\mathbb{R}^M)$ . We state it formally as follows [7]:

**Problem.** Let  $f_1 : \mathbb{R}^M \rightarrow \mathbb{R}$ ,  $f_1 \in \Gamma_0(\mathbb{R}^M)$ , and let  $f_2 : \mathbb{R}^M \rightarrow \mathbb{R}$  a convex function with a  $\beta$ -Lipschitz continuous gradient  $\nabla f_2$ . Assume that  $f_1(x) + f_2(x) \xrightarrow{\|x\| \rightarrow +\infty} +\infty$ . The minimization problem is stated as:

$$\min_{x \in \mathbb{R}^M} f_1(x) + f_2(x).$$

This problems admits one solution  $x^*$  which satisfies the fixed point condition:

$$x^* = \text{prox}_{\gamma;f_1}(x^* - \gamma \nabla f_2(x^*)) ,$$

for every  $\gamma > 0$  [7]. So a first approach is just to iterate the expression

$$x_{k+1} = \text{prox}_{\gamma_k;f_1}(x_k - \gamma_k \nabla f_2(x_k)) .$$

This is sometimes called a forward-backward split step, as the application of gradient descent is considered the forward step and that of the proximity operator the backward one. For the special case in which  $f_1$  is the indicator function of a convex set  $C$ , the forward step tries to minimize the differentiable function while the subsequent backward step “returns” the solution to the feasible space, i.e., projects it onto  $C$ .

From this iterative technique emerges the Forward-Backward algorithm:

**Input:**  $f_1, f_2, \beta$  a Lipschitz constant of  $\nabla f_2$ .

**Output:**  $x_k \approx \arg \min_x f_1(x) + f_2(x)$ .

$y_1 = x_0 \in \mathbb{R}^M; t_1 = 1; k = 0$ .

**while** stopping criterion not satisfied **do**

$\gamma_k \in [\epsilon, \frac{2}{\beta} - \epsilon];$

$y_k = x_k - \gamma_k \nabla f_2(x_k);$

$\lambda_k \in [\epsilon, 1];$

$x_{k+1} = x_k + \lambda_k (\text{prox}_{\gamma_n;f_1}(y_k) - x_k);$

$k = k + 1;$

**end while**

In Table II,  $f_1, f_2, (\text{prox}_{\lambda;f_1}(x))_{i,v}$  and  $(\nabla f_2)_{i,v}$  are given for the non-differentiable problems that we consider here [12]. The proximity operator is defined for  $i, v$ -th component of the vector; it operates component-wise for the case of  $\ell_1$  norm, and group-wise for the case of  $\ell_{2,1}$  norm. It is important to remark that the ENet fits very easily in the above setting, as we can add the  $\ell_2$  term to the differentiable part of the optimization problem, and the proximity operator only applies

to the  $\ell_1$  term, for which we apply soft-thresholding. In other words, it is just a simple variant of standard LA. On the other hand, GL has a slightly different proximity operator in which each variable is group-wise penalized; this forces the less “powerful” groups to be inactive.

Notice that for the Forward-Backward algorithm we have to select the step-sizes  $\lambda_k$  and  $\gamma_k$ . The FISTA algorithm [10] updates automatically these parameters, as shown in the next pseudo-code:

**Input:**  $f_1, f_2, \beta$  a Lipschitz constant of  $\nabla f_2$ .  
**Output:**  $x_k \approx \arg \min_x f_1(x) + f_2(x)$ .  
 $y_1 = x_0 \in \mathbb{R}^M; t_1 = 1; k = 0$ ;  
**while** stopping criterion not satisfied **do**  
 $x_k = \text{prox}_{\frac{1}{\beta}; f_1} \left( y_k - \frac{1}{\beta} \nabla f_2(y_k) \right)$ ;  
 $t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}$ ;  
 $y_{k+1} = x_k + \frac{t_k - 1}{t_{k+1}} (x_k - x_{k-1})$ ;  
 $k = k + 1$ ;  
**end while**

This is the algorithm that will be used in our simulations.

### III. NUMERICAL EXPERIMENTS

In this section we will apply the previously explained algorithms to the concrete problem of predicting the overall wind energy production of Spain. We shall work with five meteorological variables, namely the ECMWF surface forecasts of:

- $V$ , the norm of the wind speed,
- $V_x$ , the  $x$  component of the wind speed,
- $V_y$ , the  $y$  component of the wind speed,
- $T$ , the temperature,
- $P$ , the pressure,

at each one of the points of a  $0.5^\circ$  grid that contains 522 nodes with longitudes in the interval  $[-9.5^\circ, 4.5^\circ]$  (that makes a total of 29 “columns”) and latitudes in the interval  $[35.5^\circ, 44.0^\circ]$  (with 18 “rows”). These variables are normalized to have 0 mean and a standard deviation of 1. Total feature dimension is then  $522 \times 5 = 2610$ , making mandatory the use of sparse models. Each ECMWF grid forecast has as its target value the corresponding wind energy productions normalized to the interval  $[0, 1]$  as a percentage of the overall installed wind power of Spain (about 20 GW). Figure 1 gives an example of a 90 day evolution of the wind energy target.

Although the meteorological data are available for a large geographical area, it is also obvious that only some regions of it will be close to wind farms. A very natural approach to the problem of feature selection is to select just the grid points closest to individual wind farms or, alternatively, a number of grid points more or less centered at each farm. For example, Figure 2 depicts such an approach considering either the single points nearest to wind farms (small grid selection) or the 9 points of a subgrid centered on those closest points (large grid selection). Of course this procedure requires structural

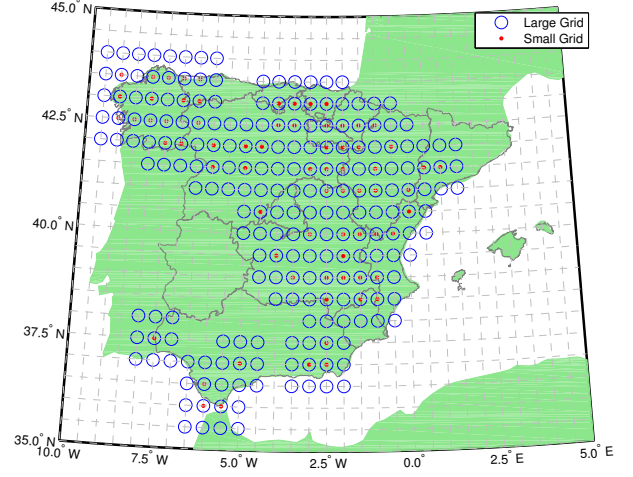


Fig. 2. Selected meteorological points for the small grid (red dots) and the large grid (blue circles).

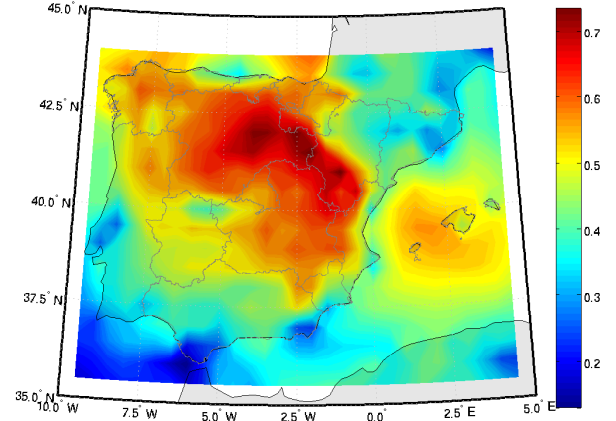


Fig. 3. Absolute correlation between the wind speed norm  $V$  and global energy production.

knowledge about the farms, which must be very precise and, at least in countries with a growing wind energy infrastructure, has to be updated quite often. Another possible drawback is to rely too much on the implicit assumption that the only relevant grid points are those close to actual wind farm location, which may not always be optimal. In fact, as a preliminary measure of the relevance of individual points, we have computed the absolute correlation between wind speed (reasonably the most important input variable) and wind energy production. As shown in Figure 3, there are grid points over the Mediterranean sea far away from any wind farm whose wind speeds are nevertheless highly correlated with wind energy.

The data used in our experiments correspond to a two year period. Since meteorological forecasts are only available every 3 hours, we will work with 8 NWP patterns per day. In order to define the training and testing sets, two different approaches are used. We have computed first a global model using as training set all the data from the first year, and applied it to the full second year as a test set. The errors corresponding

TABLE II  
APPLICATION OF PROXIMAL METHODS TO THE SPARSE REGULARIZATION PROBLEMS.

Alg	$f_1$	$f_2$	$(\text{prox}_{\lambda; f_1}(\mathbf{x}))_{i,v}$	$(\nabla f_2)_{i,v}$
LA	$\frac{\lambda_1}{M} \ w\ _1$	$\frac{1}{2N} \ Xw - y\ _2^2$	$x_{i,v} \left(1 - \frac{\lambda_1}{M} \frac{\lambda}{ x_{i,v} }\right)^+$	$\left(\frac{1}{N} X^T Xw - \frac{1}{N} X^T y\right)_{i,v}$
GL	$\frac{\lambda_1}{M} \ w\ _{2,1}$	$\frac{1}{2N} \ Xw - y\ _2^2$	$x_{i,v} \left(1 - \frac{\lambda_1}{M} \frac{\lambda}{\ x_{i,\cdot}\ _2}\right)^+$	$\left(\frac{1}{N} X^T Xw - \frac{1}{N} X^T y\right)_{i,v}$
ENet	$\frac{\lambda_1}{M} \ w\ _1$	$\frac{1}{2N} \ Xw - y\ _2^2 + \frac{\lambda_2}{2M} \ w\ _2^2$	$x_{i,v} \left(1 - \frac{\lambda_1}{M} \frac{\lambda}{ x_{i,v} }\right)^+$	$\left(\frac{1}{N} X^T Xw - \frac{1}{N} X^T y + \frac{\lambda_2}{M} w\right)_{i,v}$

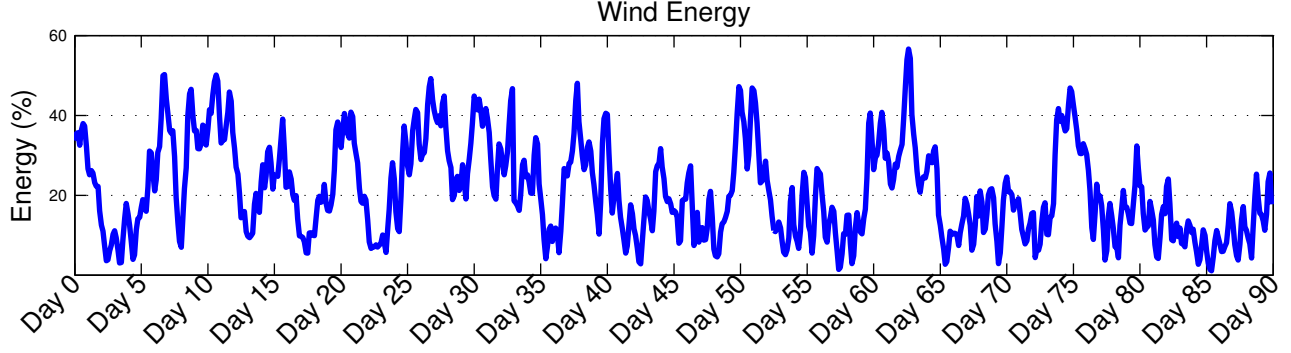


Fig. 1. Wind energy production for a 90 day period.

to this approach can help to estimate the robustness of the models, i.e., how good it remains over time. We follow a “sliding model” in the second approach, where a new model is computed every month using the previous 12 months for training and the next month for test. In other words, 12 train–test pairs are considered and once a model is built, it is applied over the daily NWP forecasts for the test month.

The models are evaluated using two measures for each framework (global and monthly models). The first error is just the Mean Absolute Error

$$\text{MAE} = \frac{1}{N} \sum_{n=1}^N |x^n \cdot w - y_n|,$$

and the second error is the Relative Mean Absolute Error,

$$\text{RMAE} = \frac{1}{N} \sum_{n=1}^N \frac{|x^n \cdot w - y_n|}{|y_n|}.$$

Thus there are four error measures for each algorithm, namely the global errors  $\text{MAE}^G$  and  $\text{RMAE}^G$  and the monthly errors  $\text{MAE}^M$  and  $\text{RMAE}^M$ . The algorithms used in our comparisons are listed in Table III.

An important issue for most of the algorithms used is the estimation of the hyper–parameters  $\lambda_1$  and  $\lambda_2$  that weight the  $\ell_1$  and  $\ell_2$  penalties. This is done by searching over a grid that defines a quantized version of parameter space. We work in both cases on a logarithmic scale that varies from  $10^{-1}$  to  $10^2$  in steps of  $10^{0.25}$ . At each point of this parameter grid, a given model is evaluated by 5–fold cross validation using as fitness the MAE. This hyper–parameter search is done only once for both global and monthly models using for this the first year data; in particular, the selected  $\lambda_1$  and  $\lambda_2$  parameters are the

TABLE III  
ALGORITHMS APPLIED.

Acronym	Description
GL	Group Lasso.
$\text{RLS}_{\text{GL}}$	RLS over the features selected by GL.
OLS	Least Squares.
RLS	Regularized Least Squares.
LA	Lasso.
$\text{RLS}_{\text{LA}}$	RLS over the features selected by LA.
ENet	Elastic–Network (RLS + LA).
$\text{RLS}_{\text{SG}}$	RLS over the small grid (1 point per farm).
$\text{RLS}_{\text{LG}}$	RLS over the large grid (9 points per farm).

same for the global and monthly models. Moreover, in order to force sparseness when a  $\ell_1$  penalty is used, we discard in the hyper–parameters search those models with more than 35% of active components.

Table IV contains the simulation results. Column 1 identifies each algorithm and columns 2 to 5 contain its corresponding four error measures described above as well as their standard deviations. They are ordered in terms of the sum of their respective rankings with respect to each one of the four errors. Table V contains in column 2 the average sparseness of the final models and the  $\log_{10}$  values of the hyper–parameters  $\lambda_1$  and  $\lambda_2$  in columns 3 and 4 respectively (column 1 identifies again each algorithm).

Looking at Table IV it can be seen that the testing performances of the different models are rather similar except for OLS. The best algorithm for this problem, according to the global rank, is  $\text{RLS}_{\text{LG}}$ , i.e., RLS over the large wind farm grid,

TABLE IV  
RESULTS OF THE SIMULATIONS, ORDERED BY GLOBAL RANK.

Method	MAE <sup>G</sup>	RMAE <sup>G</sup>	MAE <sup>M</sup>	RMAE <sup>M</sup>
<b>RLS<sub>LG</sub></b>	3.53 ± 3.1	20.20 ± 50.3	3.47 ± 3.1	19.84 ± 49.9
<b>ENet</b>	3.54 ± 3.2	20.11 ± 54.0	3.49 ± 3.1	20.01 ± 53.7
<b>LA</b>	3.55 ± 3.2	20.13 ± 54.1	3.49 ± 3.1	20.05 ± 53.7
<b>RLS</b>	3.59 ± 3.2	20.58 ± 52.6	3.55 ± 3.1	20.43 ± 52.3
<b>RLS<sub>GL</sub></b>	3.71 ± 3.3	20.52 ± 50.4	3.62 ± 3.2	20.20 ± 49.6
<b>GL</b>	3.62 ± 3.2	21.13 ± 53.8	3.59 ± 3.1	21.03 ± 53.8
<b>RLS<sub>LA</sub></b>	3.84 ± 3.4	21.43 ± 53.0	3.73 ± 3.3	21.23 ± 52.0
<b>RLS<sub>SG</sub></b>	3.88 ± 3.4	22.68 ± 52.6	3.80 ± 3.3	22.25 ± 53.3
<b>OLS</b>	8.22 ± 6.5	49.03 ± 88.1	7.53 ± 5.9	45.40 ± 81.7

TABLE V  
PARAMETERS AND SPARSITY LEVELS OBTAINED DURING TESTING.

Method	Act W	$\log_{10} \lambda_1$	$\log_{10} \lambda_2$
<b>RLS<sub>LG</sub></b>	41.8%	×	+1.50
<b>ENet</b>	26.3%	-0.25	+0.25
<b>LA</b>	26.9%	-0.25	×
<b>RLS</b>	100.0%	×	+1.75
<b>RLS<sub>GL</sub></b>	25.8%	×	+1.00
<b>GL</b>	25.8%	+0.50	×
<b>RLS<sub>LA</sub></b>	26.9%	×	+0.75
<b>RLS<sub>SG</sub></b>	13.6%	×	+0.25
<b>OLS</b>	100.0%	×	×

although the Elastic-Net performs essentially as well (ENet has a slightly better relative error for the global model, and a slightly worse one for the rest of the errors). Closely after these two algorithms comes LA and then RLS. The two-step algorithm RLS<sub>GL</sub> of RLS over the features selected by Group Lasso and the original GL follow next, and finally RLS<sub>LA</sub> and RLS<sub>SG</sub>. As expected given the similar orders of sample size and dimension, unregularized linear regression OLS performs very badly due to a clear case of over-fitting, as the training error (not shown) turns out to be very low compared to that of the regularized models.

It is interesting to identify where are located the grid points selected by the various sparse models studied. Figure 4 shows this for the GL, LA and ENet models. For GL the active weights are the same for all the meteorological variables, so an average of the 5 weights is depicted. For LA and ENet only the weights for the wind speed norm are displayed. It should be noticed that the grid points activated by LA and ENet are essentially the same, but ENet is less sparse. The reason for this is a smaller sparsity parameter  $\lambda_1$ , due to the presence of the  $\ell_2$  regularization term. On the other hand, notice that the three depicted methods detect relevant points over the Mediterranean sea. This is more clear for GL as it is the more structured model; the other two models seem to do some kind of sub-sampling over the whole space. While slightly surprising because there are obviously no wind farms in the middle of the sea, this is nevertheless in accordance with the correlation levels depicted in Figure 3.

As a summary, even without the help of structural information, ENet performs as well as RLS<sub>LG</sub> and it also achieves a more sparse model.

#### IV. DISCUSSION AND CONCLUSIONS

In this work we have revised some classical regularized linear algorithms, showing that those involving a non-differentiable  $\ell_1$  term can be put under the unifying paradigm of Proximal Optimization. In particular, this enables to use a unique optimization technique (the FISTA algorithm in our case) to train an array of linear models making use of this and similar norms, such as Lasso, Group Lasso and Elastic-Net. The use of such sparseness enforcing penalties is mandatory when sample size and dimension are similar, which is the case of the large area wind energy forecasting problem addressed here. In this line we have compared these methods in the problem of Spain's global wind energy production, using as input numerical weather forecast for a grid covering a large area enclosing the Iberian peninsula.

Our results show that ENet is as good as RLS<sub>LG</sub>, a Ridge Regression model making use of prior information about the problem to perform an intelligent feature selection, furthermore yielding significantly more sparse models. ENet beats also the classical LA model, something not surprising considering that the latter can be regarded as a particular case of the former.

Concluding, we have shown ENet to be a good model for wind power prediction which does not require any expert knowledge or structural information about wind farms, while producing a sparse selection of meteorological nodes. Such sparsity is highly desirable in a real wind power forecasting system, where the economic cost of acquiring daily NWP data scales with the number of nodes required. Finally, the optimization problem underlying ENet is solved easily and elegantly under the general Proximal Optimization framework.

As future work, we intend to consider models with more advanced structure-inducing properties. In fact, a drawback of ENet or LA is that the obtained features do not present a clear geographical structure, as they are sparse over all the variables considered independently of the position of grid points. GL, on the other hand, is able to find a more defined structure, but still somewhat far from the "expert" one showed in Figure 2. Therefore, it is very reasonable to study other models taking these considerations into account. An option for this is the Fused Lasso model [13], for which there exist very efficient algorithms such as [14] that encourage weights corresponding to geographically-close points to have the same coefficients, thus providing an "spatial grouping" effect. However, a problem of this method is that it does not consider multivariate points, as the ones present in the wind energy prediction problem. Thus, further work towards a Group Fused Lasso algorithm would be of interest.

Apart from this it would also be of interest to explore  $\ell_0$  models that enforce an even greater degree of sparsity, such as the Garrote [15]. This could possibly be further combined with an  $\ell_2$  regularization term to obtain a kind of  $\ell_0$ -sparse



ENet model. Unfortunately, models making use of the  $\ell_0$  norm are known to result in non-convex NP-hard problems, thus demanding further research towards their applicability in practical settings.

#### ACKNOWLEDGEMENT

The authors of the paper acknowledge partial support from grant TIN2010-21575-C02-01 of the TIN Subprogram from Spain's MICINN and of the Cátedra UAM-IIC en Modelado y Predicción. The first author is also supported by the FPU-MEC grant AP2008-00167. We also thank Red Eléctrica de España, Spain's TSO, for providing historic wind energy data.

#### REFERENCES

- [1] (2005) European center for medium-range weather forecasts. [Online]. Available: <http://www.ecmwf.int/>
- [2] (2012) Global forecast system. [Online]. Available: <http://www.emc.ncep.noaa.gov/GFS/>
- [3] C. Monteiro, R. Bessa, V. Miranda, A. Botterud, J. Wang, and G. Conzelmann, "Wind power forecasting: State-of-the-art 2009," INESC Porto and Argonne National Laboratory, Tech. Rep., 2009.
- [4] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Statist. Soc. Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.
- [5] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society – Series B: Statistical Methodology*, vol. 68, no. 1, pp. 49–67, 2006.
- [6] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society – Series B: Statistical Methodology*, vol. 67, no. 2, pp. 301–320, 2005.
- [7] P. L. Combettes and J. C. Pesquet, "Proximal splitting methods in signal processing," *Recherche*, vol. 49, pp. 1–25, 2009.
- [8] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 12, pp. 55–67, 1970.
- [9] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.
- [10] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.
- [11] S. Mosci, L. Rosasco, M. Santoro, A. Verri, and S. Villa, "Solving structured sparsity regularization with proximal methods," in *ECML/PKDD (2)*, Berlin, Heidelberg, 2010, pp. 418–433.
- [12] M. Kowalski and B. Torrèsani, "Structured sparsity: from mixed norms to structured shrinkage," in *SPARS'09 – Signal Processing with Adaptive Sparse Structured Representations*, R. Gribonval, Ed. Saint Malo, France: Inria Rennes – Bretagne Atlantique, 2009.
- [13] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, "Sparsity and smoothness via the fused lasso," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 1, pp. 291–308, 2005.
- [14] A. Barbero and S. Sra, "Fast newton-type methods for total variation regularization," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, New York, NY, USA, June 2011, pp. 313–320.
- [15] L. Breiman, "Better subset regression using the nonnegative garrote," *Technometrics*, vol. 37, no. 42, pp. 373–384, 1995.

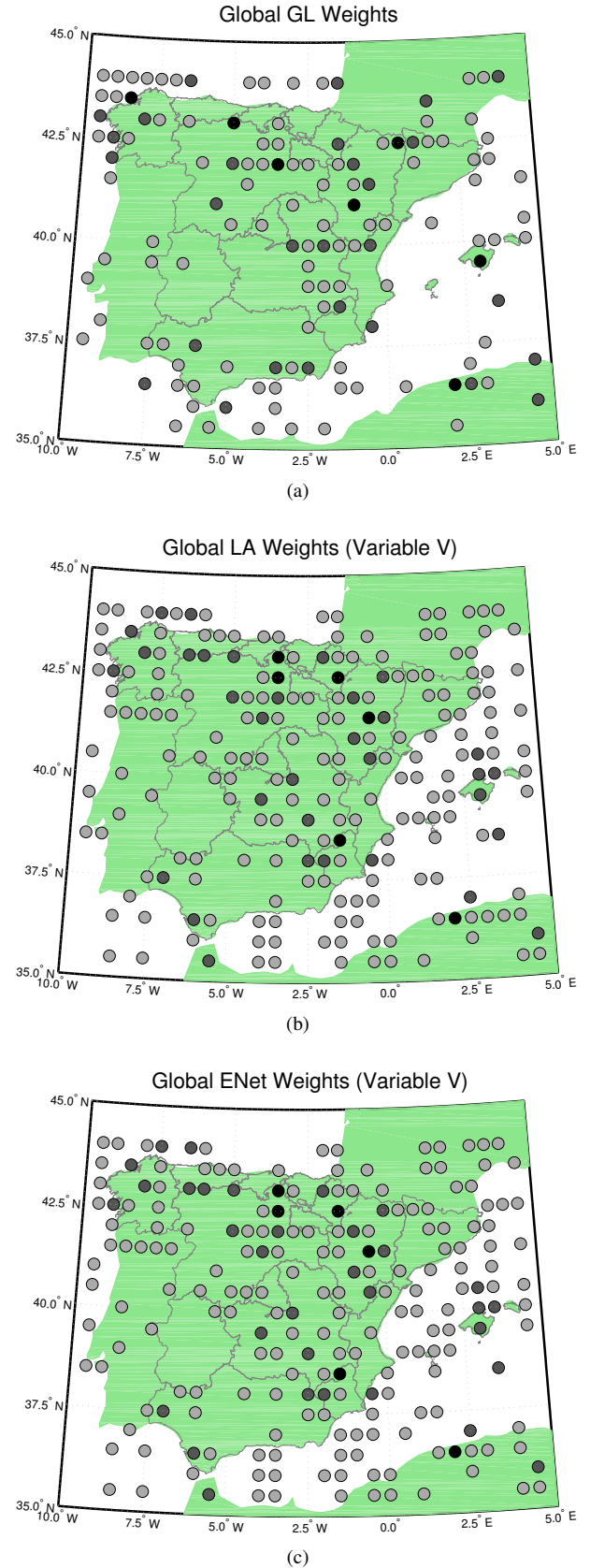


Fig. 4. Global weights for 3 of the models. Darker points represent bigger absolute weights. (a): Average global weights for GL. (b): Wind speed norm global weights for LA. (c): Wind speed norm global weights for ENet.