

---

# A Comparative Study of Local Optimization Methods for Regularized Least Squares Regressions in Depression Scale Analysis

---

Ángela Téllez González  
Universidad Carlos III de Madrid

## Abstract

First-order optimization methods are compared for convex regularized least squares in depression scale modelling, including Gradient Descent, Nesterov acceleration, the Subgradient Method, ISTA, and FISTA. Using NHANES accelerometer and sociodemographic data, convergence, speed, sparsity, and stability are analysed across penalties. Nesterov improves performance for smooth objectives, while FISTA clearly outperforms ISTA and subgradient updates for non-smooth ones. Despite similar predictive accuracy after tuning, the choice of penalty and optimizer strongly affects sparsity and stability, which are key for interpretability.

## 1 Introduction and Motivation

Understanding how behavioural and sociodemographic factors relate to depression severity is an important goal in public health. Large surveys such as NHANES [1] provide rich accelerometer-based and demographic information, but modelling PHQ-9 scores in this setting requires handling many correlated predictors. Regularized linear regression is a natural choice: penalties such as  $\ell_1$ ,  $\ell_2$ , and intermediate  $\ell_p$  norms improve generalization and yield more interpretable models.

These penalties lead to convex optimization problems with different degrees of smoothness. Smooth objectives are well suited to classical gradient-based methods, whereas sparsity-inducing penalties involving  $\ell_1$  require subgradient or proximal algorithms. As a result, the choice of optimization method directly affects convergence speed, computational cost, and the structure of the fitted model.

In this work, several first-order algorithms for convex regularized least squares are compared using PHQ-9

prediction from NHANES accelerometer and sociodemographic data as a case study. Convergence behaviour, efficiency, sparsity, and stability across regularizations are compared, providing practical guidance for choosing both the penalty and the optimization algorithm in health-related regression problems.

## 2 Problem Formulation

Let  $X \in R^{n \times d}$  be the predictor matrix ( $n = 2519$ ,  $d = 163$ ), containing accelerometer-based and sociodemographic features, and let  $y \in R^n$  be the PHQ-9 scores. We estimate coefficients  $\beta \in R^d$  by solving the least squares problem

$$\min_{\beta} f(\beta) = \min_{\beta} \frac{1}{2} \|y - X\beta\|_2^2.$$

### 2.1 Regularized Least Squares Model

Because many predictors are correlated and sparse solutions are desirable for interpretability, we consider the regularized formulation:

$$\min_{\beta} \frac{1}{2} \|y - X\beta\|_2^2 + \lambda g(\beta),$$

where  $\lambda > 0$  controls shrinkage. For Elastic Net, the penalty is

$$g(\beta) = \alpha \|\beta\|_1 + (1 - \alpha) \|\beta\|_2^2,$$

with  $\alpha \in [0, 1]$  balancing sparsity and smoothness.

We also study intermediate  $\ell_p$  penalties,

$$g(\beta) = \|\beta\|_p^p, \quad p \in \{1.2, 1.5, 1.8\},$$

which interpolate between Lasso and Ridge while remaining convex for  $p \geq 1$ .

Non-smooth penalties involving  $\ell_1$  introduce non-differentiability at zero and require subgradient or proximal methods, whereas smooth penalties ( $p > 1$  or Ridge) are manageable to classical gradient-based optimization. This motivates the comparison of different first-order algorithms in the next section.

### 3 Optimization Methods

Several first-order methods are compared, all relying on the gradient of the smooth data-fitting term, whose Hessian  $\nabla^2 f(\beta) = X^\top X \succeq 0$  shows that  $f$  is convex with Lipschitz-continuous gradient. The full objective is written as  $F(\beta) = f(\beta) + g(\beta)$ , where  $g(\beta)$  contains the regularization terms. Smooth formulations can be tackled with standard gradient-based methods, while the non-smooth  $\ell_1$  penalty motivates subgradient and proximal algorithms.

#### 3.1 Gradient Descent and Nesterov Accelerated Gradient (NAG)

Gradient Descent (GD) updates the parameters along the negative gradient of the objective. For differentiable convex functions it converges at rate  $O(1/k)$ , but its performance is highly sensitive to the choice of step size. To avoid manual tuning, a backtracking line search that reduces the step size until Armijo's sufficient decrease condition is met is also implemented (see Appendix A.1).

Nesterov's Accelerated Gradient (NAG) [2] improves Gradient Descent by evaluating the gradient at a *look-ahead* point, which yields the optimal  $O(1/k^2)$  convergence rate for smooth convex functions. The method maintains an extrapolated sequence  $y_k = \beta_k + \gamma_k(\beta_k - \beta_{k-1})$  and updates following  $\beta_{k+1} = y_k - t \nabla f(y_k)$ , where  $t > 0$  is a fixed step size and  $\gamma_k \in (0, 1)$  controls the momentum. NAG is therefore, as Gradient Descent, suited only to smooth penalties, while its proximal extension, FISTA, handles non-smooth objectives.

#### 3.2 Subgradient Method

The Subgradient Method [3] extends Gradient Descent to convex but non-differentiable objectives, such as those involving the  $\ell_1$  penalty (see Appendix A.2). At each iteration it uses a subgradient  $g_k \in \partial F(\beta_k)$  in the update  $\beta_{k+1} = \beta_k - t_k g_k$ , with step sizes  $t_k$  decreasing to zero and satisfying  $\sum_k t_k = \infty$ . For convex problems this scheme is guaranteed to converge, but only at rate  $O(1/\sqrt{k})$ , which makes it considerably slower than proximal-gradient methods in practice. In our experiments it serves mainly as a baseline for Lasso-type regularization.

#### 3.3 Proximal Algorithms (ISTA & FISTA)

Proximal algorithms [4] solve composite objectives  $F(\beta) = f(\beta) + g(\beta)$  where  $f$  is smooth and  $g$  is convex but non-differentiable. This setting includes Lasso and Elastic Net, whose  $\ell_1$  term is handled through the proximal operator.

**ISTA.** ISTA applies a proximal gradient update

$$\beta_{k+1} = \text{prox}_{\frac{1}{L}g} \left( \beta_k - \frac{1}{L} \nabla f(\beta_k) \right),$$

yielding sparse solutions and converging at rate  $O(1/k)$ .

**FISTA.** FISTA [5] incorporates Nesterov momentum, achieving the accelerated  $O(1/k^2)$  rate. Each iteration forms an extrapolated point and performs a proximal step at that location. In practice FISTA is much faster than ISTA and is our preferred method for Lasso and Elastic Net.

See details in Appendix A.3.

## 4 Experiments and Results

We compare the methods across different regularizations and penalty values ( $\lambda$ ), focusing on convergence trajectories, computation time, and sparsity. After hyperparameter selection, all regularizations achieve very similar test MSE (Appendix B). The main differences lie in convergence behaviour and interpretability.

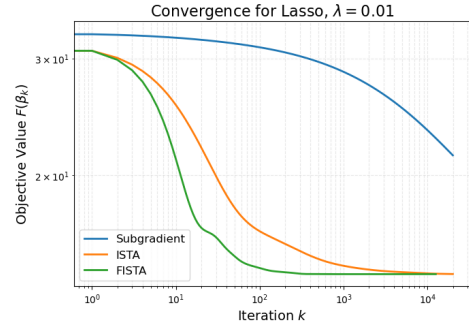


Figure 1: Convergence trajectories of Subgradient, ISTA, and FISTA for Lasso with  $\lambda = 0.01$ . The objective value  $F(\beta_k)$  is shown on a logarithmic scale.

Figure 1 compares the trajectories of FISTA, ISTA and the Subgradient Method for Lasso. FISTA converges the fastest, ISTA is slower, and the Subgradient Method is markedly slower. ISTA and FISTA start from the same objective value because both evaluate  $F(\beta_0)$  before updating, whereas the Subgradient Method can increase the objective in its first step and thus begins from a higher point. The small non-monotonic fluctuation in the FISTA curve is due to its momentum term, which may briefly overshoot before accelerating the descent.

Similarly, Figure 2 shows the trajectories of Gradient Descent (GD) with a fixed learning rate, GD with backtracking, and Nesterov's method for Ridge. GD with backtracking attains a lower objective in the initial iterations because the line search adapts the step

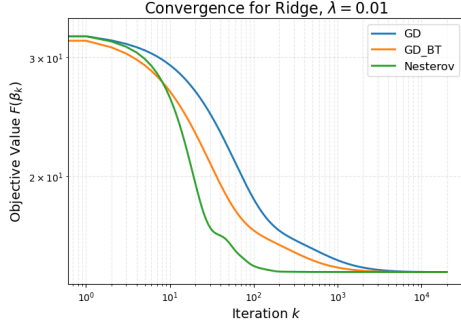


Figure 2: Convergence trajectories of Gradient Descent, Gradient Descent with Backtracking and Nesterov for Ridge with  $\lambda = 0.01$ . The objective value  $F(\beta_k)$  is shown on a logarithmic scale.

size to the local curvature, often selecting more effective steps than the fixed learning rate. As iterations progress, Nesterov’s accelerated updates yield a faster convergence rate and overtake both GD variants. The slight non-monotonic behaviour in the Nesterov curve again reflects momentum-induced overshoot.

For the intermediate  $p$ -norm regularizations, the convergence behaviour is consistent with the previously analysed cases. Figure 3 shows the trajectories for  $p = 1.2$  under two values of  $\lambda$ : larger  $\lambda$  yields higher objective values at convergence due to the stronger regularization, but also smoother landscapes and faster convergence. A qualitatively similar pattern appears for Elastic Net in Figure 4: increasing  $\lambda$  enlarges the separation between the trajectories, reflecting a stronger regularization effect. For  $\lambda = 0.01$ , the differences in objective values across  $\alpha$  are small but already translate into distinct sparsity patterns (Figure 5); for larger  $\lambda$  these differences become more pronounced.

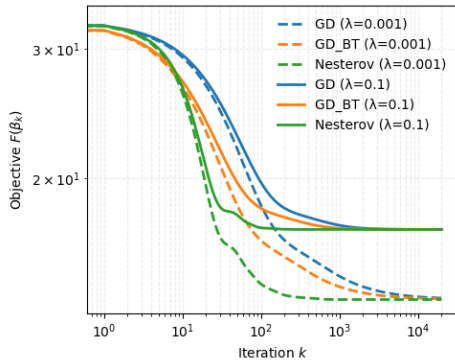


Figure 3: Convergence trajectories for  $p = 1.2$  with different values of  $\lambda$ .

Analyzing computational cost, despite performing different update rules, all methods exhibit similar average times per iteration (Table 1). This is expected,

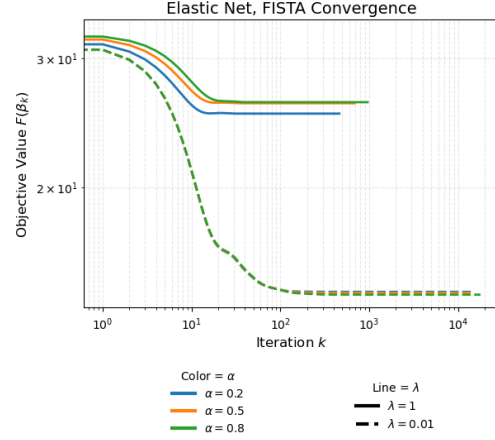


Figure 4: Convergence trajectories of Elastic Net using FISTA for different values of  $\alpha$ . The figure shows two regularization strengths,  $\lambda = 1$  and  $\lambda = 0.01$ .

since each algorithm requires one gradient evaluation per iteration and this is the computationally dominant operation. The additional steps in accelerated or proximal methods (momentum updates, soft-thresholding) are inexpensive vector operations. Small differences in timing arise mainly from implementation-level factors rather than from meaningful algorithmic complexity. Consequently, the practical efficiency of each method is determined not by the cost of a single iteration but by how many iterations it requires to converge.

Sparsity and shrinkage patterns are examined in Figures 5 and 6. Figure 5 displays sorted coefficient magnitudes (log scale). In the top panel, as the norm  $p$  moves from Lasso ( $p = 1$ ) toward Ridge ( $p = 2$ ), smaller  $p$  values induce stronger shrinkage and push many coefficients towards zero, whereas larger  $p$  yield denser, smoother coefficients. The bottom panel shows Elastic Net solutions for different  $\alpha$ . Higher  $\alpha$  (closer to Lasso) produce more coefficients exactly equal to zero, while smaller  $\alpha$  shift the solution towards Ridge behaviour, with less sparsity.

Figure 6 summarizes how Elastic Net changes with  $\alpha$  in terms of the number of nonzero coefficients (left axis) and the overall shrinkage  $\sum |\beta|$  (right axis). As  $\alpha$  approaches 1, both curves decrease, reflecting stronger sparsity and shrinkage. A small fluctuation near  $\alpha = 1$  corresponds to the pure Lasso case and illustrates the instability associated with sharp  $\ell_1$  thresholding. For  $\alpha < 1$ , the  $\ell_2$  component stabilizes the solution. This stability effect is also reflected in Table 2. Lasso shows slightly higher similarity in the top-10 selected features, but Elastic Net exhibits more stable selections when larger sets of coefficients are considered, indicating that the  $\ell_2$  term improves robustness.

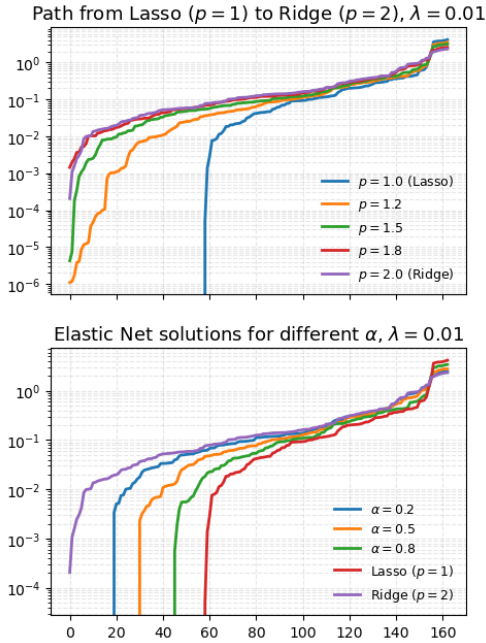


Figure 5: Sorted absolute coefficients obtained with  $\lambda = 0.01$  for Lasso, Ridge and intermediate  $\ell_p$  penalties (top) and Elastic Net solutions for several  $\alpha$  values.

Table 1: Average time per iteration (ms) for each optimization method.

GD	NAG	GD-BT	ISTA	FISTA	Subg.
1.263	1.293	1.485	1.555	1.510	1.686

## 5 Conclusions and Contributions

Across all configurations, once the regularization parameter  $\lambda$  is tuned by cross-validation, the different penalties achieve very similar test performance, with mean squared errors around 11.5 on the PHQ-9 scale (root mean squared error  $\approx 3.4$  on a 0–27 range). Thus, predictive accuracy is largely comparable across Ridge, Lasso, Elastic Net and intermediate  $\ell_p$  norms.

From an optimization viewpoint, the results confirm the theory. For smooth penalties (Ridge and  $\ell_p$  with  $p > 1$ ), Nesterov’s accelerated gradient clearly outperforms standard Gradient Descent and Gradient Descent with backtracking. For non-smooth penalties, FISTA consistently converges faster than ISTA, while the Subgradient Method is substantially slower.

Structurally, the sparsity and shrinkage analyses suggest that Lasso and Elastic Net with larger values of  $\alpha$  tend to produce sparser models, although this sparsity may come with increased variability across cross-

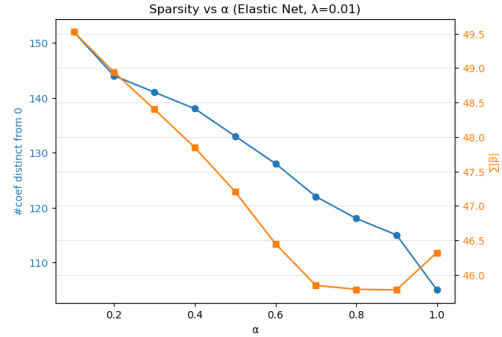


Figure 6: Number of coefficients distinct from zero and absolute sum of coefficients in Elastic Net as a function of parameter  $\alpha$ .

Table 2: Stability analysis over 5-fold CV for Lasso and Elastic Net ( $\alpha = 0.8$ ). Values correspond to the Jaccard similarity (J) of the top- $J$  selected features across folds.

Method	J. 10	J. 30	J. 50
LASSO	0.891	0.473	0.474
ElasticNet ( $\alpha=0.8$ )	0.836	0.582	0.545

validation folds. Introducing an  $\ell_2$  component generally leads to smoother and less sparse coefficient profiles, and may help mitigate some of this variability. In our experiments, the combination of  $\ell_1$  and  $\ell_2$  penalties appears to provide a balance between sparsity and stability, though the extent of this effect may depend on the specific dataset and regularization settings.

This work makes three contributions. First, we present a comparison of several first-order optimization methods for convex regularized least squares in depression scale modelling, examining their practical behaviour on a high-dimensional health dataset. Second, we explore the use of a family of regularized linear models (Ridge, Lasso, Elastic Net and intermediate  $\ell_p$  penalties with  $p \in 1.2, 1.5, 1.8$ ) to predict continuous PHQ-9 scores from NHANES 2005–2006 accelerometer and sociodemographic data. To our knowledge, such a combined analysis has not been extensively documented for this dataset. Third, we study sparsity, shrinkage and stability within this setting, observing that the inclusion of an  $\ell_2$  component can provide improvements in the stability and interpretability of the resulting models, a behaviour that, to the best of our understanding, has not been previously discussed for this dataset.

Code is available at [6].

## References

- National Center for Health Statistics (NCHS) . National Health and Nutrition Examination Survey: 2005–2006 Data Documentation, Codebook, and Frequencies. U.S. Department of Health and Human Services, Centers for Disease Control and Prevention. Available at: <https://wwwn.cdc.gov/nchs/nhanes/>
- Y. Nesterov (1983). A method for solving the convex programming problem with convergence rate  $O(1/k^2)$ . *Soviet Mathematics Doklady*, **27**:372–376.
- N. Z. Shor (1985). *Minimization Methods for Nondifferentiable Functions*. Springer Series in Computational Mathematics, Springer.
- N. Parikh and S. Boyd (2014). Proximal algorithms. *Foundations and Trends in Optimization*, **1**(3):127–239.
- A. Beck and M. Teboulle (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, **2**(1):183–202.
- Á. Téllez González (2025). Optimization-project (GitHub repository). Available at: <https://github.com/angelatellezz07/Optimization-project>.

## A More on optimization methods

### A.1 Backtracking Line Search and Armijo Rule

Given a current iterate  $\beta_k$  and descent direction  $-\nabla f(\beta_k)$ , backtracking selects the largest step size  $t = \rho^m t_0$  ( $m = 0, 1, 2, \dots$ ) such that the Armijo condition holds:

$$F(\beta_k - t\nabla f(\beta_k)) \leq F(\beta_k) - ct \|\nabla f(\beta_k)\|^2,$$

with parameters  $c \in (0, 1)$ ,  $\rho \in (0, 1)$ , and initial step  $t_0 > 0$ .

### A.2 Subgradient of the $\ell_1$ Norm

For completeness, the subgradient of  $|\beta_i|$  used in Lasso is

$$\partial|\beta_i| = \begin{cases} \{1\}, & \beta_i > 0, \\ \{-1\}, & \beta_i < 0, \\ [-1, 1], & \beta_i = 0. \end{cases}$$

### A.3 Details on Proximal Algorithms

Proximal gradient methods are designed to solve composite optimization problems of the form

$$F(\beta) = f(\beta) + g(\beta),$$

where  $f$  is convex and smooth, and  $g$  is convex but possibly non-differentiable (e.g. the  $\ell_1$  norm). A key property enabling these methods is that the smooth component  $f$  has a Lipschitz-continuous gradient, meaning that there exists  $L > 0$  such that

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \quad \forall x, y.$$

This condition implies the quadratic upper bound

$$f(y) \leq f(x) + \nabla f(x)^\top (y - x) + \frac{L}{2} \|y - x\|^2.$$

Intuitively, this inequality ensures that  $f$  can be globally majorized by a quadratic model whose curvature is controlled by  $L$ .

This motivates the surrogate function used in proximal gradient methods:

$$Q_L(\beta; \beta_k) = f(\beta_k) + \nabla f(\beta_k)^\top (\beta - \beta_k) + \frac{L}{2} \|\beta - \beta_k\|^2 + g(\beta).$$

At each iteration, ISTA replaces the original objective  $F$  by this surrogate and minimizes it exactly. Because the quadratic term dominates the linearization,  $Q_L$  is strongly convex and has a unique minimizer. Minimizing  $Q_L$  yields the proximal gradient update

$$\beta_{k+1} = \text{prox}_{\frac{1}{L}g}(\beta_k - \frac{1}{L}\nabla f(\beta_k)),$$

where the proximal operator of  $g$  is defined as

$$\text{prox}_{tg}(z) = \arg \min_u \left\{ g(u) + \frac{1}{2t} \|u - z\|^2 \right\}.$$

**ISTA for Lasso.** For the Lasso, the non-smooth component is  $g(\beta) = \lambda \|\beta\|_1$ , whose proximal operator corresponds to elementwise soft-thresholding:

$$\text{prox}_{t\lambda\|\cdot\|_1}(z_i) = \text{sign}(z_i) \max\{|z_i| - t\lambda, 0\}.$$

Thus ISTA consists of a gradient descent step on the smooth loss  $f$  followed by a soft-thresholding step that promotes sparsity through the  $\ell_1$  penalty. ISTA converges at rate  $O(1/k)$ .

**Acceleration via FISTA.** FISTA [5] improves ISTA by introducing a momentum term via the extrapolated sequence

$$y_k = \beta_k + \frac{t_{k-1} - 1}{t_k} (\beta_k - \beta_{k-1}), \quad t_k = \frac{1 + \sqrt{1 + 4t_{k-1}^2}}{2}.$$

The proximal update is then performed at  $y_k$ :

$$\beta_{k+1} = \text{prox}_{\frac{1}{L}g}(y_k - \frac{1}{L}\nabla f(y_k)).$$

This modification achieves the optimal convergence rate

$$F(\beta_k) - F^* = O(1/k^2),$$

which is known to be the fastest possible rate for first-order methods applied to this class of problems.

## B More results

Table 3: Comparison of Regularization Methods

Reg.	$\lambda^*$	Method	F Final	MSE Test
Ridge	1e-2	NAG	14.4	11.7
$p=1.8$	1e-2	NAG.	14.4	11.7
$p=1.5$	1e-2	NAG	14.3	11.7
$p=1.2$	0.1	NAG	17.0	11.6
Lasso	0.1	FISTA	16.9	11.5
ElasticNet ( $\alpha=0.8$ )	1e-2	FISTA	14.3	11.7