Linear Regression Analytics in DROP

v5.70 13 September 2022

Tikhonov Regularization

Overview

1. Regularization of Ill-posed Problems: *Tikhonov regularization* is a method for regularization of ill-posed problems (Wikipedia (2022)).
2. Alternate Terms for Tikhonov Regularization: In statistics, this method is known as *ridge regression* – in machine learning, it and its modifications are known as *weight decay*, and with multiple independent discoveries, it is also known as the *Tikhonov-Miller* method, the *Phillips-Twomey* method, the *constrained linear inversion* method, l_2 *regularization*, and the method of *linear regularization*. It is related to the Levenberg-Marquardt method of non-linear least squares problems (Wikipedia (2022)).
3. Mitigation of the Multi-collinearity Challenge: It is particularly useful to mitigate the problem of multi-collinearity in linear regression, which commonly occurs in models with large number of parameters (Kennedy (2003)).
4. Improved Efficiency for a Given Bias: In general, the method provides an improved efficiency in parameter estimation problems in exchange for a tolerable amount of bias (Gruber (1998)).
5. Adding Positive Elements to Diagonals: In the simplest case, the problem of a near singular moment matrix $X^T X$ is alleviated by adding positive elements to the diagonals, thereby decreasing its condition number.
6. Ridge vs. the OLS Estimator: Analogous to the OLS Estimator, the simple ridge estimator is then given by

$$\hat{\beta}_R = [X^T X + \lambda I]^{-1} X^T y$$

where y is the regressand, X is the design matrix, I is the identity matrix, and the ridge parameter

$$\lambda \geq 0$$

serves as the constant shifting the diagonals of the moment matrix (Khalaf and Shukur (2005)).

7. Lagrangian of Constrained Least Squares: It can be shown that this estimator is the solution to the least-squares problem subject to the constraint

$$\beta^T \beta = c$$

which can be expressed as a Lagrangian

$$\min_{\beta} \{(y - X\beta)^T (y - X\beta) + \lambda(\beta^T \beta - c)\}$$

which shows that λ is nothing but the Lagrange multiplier of the constraint.

8. λ Determined by a Heuristic: Typically, λ is chosen according to a heuristic criterion, so the constraint will not be satisfied exactly.
9. OLS corresponds to Non-binding Constraint: Specifically in the case of

$$\lambda = 0$$

in which the constraint is non-binding, the ridge estimator reduces to ordinary lease squares. A more general approach to Tikhonov regularization is discussed later in this chapter.

Tikhonov Regularization

1. Ordinary Least-squares Linear Regression: Suppose that for a known matrix A and vector b one wishes to find a vector x such that

$$Ax = b$$

The standard approach is ordinary least squares linear regression.

2. Over/Under-determined System of Equations: However, if no x satisfies the equation or more than one x does – that is, the solution is not unique – the problem is said to be ill-posed. In such cases, the ordinary least squares method leads to an over-determined – or more often an under-determined system of equations.
3. Forward-Direction Low-Pass Filters: Most real-world phenomena have the effect of low-pass filters in the forward direction where A maps x to b.
4. High-pass Filter Amplifier Noise: Therefore, in solving the inverse problem, the inverse mapping operator of the high-pass filter that has the undesirable tendence of amplifying noise, i.e., eigenvalues/singular values are largest in the reverse mapping where they were smallest in the forward mapping.
5. Nullify every Element of x: In addition, ordinary least squares implicitly nullifies every element of the reconstructed version of x that is in the null-space of A rather than allowing for a model to be used as a prior for x.
6. Minimize Sum of Squared Residuals: OLS seeks to minimize the sum of squared residuals, which can be compactly written as $\|Ax - b\|_2^2$ where $\|\cdot\|_2$ is the Euclidean norm.

7. Regularization Term included in Minimization: In order to give preference to a particular solution with desirable properties, a regularization term can be included in the minimization of $\|Ax - b\|_2^2 + \|\Gamma x\|_2^2$ for some suitably chosen *Tikhonov matrix* Γ.
8. Tikhonov as Scaled Identity Matrix: In many cases, this matrix is chosen as a scalar multiple of the identity matrix

$$\Gamma = \alpha I$$

giving preferences to solutions with smaller norms; this is known as l_2 *regularization*.
9. High-pass Operators Enforce Smoothness: In other cases, the high-pass operators, e.g., the difference operator or a weighted Fourier operator, may be used to enforce smoothness if the underlying vector is believed to be mostly continuous.
10. Computing a Closed Form Solution: This regularization improves the conditioning of the problem, thus enabling a direct numerical solution. An explicit solution, denoted by $\hat{x}$, is given by

$$\hat{x} = [A^T A + \Gamma^T \Gamma]^{-1} A^T b$$

11. Regularization Controlled by Scale of Γ: The effect of regularization may be varied by the scale of Γ. For

$$\Gamma = 0$$

this reduces to the unregularized least-squares solution, provided $[A^T A]^{-1}$ exists.
12. Contexts where l_2 Regularized is Used: l_2 is used in many contexts aside from linear regression, such as classification with logistic regression or support vector machine (Fan,

Chang, Hsieh, Wang, and Lin (2008)), and matrix factorization (Guan, Tao, Luo, and Yuan (2012)).

Generalized Tikhonov Regularization

1. Multivariate Normal Distribution of x: For general multivariate normal distributions of x and data error, one can apply a transformation of variables to reduce to the case above.
2. Weighted Norm Squared Covariate Penalty: Equivalently, one can seek and x to minimize $\|Ax - b\|_P^2 + \|x - x_0\|_Q^2$ where $\|x\|_Q^2$stands for the weighted square norm $x^T Q x$ – to be compared with the Mahalanobis distance.
3. Defining the Terms in the Optimization: In the Bayesian interpretation, P is the inverse covariance matrix of b, x_0 is the expected value of x, and Q is the inverse covariance matrix of x.
4. Tikhonov Matrix as a Whitening Filter: The Tikhonov matrix is then given as a factorization of

$$Q = \Gamma^T \Gamma$$

   i.e., the Cholesky factorization, and is considered a whitening filter.
5. Optimal Solution of the Generalized Problem: This generalized problem has an optimal solution x^* which can be set explicitly using the formula

$$x^* = [A^T P A + Q]^{-1}[A^T P b + Q x_0]$$

   or equivalently

$$x^* = [A^T P A + Q]^{-1}[A^T P(b - Ax_0)]$$

Lavrentiev Regularization

1. Lavrentiev Scheme for Simplifying Tikhonov: In some situations, one can avoid using the transpose A^T, as proposed in Lavrentiev (1967).
2. Condition – A that is PSD: For example, if A is symmetric positive definite, i.e.,

$$A^T = A > 0$$

so, its inverse is A^{-1}, which can thus be used to setup the weighted norm squared

$$\|x\|_p^2 = x^T A^{-1} x$$

in the generalized Tikhonov regularization, leading to minimizing $\|Ax - b\|_{A^{-1}}^2 + \|x - x_0\|_Q^2$ or, equivalently up to a constant term, $x^T(A + Q)x - 2x^T(b + Qx_0)$

3. Optimal Solution to Minimization Problem: This minimization problem has an optimal solution x^* which can be written explicitly using the formula

$$x^* = (A + Q)^{-1}(b + Qx_0)$$

which is nothing but the solution to the generalized Tikhonov problem, where

$$A = A^T = p^{-1}$$

4. Advantage over Original Tikhonov Regularization: The Lavrentiev regularization, is applicable, is advantageous to the original Tikhonov regularization, since the Lavrentiev matrix $A + Q$ can be better conditioned, i.e., have a smaller condition number, compared to the Tikhonov matrix $A^T A + \Gamma^T \Gamma$.

Regularization in Hilbert Space

1. Ill-conditioned Discretization of Integral Equations: Typically, discrete ill-conditioned problems result from discretization of integral equations, and one can formulate a Tikhonov regularization in the original infinite-dimensional context.
2. Compact Operator on Hilbert Spaces: In the above, one can interpret A as a compact operator on Hilbert spaces, and x and b as elements in the domain and the range of A.
3. Self-adjoint Bounded Invertible Operator: The operator $A^* A + \Gamma^* \Gamma$ is then a self-adjoint bounded invertible operator.

Relation to Singular-Value Decomposition and Wiener Filter

1. Least-squares Solution using SVD: With

$$\Gamma = \alpha I$$

this least-squares solution can be analyzed in a special way using the singular-value decomposition.

2. Re-casting the Tikhonov Regularized Solution: Given the singular value decomposition

$$A = U\Xi V^T$$

with singular σ_i, the Tikhonov regularized solution can be expressed as

$$\hat{x} = VDU^{-1}b$$

where D has diagonal values

$$D_{ii} = \frac{{\sigma_i}^2}{{\sigma_i}^2 + \alpha^2}$$

and is zero elsewhere.

3. Impact of the Tikhonov Parameter: This demonstrates the effect of the Tikhonov parameter on the condition number of the regularized problems.
4. Hansen's Extension to Generalized SVD: For the generalized case, a similar representation can be derived using a generalized singular-value decomposition (Hansen (1998)).
5. Relation to the Wiener Filter: Finally, it is related to the Wiener filter

$$\hat{x} = \sum_{i=1}^{q} f_i \frac{u_i^T b}{\sigma_i} v_i$$

where the Weiner weights are

$$f_i = \frac{{\sigma_i}^2}{{\sigma_i}^2 + \alpha^2}$$

and q is the rank of A.

Determination of the Tikhonov Factor

1. Bayesian Interpretation to determine α: The optimal regularization α is usually unknown and often in practical problems is determined by an *ad hoc* method. A possible approach relies on the Bayesian interpretation described below.
2. Other Approaches to determine α: Other approaches include the discrepancy principle, cross-validation, L-curve method (Hansen (2005)), restricted maximum likelihood, and unbiased predictive risk estimator.
3. Leave-one-out Cross-validation: Wahba proved that the optimal parameter, in the lease of leave-one-out cross-validation minimizes (Golub, Heath, and Wahba (1979), Wahba (1990))

$$G = \frac{RSS}{\tau^2} = \frac{\left\|X\hat{\beta} - y\right\|^2}{[Tr\{I - X(X^TX + \alpha^2 I)^{-1}X^T\}]^2}$$

where RSS is the residual sum of squares, and τ is the effective number of degrees of freedom.

4. Simplifying above Expression using SVD: Using the previous SVD decomposition, the above expression can be simplified as

$$RSS = \left\| y - \sum_{i=1}^{q} (u_i^T b) u_i \right\|^2 + \left\| \sum_{i=1}^{q} \frac{\alpha^2}{{\sigma_i}^2 + \alpha^2} (u_i^T b) u_i \right\|^2$$

$$RSS = RSS_0 + \left\| \sum_{i=1}^{q} \frac{\alpha^2}{{\sigma_i}^2 + \alpha^2} (u_i^T b) u_i \right\|^2$$

and

$$\tau = m - \sum_{i=1}^{q} \frac{{\sigma_i}^2}{{\sigma_i}^2 + \alpha^2} = \tau = m - q + \sum_{i=1}^{q} \frac{\alpha^2}{{\sigma_i}^2 + \alpha^2}$$

Relation to Probabilistic Formulation

1. Probabilistic Formulation of the Inverse Problem: The probabilistic formulation of the inverse problem introduces – when all uncertainties are Gaussian – covariance matrix C_M representing the *a priori* uncertainties on the model parameters, and a covariance matrix C_D representing the uncertainties on the observed parameters (Tarantola (2005)).
2. Diagonal and Isotropic Uncertainty Matrix: In the special case when these two matrices are diagonal and isotropic

$$C_M = {\sigma_M}^2 I$$

and

$$C_D = {\sigma_D}^2 I$$

and, in this case, the equation of inverse theory reduces to the equations above, with

$$\sigma = \frac{\sigma_D}{\sigma_M}$$

Bayesian Interpretation

1. Justification for the Bayesian Viewpoint: Although at first the choice of the solution to this regularized problem may look artificial, and indeed the matrix Γ seems rather arbitrary, the process can be justified from a Bayesian point of view.
2. Assumptions to overcome Ill-Posedness: Note that for an ill-posed problem one must necessarily introduce some additional assumptions in order to get a unique solution.
3. Multivariate Normal Prior Probability: Statistically, the probability distribution of x is sometimes taken to be a multinomial normal distribution.
4. Assumptions for Simplifying the Analysis: For simplicity, the following assumptions are made; the means are zero; their components are independent; the components have the same standard deviation σ_x.
5. Distribution of the Data Errors: The data are also subject to errors, and the errors b are also assumed to be independent with zero and standard deviation σ_b.

6. Tikhonov Regularization as Most Likely Solution: Under these assumptions the Tikhonov-regularized solution is the most probable solution given the data and *a priori* distribution of x, according to Bayes' theorem (Vogel (2002)).
7. Assumptions of Homoscedasticity and Uncorrelatedness: If the assumption normality is replaced with the assumption of homoscedasticity and uncorrelatedness of errors, and if one still assumes zero mean, then he Gauss-Markov theorem entails that the solution is the minimal unbiased linear estimator (Amemiya (1985)).

References

- Amemiya, T. (1985): *Advanced Econometrics* **Harvard University Press** Cambridge, MA
- Fan, R. E., K. W. Chang, C. J. Hsieh, X. R. Wang, and C. J. Lin (2008): LIBLINEAR: Abounded Library for Large Linear Classification *Journal of Machine Learning Research* **8** 1871-1874
- Golub, G., M. Heath, and G. Wahba (1979): Generalized Cross-validation as a Method for Choosing a Good Ridge Parameter *Technometrics* **21 (2)** 215-223
- Gruber, M. (1998): *Improving Efficiency by Shrinkage: The James-Stein and Ridge Regression Estimators* **CRC Press** Boca Raton, FL
- Guan, N., D. Tao, Z. Luo, and B. Yo (2012): Online Non-negative Matrix Factorization with Robust Stochastic Approximation *IEEE Transactions on Neural Networks and Learning Systems* **23 (7)** 1087-1099
- Hansen, P. C., (1998): *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion 1st Edition* **SIAM** Philadelphia, PA
- Hansen, P. C. (2005): The L-curve and its Use in the Numerical Treatment of Inverse Problems
- Kennedy, P. (2003): *A Guide to Econometrics 5th Edition* **MIT Press** Cambridge, MA
- Khalaf, G., and G. Shukur (2005): Choosing Ridge Parameter for Regression Problems *Communications in Statistics – Theory and Methods* **34 (5)** 1177-1182

- Lavrentiev, M. M. (1967): *Some Improperly Posed Problems of Mathematical Physics* **Springer** New York, NY
- Ng. A. (2004): Feature Selection, L1 vs. L2 Regularization, and Rotational Invariance
- Tarantola, A. (2005): *Inverse Problem Theory and Methods for Model Parameter Estimation 1st Edition* **SIAM** Philadelphia, PA
- Vogel, C. R. (2002): *Computational Methods for Inverse Problems* **SIAM** Philadelphia, PA
- Wahba, G. (1990): *Spline Models for Observational Data* **SIAM** Philadelphia, PA
- Wikipedia (2022): Tikhonov Regularization

Lasso

Overview

1. Least Absolute Shrinkage/Selection Operator: *Lasso – least absolute shrinkage and selection operator*, also *lasso* or *LASSO* – is a regression analysis method that performs both variable selection and regularization in order to enhance the predictor accuracy and interpretability of the resulting statistical model (Wikipedia (2022)). It was originally introduced in geophysics (Santosa and Symes (1986)), and later by Tibshirani (1996), who coined the term.
2. Formulation for Linear Regression Models: Lasso was originally formulated for linear regression models. This simple case reveals a substantial amount about the estimator.
3. Ridge Regression/Best Subset Selection: These include its relationship to ridge regression and best subset selection and the connections between lasso coefficient estimate and so-called coefficients.
4. Non-unique Nature of Coefficients: It also reveals that – like standard linear regression – the coefficients do not need to be unique if covariates are collinear.
5. Extension to other Statistical Models: Though originally defined for linear regression, lasso regression is easily extended to other statistical models including generalized linear models, generalized estimating equations, proportional hazards model, and M-estimators (Tibshirani (1996, 1997)).
6. Applying Lasso to Subset Selection: Lasso' ability to perform subset selection relies on the form of constraint and has a variety of interpretations including in terms of geometry, Bayesian statistics, and convex analysis.
7. Relation to Basis Pursuit Denoising: The lasso is also related closely to basis pursuit denoising.

Basic Form – Least Squares

1. Instances of Outcome/Covariate Vectors: Consider a sample consisting of N cases, each of which consists of p covariates and a single outcome. Let y_i be the outcome and

$$x_i := (x_1, \cdots, x_p)_i^T$$

be the covariate vector for the i[th] case.

2. Optimization Setup for the Lasso: Then the objective of the lasso is to solve

$$\min_{\beta_0, \beta} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - {x_i}^T \beta)^2 \right\}$$

subject to

$$\sum_{j=1}^{p} |\beta_j| \leq t$$

(Tibshirani (1996)).

3. Coefficient Vector to be Estimated: Here β_0 is the constant coefficient

$$\beta := (\beta_1, \cdots, \beta_p)$$

is the coefficient vector, and t is a prespecified free parameter that determines the degree of regularization.

4. Compact Expression for the Optimization: Letting X be the covariate matrix, so that

$$X_{ij} = (x_i)_j$$

is the i[th] row of X, the expression can be written more compactly as

$$\min_{\beta_0, \beta} \{\|y - \beta_0 - X\beta\|_2^2\}$$

subject to

$$\|\beta\|_1 \leq t$$

where

$$\|u\|_p = \left(\sum_{i=1}^{N} |u_i|^p\right)^{\frac{1}{p}}$$

is the standard l_p norm.

5. Converting Variables to Zero-mean: Denoting the scalar mean of the data points x_i by the $\bar{x}$ and the mean of the response variables y_i and $\bar{y}$, the resulting estimate for β_0 is

$$\hat{\beta}_0 = \bar{y} - \bar{x}^T\beta$$

so that

$$y_i - \hat{\beta}_0 - {x_i}^T\beta = y_i - (\bar{y} - \bar{x}^T\beta) - {x_i}^T = (y_i - \bar{y}) - (x - \bar{x})^T\beta$$

and, therefore, it is standard to work with variables that have been zero-mean.

6. Standardization of Predictor and Response: Additionally, the covariates are typically standardized to

$$\sum_{i=1}^{N} {x_i}^2 = 1$$

so that the solution does not depend on the measurement scale.

7. Lagrangian Formulation of the Lasso: It can be helpful to rewrite

$$\underset{\beta \in \mathbb{R}^p}{min} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 \right\}$$

subject to

$$\|\beta\|_1 \leq t$$

in the so-called Lagrangian form

$$\underset{\beta \in \mathbb{R}^p}{min} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 \right\}$$

where the exact relationship between t and λ is data dependent.

Basic Form – Orthonormal Covariates

1. Basic Properties of the Lasso: Some basic properties of the Lasso estimator can now be considered. Assuming first that the covariates are orthonormal so that

$$x_i{}^T x_j = \delta_{ij}$$

where δ_{ij} is the Kronecker delta, or equivalently

$$X^T X = I$$

then using the sub-gradient methods, Tibshirani (1996) showed that

$$\hat{\beta}_j = S_{N\lambda}(\hat{\beta}_{j,OLS}) = \hat{\beta}_{j,OLS} \max\left(0, 1 - \frac{N\lambda}{|\hat{\beta}_{j,OLS}|}\right)$$

where

$$\hat{\beta}_{OLS} = [X^T X]^{-1} X^T y$$

2. S_α – the Soft Thresholding Operator: S_α is referred to as the *soft-thresholding operator*, since it translates values towards – making them exactly zero if they are small enough – instead of setting smaller values to zero and leaving larger ones untouched as the *hard thresholding operator*, often denoted H_α, would.
3. Objective of the Ridge Regression: In ridge regression the objective is to minimize

$$\underset{\beta \in \mathbb{R}^p}{min} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \right\}$$

yielding

$$\hat{\beta}_j = \frac{\hat{\beta}_{j,OLS}}{1 + N\lambda}$$

4. Shrinks Coefficients by Uniform Factor: Ridge regression shrinks all coefficients by a uniform factor $\frac{1}{1+N\lambda}$ and does not set any coefficients to zero.
5. General Solution to Ridge Regression:

$$\hat{\beta} = [X^T X + N\lambda I]^{-1} X^T y$$

6. Comparison to Best Subset Selection: It can also be compared to regression be best subset selection, in which the goal is to minimize

$$\underset{\beta \in \mathbb{R}^p}{min} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 + \lambda \|\beta\|_0 \right\}$$

where $\|\beta\|_0$ is the "l_0 norm", which is defined as

$$\|z\| = m$$

if exactly m components of z are non-zero.

7. H_α – the Hard Thresholding Function: In this case, it can be shown that

$$\hat{\beta}_j = H_{N\lambda}(\hat{\beta}_{j,OLS}) = \hat{\beta}_{j,OLS} \mathbb{I}(|\hat{\beta}_{j,OLS}| \geq \sqrt{N\lambda})$$

where H_α is the so-called hard thresholding function and $\mathbb{I}$ is an indicator function – it is 1 if its argument is true and 0 otherwise.

8. Combining Ridge and Best Subset: Therefore, the lasso estimates share features of both ridge and best subset selection regression since they both shrink the magnitude of all the coefficients, as in the best subset selection case.
9. Difference between Ridge and Lasso: Additionally, while ridge regression scales all of the coefficients by a constant factor, lasso instead translates the coefficients towards zero by a constant value and sets them to zero if they reach zero.

Correlated Covariates

1. Identical Covariates for each Observation: In one special case two covariates, say j and k, are identical for each observation, so that

$$x_{(j)} = x_{(k)}$$

where

$$x_{(j),i} = x_{(k),i}$$

Then the values of β_j and β_k that minimize the lasso objective are not uniquely determined.

2. Continuum of Valid Lasso Minimizers: In fact, if some $\hat{\beta}$ in which

$$\hat{\beta}_j \hat{\beta}_k \geq 0$$

then if

$$s \in [0, 1]$$

replacing $\hat{\beta}_j$ by $s(\hat{\beta}_j + \hat{\beta}_k)$ and $\hat{\beta}_k$ by $(1 - s)(\hat{\beta}_j + \hat{\beta}_k)$ while keeping all other $\hat{\beta}_i$ fixed, gives a new solution, so the lasso objective function then has a continuum of valid minimizers (Zou and Hastie (2005)).

3. Variants that Address this Drawback: Several variants of the lasso, including the Elastic Net regularization, have been designed to address this shortcoming.

General Form

1. Extending Lasso to other Models: Lasso regularization can be extended to other objective functions such as those for generalized linear models, generalized estimating equations, proportional hazard models, and M-estimators (Tibshirani (1996, 1997)).
2. Lasso Regularization Version of Estimator: Given the objective function

$$\frac{1}{N}\sum_{i=1}^{N} f(x_i, y_i, \alpha, \beta)$$

the lasso regularized version of the estimator s the solution to

$$\min_{\alpha,\beta} \frac{1}{N}\sum_{i=1}^{N} f(x_i, y_i, \alpha, \beta)$$

subject to

$$\|\beta\|_1 \leq t$$

where only β is penalized while α is free to take any allowed value, just as β_0 was not penalized in the basic case.

Interpretations – Geometric Interpretation

1. Lasso vs. Ridge Regression Coefficients: Lasso can set the coefficients to zero, while the superficially similar ridge regression cannot. This is due to the difference in the shape of their constraint boundaries.
2. Objective Function that is Minimized: Both lasso and ridge regression can be interpreted as minimizing the objective function

$$\min_{\beta_0, \beta} \left\{ \frac{1}{N} \|y - \beta_0 - X\beta\|_2^2 \right\}$$

but with respect to different constraints:

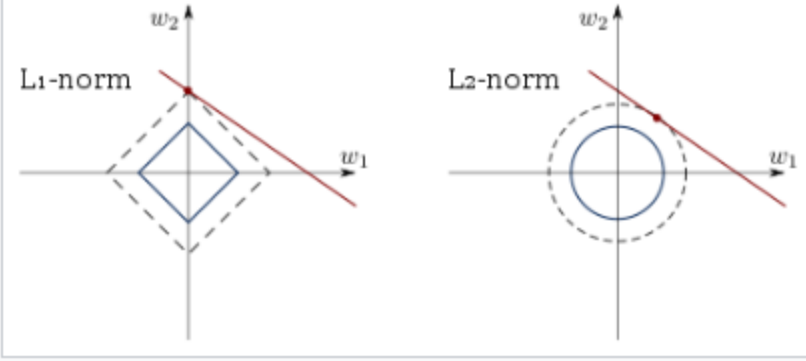
$$\|\beta\|_1 \leq t$$

for lasso and

$$\|\beta\|_2 \leq t$$

for ridge.

3. Pictorial Representation of Constraint Region:



Forms of the constraint regions for lasso and ridge regression.

The figure attached shows that the constraint region defined by the l_1 norm is a square rotated so that its corners lie on the axes – in generate a cross-polytope, while the region defined by the l_2 norm is a circle – in general an n-sphere – which is rotationally invariant and, therefore, has no corners.

4. Convex Constraint Tangential to Contour: As can be seen in the figure, a convex object that lies tangent to the boundary, such as the line shown, is likely to encounter a corner – or a higher-dimensional equivalent – of a hypercube, for which some of the components of β are identically zero, while in the case of an n-sphere, the points on the boundary for which some of the components of β are zero are not distinguished from others and the convex object is no more likely to contact a point at which some components of β are zero than one for which none of them are.

Interpretations – Making λ Easier to interpret using an Accuracy-Simplicity Tradeoff

1. Impact of the Degree of Shrinkage: The lasso can be rescaled so that it becomes easy to anticipate and influence the degree of shrinkage associate with a given value of λ.

2. X using z-scores; y-centered: It is assumed that X is standardized with z-scores and that y is centered – zero-mean.
3. OLS and Hypothesized Regression Coefficients: Let $\beta_{HYPOTHESIS}$ represent the hypothesized regression coefficients and the β_{OLS} refer to the data-optimized ordinary least-squares solutions.
4. Tradeoff between OLS and Hypothesis: One can then define the Lagrangian as a tradeoff between the in-sample accuracy of data-optimized solutions and the simplicity of sticking to the hypothesized values (Motamedi, Sanchez, Mehri, and Ghasemi (2021)). This results in

$$\underset{\beta \in \mathbb{R}^p}{min} \left\{ \frac{(y - X\beta)^T (y - X\beta)}{(y - X\beta_{HYPOTHESIS})^T (y - X\beta_{HYPOTHESIS})} + 2\lambda \sum_{i=1}^{p} \frac{|\beta_i - \beta_{HYPOTHESIS,i}|}{q_i} \right\}$$

   where q_i is specified below.
5. Balance of λ between Components: The first fraction represents relative accuracy, the second fraction relative simplicity, and λ balances between the two.
6. Definition of Relative Simplicity: Given a single regressor, relative simplicity can be defined by specifying q_i as $|\beta_{OLS} - \beta_{HYPOTHESIS}|$, which is the minimum amount of deviation from $\beta_{HYPOTHESIS}$ when

$$\lambda = 0$$

7. Solution in Terms of R^2: Assuming that

$$\beta_{HYPOTHESIS} = 0$$

   the solution path can be defined in terms of R^2:

$$\beta_{l_1} = \begin{cases} \left(1 - \frac{\lambda}{R^2}\right)\beta_{OLS} & \lambda \leq R^2 \\ 0 & \lambda > R^2 \end{cases}$$

8. $\lambda = 0$ – Solution Imposes OLS: If

$$\lambda = 0$$

the ordinary least-squares solution OLS is used. The hypothesized value of

$$\beta_{HYPOTHESIS} = 0$$

is selected of λ is greater R^2

9. λ Representing the Proportional Influence: Furthermore, if

$$R^2 = 1$$

then λ represents the proportional influence of

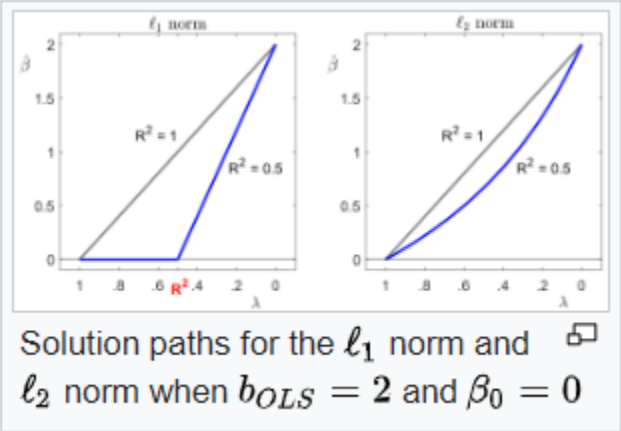
$$\beta_{HYPOTHESIS} = 0$$

In other words, $\lambda \times 100\%$ measures in percentage terms the minimal amount of influence the hypothesized value has relative to the data-optimized OLS solution.

10. l_2 norm used to Penalize Deviations: If an l_2 norm is used to penalized deviations from zero given a single regressor, the solution path is

$$\beta_{l_2} = \beta_{OLS}\left[1 + \frac{\lambda}{R^2(1-\lambda)}\right]^{-1}$$

11. β_{l_1}/β_{l_2} Dependence on λ/R^2:



Solution paths for the ℓ_1 norm and ℓ_2 norm when $b_{OLS} = 2$ and $\beta_0 = 0$

Like β_{l_1}, β_{l_2} moves in the direction of the point

$$\lambda = R^2$$

$$\beta = 0$$

when λ is close to zero; unlike β_{l_1}, the influence of R^2 diminishes in β_{l_2} if λ increases, as shown in the figure.

12. Point at which the Parameter is Activated: Given multiple regressors, the point at which the parameter is activated, i.e., allowed to deviate from $\beta_{HYPOTHESIS}$ - is also determined by a regressor's contribution to R^2 accuracy. First

$$R^2 = 1 - \frac{(y - X\beta)^T (y - X\beta)}{(y - X\beta_{HYPOTHESIS})^T (y - X\beta_{HYPOTHESIS})}$$

13. Interpretation of the R^2 Value: An R^2 of 75% means that an in-sample accuracy improves by 75% if the unrestricted OLS solutions are used instead of the hypothesized $\beta_{HYPOTHESIS}$ values.
14. Contribution to Deviation from Hypothesis: The individual contribution to deviation from each hypothesis can be computed with the $p \times p$ matrix

$$R_{\otimes} = [X^T \tilde{y}_0][X^T \tilde{y}_0]^T [X^T X]\left[\tilde{y}_0{}^T \tilde{y}_0\right]$$

where

$$\tilde{y}_0 = y - X\beta_{HYPOTHESIS}$$

15. Diagonal Elements of $R_{\otimes}/R^2$: If

$$\beta = \beta_{OLS}$$

when R^2 is computed, then the diagonal elements of $R_{\otimes}$ sum to R^2. The diagonal $R_{\otimes}$ values may be smaller than 0, or less often, larger than 1.
16. Diagonal Elements under Uncorrelated Regressors: If the regressors are uncorrelated, then the ith diagonal element of $R_{\otimes}$ simply corresponds to the R^2 value between x_i and y_i.

17. Rescaled Version of the Adaptive Lasso: A rescaled version of the adaptive lasso can be obtained by setting

$$q_{ADAPTIVE\ LASSO,i} = \left|\beta_{OLS,i} - \beta_{HYPOTHESIS,i}\right|$$

(Zou (2006)).

18. Diagonal Element Entry of $R_{\otimes}$: If regressors are uncorrelated, the point of activation of the ith parameter is given by the ith diagonal elements of $R_{\otimes}$.
19. λ – Minimal Influence of $\beta_{HYPOTHESIS}$: Assuming for convenience that $\beta_{HYPOTHESIS}$ is a vector of zeros

$$\beta_i = \begin{cases} \left[1 - \dfrac{\lambda}{R_{ii\otimes}}\right]\beta_{OLS,i} & \lambda \leq R_{ii\otimes} \\ 0 & \lambda > R_{ii\otimes} \end{cases}$$

That is, if regressors are uncorrelated, λ again specifies the minimal influence of $\beta_{HYPOTHESIS}$.

20. Parameter Activation when Regressors are Correlated: Even when regressors are correlated, the first time that a regression parameter is activated occurs when λ is equal to the highest diagonal element $R_{\otimes}$.
21. Rescaled Version of the Lasso: These results can be compared to a rescaled version of the lasso by defining

$$q_{lasso} = \frac{1}{p}\sum_{l}\left|\beta_{OLS,l} - \beta_{HYPOTHESIS,l}\right|$$

which is the absolute deviation of β_{OLS} from $\beta_{HYPOTHESIS}$.

22. Activation of the Corresponding Regressor: Assuming that the regressors are uncorrelated, the point of activation of the i[th] regressor is given by

$$\tilde{\lambda}_{lasso,i} = \frac{1}{p}\sqrt{R_{i\otimes}}\sum_{l=1}^{p}\sqrt{R_{l\otimes}}$$

23. Locating the Point of Activation: For

$$p = 1$$

the point of activation is again given by

$$\tilde{\lambda}_{lasso,i} = R^2$$

If $\beta_{HYPOTHESIS}$ is a vector of zeros, and a subset of p_B relevant properties are equally responsible for a perfect fit of

$$R^2 = 1$$

then this subset is activated at a λ value of $\frac{1}{p}$

24. Delay Introduced by Irrelevant Regressor: The point of activation of the relevant regressor the equals

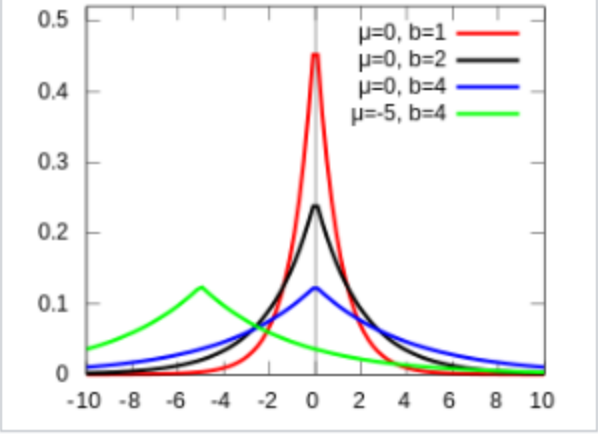
$$\frac{1}{p}\frac{1}{\sqrt{p_B}}p_B\frac{1}{\sqrt{p_B}}=\frac{1}{p}$$

In other words, the inclusion of irrelevant regressors delays the point that relevant regressors are activated by this rescaled lasso.

25. Instance of the ‘iASTc’ Estimator: The adaptive lasso and the lasso are special cases of the ‘iASTc’ estimator. This estimator only groups parameters together if the absolute correlation among regressors is larger than a user-specified value.

Interpretations – Bayesian Interpretation

1. Lasso Coefficients have Laplace Prior: Just as ridge regression can be interpreted as the linear regression for which the coefficients have been assigned normal prior distributions, lasso can be interpreted as linear regression for which the coefficients have Laplace prior distribution.
2. Laplace Sharply Peaks at Zero:



Laplace distributions are sharply peaked at their mean with more probability density concentrated there compared to a normal distribution.

The Laplace distribution peaks sharply at zero – its first derivative is discontinuous at zero – and it concentrates its probability mass close to zero than does the normal distribution.

3. Setting the Coefficients to Zero: This provides an alternate explanation of why lasso tends to set some coefficients to zero, while ridge regression does not (Tibshirani (1996)).

Interpretations - Convex Relaxation Interpretation

1. Convex Relaxation of Subset Selection: Lasso can also be viewed as a convex relaxation of the best subset selection regression problem, which is to find the subset $\leq k$ of covariates that result in the smallest value of the objective function for some fixed

$$k \leq n$$

where n is the total number of covariates.

2. l_0 norm – Nonzero Entries Count: The "l_0 norm" $\|\cdot\|_0$ – the number of nonzero entries of a vector – is the limiting case of "l_p norm", of the form

$$\|x\|_p = \left[\sum_{i=1}^{n} |x_j|^p\right]^{\frac{1}{p}}$$

Here the quotation marks signify that these are not really norms for

$$p < 1$$

since $\|\cdot\|_p$ is not convex for

$$p < 1$$

so, the triangle inequality does not hold.

3. Smallest l_p Convex in p: Therefore, since

$$p = 1$$

is the smallest value for which the "l_p norm" is convex – and therefore actually a norm – the lasso is, in some sense, the best convex approximation to the best subset selection problem, since the region defined by

$$\|x\|_1 \leq t$$

that is convex hull of the region defined by

$$\|x\|_p \leq t$$

for

$$p < 1$$

Generalizations

1. Remedy Limitations of Original Technique: Lasso variants have been created in order remedy limitations of the original technique and to make method useful for particular problems. Almost all of these focus on respecting or exploiting the dependencies among covariates.
2. Additional Ridge-Regression Like Penalty: Elastic-net regularization adds an additional ridge-regression like penalty that improves performance when the number of predictors is larger than the sample size, allows the method to select strongly correlated variables together, and improves overall prediction accuracy (Zou and Hastie (2005)).
3. Single Unit of Related Covariates: Group lasso allows group of related covariates to be selected as a single unit, which can be useful in settings where it does not make sense to include some covariates without others (Yuan and Li (2006)).

4. Sparse and Overlap Group Lasso: Further extensions of the group lasso perform variable selection within groups – sparse group lasso – and allow overlap between groups, the overlap group lasso (Puig, Wiesel and Hero III (2009), Jacob, Obozinski, and Vert (2009)).
5. Spatial or Temporal Problem Characteristics: Fused lasso can account for spatial or temporal characteristics of a problem, resulting in estimates that better match the system structure (Tibshirani, Saunders, Rosset, Zhu, and Knight (2005)).
6. Fitting the Lasso-regularized Models: Lasso-regularized models can be fit using techniques including sub-gradient methods, least-angle regression (LARS), and proximal gradient methods.
7. Optimal Value for the Regularization Parameter: Determining the optimal value for the regularization parameter is an important part of ensuring the model performs well; it is typically chosen using cross-validation.

Generalizations – Elastic Net

1. More Covariates than Sample Size: When

$$p > n$$

   i.e., the number of covariates is greater than the sample size, the lasso can only select n covariates, even when more are associated with the outcome, and it tends to select one covariate from a set of highly correlated covariates (Zou and Hastie (2005)).
2. Bigger Sample Size than Covariates: Additionally, even when

$$n > p$$

ridge regression tends to perform better given strongly correlated covariates.

3. Formulation of the Elastic Net: The elastic net extends the lasso by adding an additional l_2 penalty term giving

$$\underset{\beta \in \mathbb{R}^p}{min} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 + \lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2^2 \right\}$$

which is equivalent to solving

$$\underset{\beta \in \mathbb{R}^p}{min} \left\{ \frac{1}{N} \|y - \beta_0 - X\beta\|_2^2 \right\}$$

subject to

$$\alpha \|\beta\|_1 + (1 - \alpha) \|\beta\|_2^2 \leq t$$

where

$$\alpha = \frac{\lambda_2}{\lambda_1 + \lambda_2}$$

4. Lasso Form of the Formulation: The above can be written in a simple lasso form as

$$\underset{\beta^* \in \mathbb{R}^p}{min} \left\{ \frac{1}{N} \|y^* - X^*\beta^*\|_2^2 + \lambda^* \|\beta^*\|_1 \right\}$$

letting

$$X^*_{p\times(n+p)} = \frac{1}{\sqrt{1+\lambda_2}} \begin{pmatrix} X \\ \sqrt{\lambda_2} I_{p\times p} \end{pmatrix}$$

$$y^*_{(n+p)} = \begin{pmatrix} y \\ 0^p \end{pmatrix}$$

$$\lambda^* = \frac{\lambda_1}{\sqrt{1+\lambda_2}}$$

$$\beta^* = \beta\sqrt{1+\lambda_2}$$

5. <u>Situation where Covariates are Orthogonal</u>: Here

$$\hat{\beta} = \frac{\hat{\beta}^*}{\sqrt{1+\lambda_2}}$$

which, when the covariates are orthogonal to each other, gives

$$\hat{\beta}_j = \frac{\hat{\beta}^*_{j,OLS}}{\sqrt{1+\lambda_2}} \max\left(0, 1 - \frac{\lambda^*}{|\hat{\beta}^*_{j,OLS}|}\right) = \frac{\hat{\beta}^*_{j,OLS}}{1+\lambda_2} \max\left(0, 1 - \frac{\lambda}{|\hat{\beta}^*_{j,OLS}|}\right) = \frac{\hat{\beta}^*_{j,Lasso}}{1+\lambda_2}$$

6. Combination of Ridge and Lasso: Thus, the result of the elastic net penalty is a combination of the effects of the ridge and the lasso penalties.
7. Convex Nature of the Penalty: Returning to the general case, the fact that the penalty function is strictly convex means that if

$$x_{(j)} = x_{(k)}$$

then

$$\hat{\beta}_j = \hat{\beta}_k$$

which is a change from lasso (Zou and Hastie (2005)).

8. Relation between any Two $\hat{\beta}_j$'s: In general, if

$$\hat{\beta}_j \hat{\beta}_k > 0$$

$$\frac{|\hat{\beta}_j - \hat{\beta}_k|}{\|y\|} \leq \frac{\sqrt{2(1-\rho_{jk})}}{\lambda_2}$$

where

$$\rho = X^T X$$

is the sample correlation matrix because the x's are normalized.

9. Regression Coefficients of Correlated Covariates: Therefore, highly correlated covariates tend to have similar regression coefficients, with the degree of similarity depending on both $\|y\|_1$ and λ_2, which is different from lasso.
10. Grouping Effect of Correlated Variables: This phenomenon, in which strongly correlated covariates have similar regression coefficients, is referred to as the group effect.
11. Importance of Locating Group Covariates: Grouping is desirable since, in applications such as tying genes to a disease, finding all the associated covariates is preferable, rather than selecting one from each set of correlated covariates, as lass often does (Zou and Hastie (2005)). In addition, selecting only one is each group results in increasing prediction error, since the model is less robust, which is why ridge regression often outperforms lasso.

Generalizations – Group Lasso

1. Selecting Predefined Groups of Covariates: Yuan and Lin (2006) introduced the group lasso to allow predefined groups of covariates to jointly be selected into or out of a model.
2. Categorical Variables Coded as a Collection: This is useful in many settings, perhaps most obvious when a categorical variable is coded as a collection of binary covariates. In this case, group lasso can ensure that all the variables encoding the categorical covariate are included or excluded together.
3. Natural Grouping in Biological Studies: Another setting in which grouping is natural is in biological studies. Since genes and proteins lie in known pathways, which pathways are related to an outcome may be more significant than whether individual genes are.
4. Natural Generalization of Standard Lasso: The objective function for the group lasso is a natural generalization of the standard lasso objective

$$\min_{\beta \in \mathbb{R}^p} \left\{ \left\| y - \sum_{j=1}^{J} X_j \beta_j \right\|_2^2 + \lambda \sum_{j=1}^{J} \|\beta_j\|_{K_j} \right\}$$

$$\|z\|_{K_j} = \sqrt{z^T K_j z}$$

where the design matrix X and the covariate matrix β have been replaced by a collection of design matrices X_j and covariate vectors β_j, one for each of the J groups.

5. Penalty - Sum over l_2-norm: Additionally, the penalty term is now a sum over l_2 norms defined by the positive definite matrix K_j.
6. Reduction to Standard Lasso/Ridge: If each covariate is in its own group and

$$K_j = I$$

then this reduces to the standard lasso, while if there is only a single group and

$$K_1 = I$$

this reduces to ridge regression.

7. l_2 norm on the Subspaces: Since the penalty reduces to an l_2 norm on the subspaces defined by each group, it cannot select out only some of the covariates from the group, just as the ridge regression cannot.
8. Sum over different Subspace Norms: However, since the penalty is the sum over different subspace norms, as in the standard lasso, the constraint has some non-differentiable points, which correspond to some subspaces being identically zero.

9. Zeroed and Shrunk Coefficient Vectors: Therefore, the coefficient vectors corresponding to some subspaces can be set to zero, while the others can be shrunk.
10. Selecting Sparse Covariates in the Group: However, it is possible to extend the group lasso to the so-called sparse group lasso, which can select individual covariates within a group, by adding an l_1 penalty to each group subspace (Puig, Wiesel, and Hero III (2009)).
11. Group Lasso with Covariate Overlap: Another extension, the group lasso with overlap, allows covariates to be shared across groups, e.g., if a gene were to occur in two pathways (Jacob, Obozinski, and Vert (2009)).

Generalizations – Fused Lasso

1. Covariates with Spatial/Temporal Structures: In some cases, the phenomenon under study may have spatial or temporal structures that must be considered during the analysis, such as time-series or image-based data. Tibshirani, Saunders, Rosset, Zhu, and Knight (2005) introduced the fused lasso to extend to this type of data.
2. Formulation of the Fused Lasso Objective: The fused lasso objective is

$$\min_{\beta} \left\{ \frac{1}{N} \sum_{i=1}^{N} (y_i - {x_i}^T \beta)^2 \right\}$$

subject to

$$\sum_{i=1}^{p} |\beta_j| \leq t_1$$

and

$$\sum_{j=2}^{p} |\beta_j - \beta_{j-1}| \leq t_2$$

3. Lasso Constraint Reflecting Spatial/Temporal Structure: The first constraint is the lasso constraint, while the second directly penalized large changes with respect to spatial or temporal structure, which forces the coefficients to vary smoothly to reflect the system's underlying logic.
4. Clustered Lasso - Variate Series Grouping: Clustered Lass (She (2010)) is a generalization of the fused lasso that identifies and groups relevant covariates based on their effects, i.e., coefficients. The basic idea is to penalize the difference between the coefficients so that non-zero ones cluster. This can be modeled using the following regularization:

$$\sum_{i<j}^{p} |\beta_i - \beta_j| \leq t_2$$

5. Clustering into Highly Correlated Groups: In contrast, variables can be clustered into highly correlated groups, and then a single representative covariate can be extracted from each cluster (Reid (2015)).
6. Algorithms for Solving Fused Lasso: Algorithms exist that solve the fused lass problem, and some generalizations of it. Algorithms can solve it exactly in a finite number of operations (Bento (2018)).

Generalizations – Quasi-norms and Bridge Regression

1. Penalty Norm for Lasso Variants: Lasso, elastic net, and fused lasso construct the penalty functions from the l_1 and the l_2 norms – with weights if necessary.
2. Penalty Norm for Bridge Regression: The bridge regression uses l_p norms $p \geq 1$ and quasi-norms

$$0 < p < 1$$

   (Fu (1998)).
3. Optimizer Formulation for Bridge Regression: For example, for

$$p = \frac{1}{2}$$

   the analogue of the lasso objective in the Lagrangian form is to solve

$$\underset{\beta \in \mathbb{R}^p}{min} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 + \lambda \sqrt{\|\beta\|_{\frac{1}{2}}} \right\}$$

   where

$$\|\beta\|_{\frac{1}{2}} = \left(\sum_{j=1}^{p} \sqrt{|\beta_j|} \right)^2$$

4. Value of the Quasi-norm: It is claimed that the fractional quasi-norms l_p for

$$0 < p < 1$$

provide more meaningful results in the data analysis both theoretically and practically (Aggarwal, Hinneburg, and Keim (2001)).

5. Expectation Minimization for Convex Constraint: The non-convexity of these quasi-norms complicates the optimization problem. To solve this problem, an expectation-minimization procedure is developed (Gorban, Mirkes, and Zinovyev (2016)) and implemented (Mirkes (2018)) for minimization of the function

$$\underset{\beta \in \mathbb{R}^p}{min} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 + \lambda \sum_{j=1}^{p} \varsigma\left(\beta_j{}^2\right) \right\}$$

where $\varsigma(\gamma)$ is an arbitrary concave monotonically increasing function – for example

$$\varsigma(\gamma) = \sqrt{\gamma}$$

gives the lasso penalty and

$$\varsigma(\gamma) = \gamma^{\frac{1}{4}}$$

gives the $l_{\frac{1}{2}}$ penalty.

6. Quadratic Approximation of Sub-quadratic Growth: The efficient algorithm for minimization is based on piece-wise quadratic approximation of sub-quadratic growth PQSQ (Mirkes (2018)).

Generalizations – Adaptive Lasso

The adaptive Lasso was introduced by Zou (2006) for linear regression and by Zhang and Lu (2007) for proportional hazards regression.

Generalizations – Prior Lasso

1. Incorporation of Prior Information: The prior lasso was introduced for generalized linear models by Jiang (2016) to incorporate prior information, such as the importance of certain covariates.
2. Prior Lasso using Pseudo Responses: In prior lasso, such information is summarized into pseudo responses called prior responses $\hat{y}_p$, and then an additional criterion function is added to the usual objective function with a lasso penalty.
3. Formulation of the Prior Lasso Optimizer: Without loss of generality, in linear regression, the new objective function can be written as

$$\underset{\beta \in \mathbb{R}^p}{min} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 + \frac{\eta}{N} \left\|\hat{y}_p - X\beta\right\|_2^2 + \lambda \|\beta\|_1 \right\}$$

which is equivalent to

$$\underset{\beta \in \mathbb{R}^p}{min} \left\{ \frac{1}{N} \|\tilde{y} - X\beta\|_2^2 + \frac{\lambda}{1+\eta} \|\beta\|_1 \right\}$$

the usual lasso objective function with responses y being replaced by a weighted response of the observes responses and the prior responses

$$\tilde{y} = \frac{y + \eta \hat{y}_p}{1+\eta}$$

the adjusted response values by the prior information.

4. Balance between Prior and Data: In prior lasso, the parameter η is called the balancing parameter, in that it balances the relative importance of the data and the prior information.
5. The $\eta = 0$ and $\eta = \infty$ Cases: In the extreme case of

$$\eta = 0$$

the prior lasso is reduced to a lasso. If

$$\eta = \infty$$

the prior lasso will rely solely on the prior information to fit the model.

6. Controlling the Coefficient's Prior Variance: Furthermore, the balancing parameter η has another appealing interpretation; it controls the variance of β in its prior distribution from a Bayesian viewpoint.

7. Scenarios where Prior Lass Shines: Prior lasso is more efficient in parameter estimation and prediction – with a smaller estimation error and prediction error – when the prior information is of high quality, and is robust to low quality prior information with a good choice of the balancing parameter η.

Computing Lasso Solutions

1. Techniques for Calibrating the Lasso: The loss function of the lasso is not differentiable, but a wide variety of techniques from convex analysis and optimization theory have been developed to compute the solutions path of the lasso.
2. Coordinate Descent, Proximal/Sub-gradient, LARS: These include coordinate descent (Friedman, Hastie, and Tibshirani (2010)), sub-gradient methods, least angle regression (LARS), and proximal descent (Efron, Hastie, Johnstone, and Tibshirani (2004)).
3. Generalization of Traditional Gradient Methods: Sub-gradient methods are a natural generalization of the traditional methods such as gradient descent and stochastic gradient descent to the case in which the objective function is not differentiable at all points.
4. LARS - Scheme Customized for Lasso: LARS is a method that is closely tied to the lasso models and in many cases allow them to be efficiently, though it may not perform well in all circumstances. LARS generates complete solution paths (Efron, Hastie, Johnstone, and Tibshirani (2004)).
5. Flexibility/Performance of Proximal Methods: Proximal methods have become popular because of their flexibility and performance and are an active area of research. The choice of the method will depend on the particular lasso variant, the data, and the available resources. However, the proximal methods generally perform well.

Choice of Regularization Parameter

1. Benefits of the Appropriate λ: Choosing the regularization parameter λ is a fundamental part of the lasso. An appropriate value is essential to the performance of the lasso since it controls the strength of the shrinkage and variable selection, which, in moderation, can improve both prediction accuracy and interpretability.
2. Drawbacks of an Ill-suited λ: However, if the regularization becomes too strong, important variables may be omitted and the coefficients may be shrunk excessively, which can harm both predictive capacity and inferencing.
3. Validation using AIC and BIC: Information criteria such as the Bayesian Information Criterion BIC and Akaike Information Criterion AIC may be preferable to cross-validation because they are faster to compute and their performance is less volatile in small samples.
4. Setting Regularization with Information Criterion: An information criterion selects the estimator's regularization parameter by maximizing a model's in-sample accuracy while penalizing its effective degrees of freedom/number of parameters.
5. Measuring Effective Degrees of Freedom: Zou, Hastie, and Tibshirani (2007) proposed degrees of freedom by counting the number of parameters that deviate from zero.
6. λ vs. Degrees of Freedom: The degrees-of-freedom approach was considered flawed by Kaufman and Rosset (2014) and Janson, Fithian, and Hastie (2015) because a model's degrees of freedom might be increased even when it is penalized harder by the regularization parameter.
7. Alternative Metric – Relative Simplicity Measure: As an alternative, the relative simplicity measure defined earlier can be used to count the effective number of parameters. For the lasso, this measure is given by

$$\hat{\mathcal{P}} = \sum_{i=1}^{p} \frac{\left|\beta_{OLS,i} - \beta_{HYPOTHESIS,i}\right|}{\frac{1}{p}\sum_{l}\left|\beta_{OLS,l} - \beta_{HYPOTHESIS,l}\right|}$$

Selected Applications

LASSO has been applied in economics and finance, and was found to improve prediction and to select sometimes neglected variables, for example in corporate bankruptcy prediction literature (Shaonan, Yu, and Guo (2015)), or high-growth firms prediction (Coad and Srhoj (2020)).

References

- Aggarwal, C. C., Hinneburg, A., and D. A. Keim (1998): On the Surprising Behavior of Distance Metrics in High-dimensional Space
- Bento, J. (2018): On the Complexity of the Weighted Fused Lasso *IEEE Letters in Signal Processing* **25 (10)** 1595-1599
- Coad, A., S. Srhoj (2020): Catching Gazelles with a Lasso: Big Data Techniques for the Prediction of High-growth Firms *Small Business Economics* **55 (1)** 541-565
- Efron, B., T. Hastie, I. Johnstone, and R. Tibshirani (2004): Least Angle Regression *The Annals of Statistics* **32 (2)** 407-499
- Friedman, J., T. Hastie, and R. Tibshirani (2010): Regularization Paths of Generalized Linear Models via Coordinate Descent *Journal of Statistical Software* **33 (1)** 1-22
- Fu, W. J. (1998): The Bridge versus the Lasso *Journal of Computational and Graphical Statistics* **7 (3)** 397-416
- Gorban, A. N., E. M. Mirkes, and A. Zinovyev (2016): Piece-wise Quadratic Approximations of Arbitrary Error Functions for Fast and Robust Machine Learning
- Jacob, L., G. Obozinski, and J. P. Vert (2009): Group Lasso with Overlap and Graph Lasso
- Janson, L., W. Fithian, T. J. Hastie (2015): Effective Degrees of Freedom: A Flawed Metaphor *Biometrika* **102 (2)** 479-485

- Jiang, Y. (2016): Variable Selection with Prior Information for Generalized Linear Models via the Prior Lasso Method *Journal of the American Statistical Association* **111 (513)** 355-376
- Kaufman, S. and S. Rosset (2014): When does more Regularization imply fewer Degrees of Freedom? Sufficient Conditions and Counter-examples *Biometrika* **101 (4)** 771-784
- Mirkes, E. M. (2018): PQSQ-based Regularization Method
- Motamedi, F., H. Sanchez, A. Mehri, F. Ghasemi (2021): Adding Big-data Analysis through LASSO-Random Forest Algorithm in QSAR Studies *Bioinformatics* **37 (19)** 469-475
- Puig, A. T., A. Wiesel, and A. O. Hero III (2009): A Multidimensional Shrinkage-Thresholding Operator
- Reis, S. (2015): Sparse Regression and Marginal Testing under Cluster Prototypes *Biostatistics* **17 (2)** 364-376
- Santosa, F., and W. W. Symes (1986): Linear Inversion of Band-limited Seismograms *SIAM Journal on Scientific and Statistical Computing* **7 (4)** 1307-1330
- Shaonan, T., Y. Yan, H. Gui (2015): Variable Selection and Corporate Bankruptcy Forecasts *Journal of Baking and Finance* **52 (1)** 89-100
- She, Y. (2010): Sparse Regression with Exact Clustering *Electronic Journal of Statistics* **4** 1055-1096
- Tibshirani, R. (1996): Regression Shrinkage and Selection with the Lasso *Journal of the Royal Statistical Society B* **58 (1)** 267-288
- Tibshirani, R. (1997): The Lasso Method for Variable Selection in the Cox Model *Statistics in Medicine* **16 (4)** 385-395
- Tibshirani, R., M. Saunders, S. Rosset, J. Zhu, and K. Knight (2005): Sparsity and Smoothness via the Fused Lasso *Journal of the Royal Statistical Society B* **67 (1)** 91-108
- Wikipedia (2022): Lasso
- Yuan, M., and Y. Lin (2006): Model Selection and Estimation and Regression with Grouped Variables *Journal of the Royal Statistical Society B* **68 (1)** 49-67
- Zhang, H. H., and W. Lu (2007): Adaptive Lasso for Cox's Proportional Hazards Model *Biometrika* **94 (3)** 691-703

- Zou, H. and T. Hastie (2005): Regularization and Variable Selection vis the Elastic Net *Journal of the Royal Statistical Society B* **67 (2)** 301-320
- Zou, H. (2006): The Adaptive Lasso and its Oracle Properties *Journal of the American Statistical Association* **101 (476)** 1418-1429
- Zou, H., T. Hastie, and R. Tibshirani (2007): On the 'Degrees of Freedom' of the Lasso *The Annals of Statistics* **35 (5)** 2173-2192

Regularization Paths for Generalized Linear Models via Coordinate Descent

Abstract

1. Algorithms for Generalized Linear Models: Friedman, Hastie, and Tibshirani (2010) develop fast algorithms for estimation of generalized linear models with penalties.
2. Linear, Logistic, and Multinomial Regression: The models include linear regression, two-class logistic regression, and multinomial regression problems while the penalties include l_1 – the lasso, the l_2 – ridge regression, and the mixtures of the two – the elastic net.
3. Coordinate Descent along Regularization Path: The algorithms use cyclical coordinate descent, computed along a regularization path.
4. Large Problems with Sparse Features: The methods can handle large problems and can also deal efficiently with sparse features.
5. Comparison with other Competing Methods: In comparative timings, they find that the new algorithms are algorithms are considerably faster than competing methods.

Introduction

1. Lasso – Popular Method of Regression: The lasso (Tibshirani (1996)) is a popular method for regression that uses an l_1 penalty to achieve a sparse solution. In the signal processing literature, the lasso is known as *basis pursuit* (Chen, Donoho, and Saunders (1998)).

2. GLM and Proportional Hazard Models: This idea has been broadly applied, for example, to generalized linear models (Tibshirani (1996)) and Cox's proportional hazard model for survival data (Tibshirani (1997)). In the subsequent years, there has been an enormous amount of research activity denoted to related regularization methods.
3. Grouped Lasso: Here, variables are included or excluded in groups (Yuan and Lin (2007), Meier, van de Geer, and Buhlmann (2008)).
4. Dantzig Selector: This is a slightly modified version of the lasso (Candes and Tao (2007)).
5. Elastic Net: The elastic net was developed by Zou and Hastie (2005) for correlated variables, and uses a penalty that is part l_1 and part l_2.
6. l_1 Regularization for GLM: Park and Hastie (2007).
7. Methods that use Non-concave Penalties: These include methods such as SCAD (Fan and Li (2005)), and generalized elastic net (Friedman (2008)), and they enforce more serious variable selection than the lasso.
8. Regularization Paths for the SVM: Hastie, Rosset, Tibshirani, and Zhu (2004).
9. Graphical Lasso: This was developed by Friedman, Hastie, Hoefling, and Tibshirani (2007) for sparse covariance matrix and undirected graphs.
10. Algorithm for Computing Regularization Path: Efron, Hastie, Johnstone, and Tibshirani (2004) developed an efficient algorithm to compute the entire regularization path for the lasso for linear regression models.
11. Coefficients that are Piece-wise Linear: Their algorithm exploits the fact that the coefficient profiles are piece-wise linear, which leads to an algorithm with the same computational cost as the full least-squares fit on the data (Osborne, Presnell, and Turlach (2000)).
12. Exploitation of the Piece-wise Linearity: In some of the extensions above – Dantzig selector, elastic net, and regularization paths for SVM – piece-wise linearity can be exploited as in Efron, Hastie, Johnstone, and Tibshirani (2004) to yield efficient algorithms.
13. Piece-wise Linearity Class of Problems: Rosset and Zhu (2007) characterize the class of problems where piece-wise linearity exists – both the loss function and the penalty have to be quadratic or piece-wise linear.
14. Focus on Coordinate Descent Methods: In their paper, Friedman, Hastie, and Tibshirani (2010) focus instead on the cyclical coordinate descent methods. These methods have been proposed for the lasso a number of times, but only then was their power fully realized.

15. Prior Work in this Area: Early references include Fu (1998), Shevade and Keerthi (2003), and Daubechies, Defrise, and De Mol (2004). Van der Kooij (2007) independently used coordinate descent for solving elastic-net penalized regressions models.
16. Regularization Parameters along Entire Path: Further discoveries include Friedman, Hastie, Hoefling, and Tibshirani (2007) and Wu and Lange (2008). The first paper recognized the value of solving the problem along an entire path for the regularization parameters, using the current estimates as *warm starts*. This strategy turns out to be remarkably efficient for this problem.
17. Other Works on Coordinate Descent: Several other researchers have also re-discovered coordinate descent, many for solving the same problems extend the work of Friedman, Hastie, and Tibshirani (2010) address – notably Shevade and Keerthi (2003), Krishnapuram and Hartemink (2005), Genkin, Lewis, and Madigan (2007), and Wu, Chen, Hastie, Sobel, and Lange (2009).
18. Fast Algorithms for Fitting GLMs: Friedman, Hastie, and Tibshirani (2010) extend the work of Friedman, Hastie, Hoefling, and Tibshirani (2007) and develop fast algorithms for fitting generalized linear methods with elastic net penalties. In particular, their models include regression, two-class logistic regression, and multinomial regression problems.
19. Feature Sparsity and Large Datasets: Their algorithms can work on very large datasets, and can take advantage of the sparsity in the feature set.
20. glmnet - Publicly Available R Package: They provide a publicly available package glmnet (Friedman, Hastie, and Tibshirani (2009)) implemented in the R programming system.
21. Convergence Properties of Coordinate Descent: They do not revisit the well-established convergence properties of coordinate descent in convex problems (Tseng (2001)), however.
22. Domains with Very Large Datasets: Lasso formulation is frequently used in domains with very large datasets, such as genomics and web analysis. Consequently, a focus of Friedman, Hastie, and Tibshirani (2010)'s research has been algorithmic efficiency and speed. They demonstrate through simulations that their procedures outperform all competitors – even those based on coordinate descent.
23. Algorithm for the Elastic Net: The next section presents the algorithm for elastic net, which includes lasso and ridge regression as special cases.

24. Two Class/Multinomial Logistic Regression: The next two sections discuss two-class logistic regression and multinomial logistic regression. Comparative timings are presented in the subsequent section.
25. Regularization Paths for Limited GLMs: Although the title of the chapter advertises regularization paths for GLMs, it only covers three important members of this family. However, exactly the same technology extends trivially to other members of the exponential family, such as the Poisson model. Friedman, Hastie, and Tibshirani (2010) plan to extend their software to cover these important other cases, as well as the Cox model for survival data.
26. Algorithms for Fitting Model Families: Note that this chapter is about algorithms for fitting particular families of models, and not about the statistical properties of the models themselves.

Algorithms for the Lasso, Ridge Regression, and Elastic Net

1. Setup for Lasso Regression Formulation: Consider the usual setup for linear regression. Given a response variable

$$Y \in \mathbb{R}$$

and a predictor vector

$$X \in \mathbb{R}^p$$

the regression function is approximated by a linear model

$$\mathbb{E}[Y|X = x] = \beta_0 + x^T\beta$$

2. Observation Pairs with Standardized Variates: There are N observation pairs (x_i, y_i). For simplicity, it is assumed that x_{ij} are standardized:

$$\sum_{i=1}^{N} x_{ij} = 0$$

$$\frac{1}{N}\sum_{i=1}^{N} x_{ij}^2 = 1$$

for

$$j = 1, \dots, p$$

The algorithms generalize naturally to the unstandardized case.

3. Formulation of Elastic Net Problem: The elastic net solves the following problem

$$\underset{(\beta_0, \beta \in \mathbb{R}^{p+1})}{min} R_\lambda(\beta_0, \beta) = \underset{(\beta_0, \beta \in \mathbb{R}^{p+1})}{min} \left[\frac{1}{2N}\sum_{i=1}^{N}(y_i - \beta_0 - x^T\beta)^2 + \lambda P_\alpha(\beta)\right]$$

where

$$P_\alpha(\beta) = (1-\alpha)\frac{1}{2}\|\beta\|_{l_2}{}^2 + \alpha\|\beta\|_{l_1} = \sum_{j=1}^{p}\left[\frac{1}{2}(1-\alpha)\beta_j{}^2 + \alpha\left|\beta_j\right|\right]$$

4. α-based Elastic Net Penalty: P_α is the *elastic net penalty* (Zou and Hastie (2005)), and is a compromise between the ridge regression penalty

$$\alpha = 0$$

and the lasso penalty

$$\alpha = 1$$

5. Situation where Penalty is Impactful: This penalty is particularly in the useful in the

$$p \gg N$$

scenario, or any situation where there are many correlated predictor variables.

6. Scaled vs Unscaled Elastic Net: Zou and Hastie (2005) called this penalty *naïve* elastic net, and preferred a rescaled version which they called elastic net. This distinction is dropped here.
7. Shrinking Coefficients Towards each Other: Ridge regression is known to shrink the coefficients of correlated predictors towards each other.
8. Extreme Case - p Identical Predictors: In the extreme case of k identical predictors, they each get identical coefficients with $\frac{1}{k}$th the size any single one would get if fit alone.

9. Bayesian View of Ridge Penalty: From a Bayesian point of view, the ridge penalty is ideal if there are many predictors, and all have non-zero coefficients drawn from a Gaussian distribution.
10. Lasso Indifference to High Correlation: Lasso, on the other hand, is somewhat indifferent to very correlated predictors, and to pick one and ignore the rest. In the extreme case above, the lasso problem breaks down.
11. Laplace Prior of Lasso Penalty: The lasso penalty corresponds to a Laplace prior, which expects many coefficients to be close to zero, and a small subset to be larger and non-zero.
12. Removal of Extreme Correlation Degeneracy: The elastic net with

$$\alpha = 1 - \epsilon$$

for some small

$$\epsilon > 0$$

performs much like the lasso, but removes any degeneracies and wild behavior caused by extreme correlations.
13. Tradeoff between Ridge and Lasso: More generally, the entire family P_α creates a useful compromise between the bridge and the lasso. As α increases from 0 to 1, and given a λ, the sparsity of the solution to

$$\min_{(\beta_0, \beta \in \mathbb{R}^{p+1})} R_\lambda(\beta_0, \beta) = \min_{(\beta_0, \beta \in \mathbb{R}^{p+1})} \left[\frac{1}{2N} \sum_{i=1}^{N} (y_i - \beta_0 - x^T \beta)^2 + \lambda P_\alpha(\beta) \right]$$

i.e., the number of coefficients equal to zero, increases monotonically from 0 to the sparsity of the lasso solution.

14. Impact of Varying the α:

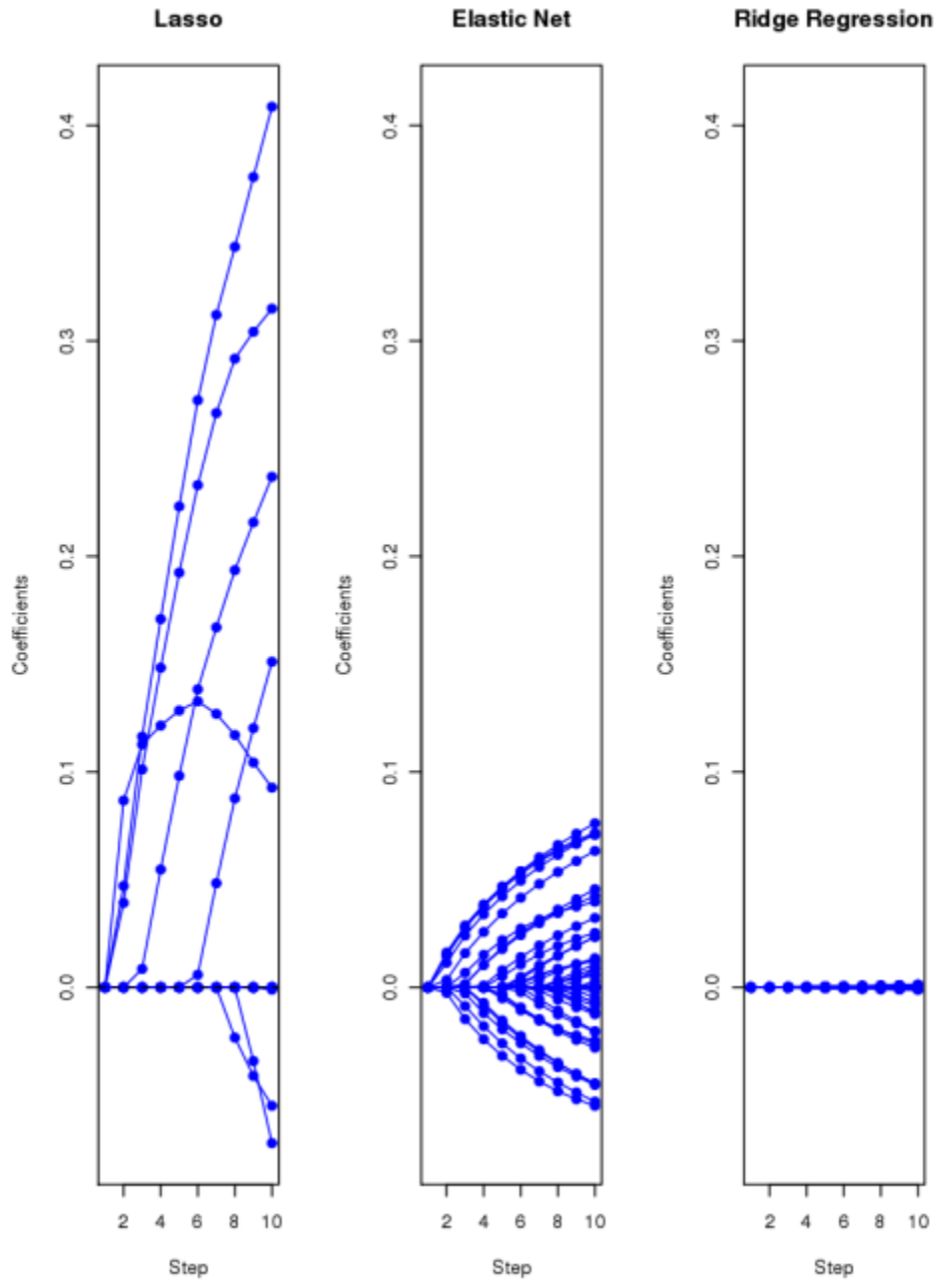


Figure 1: `Leukemia` data: profiles of estimated coefficients for three methods, showing only first 10 steps (values for λ) in each case. For the elastic net, $\alpha = 0.2$.

The attached figure shows an example that demonstrates the effect of varying α. The dataset is from Golub, Slonim, Tamayo, Huard, Gaasenbeek, Mesirov, Coller, Loh, Downing, Caligiuri, Bloomfield, and Lander (1999), consisting of 72 observations on 3571 genes measured with DNA micro-arrays.

15. Two Classes of the Datasets: The observations fall in two classes, so penalties are used in conjunction with the logistic regression models of the later section. The coefficient profiles from the first 10 steps – grid of λ values – for each of the three regularization methods are shown.
16. Lasso vs Ridge Coefficients Generated: The lasso penalty admits at most

$$N = 72$$

genes into their model, while the ridge regression gives all 3571 genes non-zero coefficients.

17. Compromise Provided by Elastic Net: The elastic net penalty provides a compromise between the two, and has the effect of averaging genes that are highly correlated and then entering the averaged genes into the model.
18. Computation of the Entire Path: Using the algorithm described below, computation of the entire path of solutions for each method, at 100 values of the regularization parameters evenly spaced on the log-scale, took under a second in total.
19. Large Number of Non-zero Coefficients: Because of the large number of coefficients for ridge penalty, they are individually much smaller than the coefficients for other metrics.
20. Coordinate Descent for Elastic Net: Consider a coordinate descent step for solving

$$\min_{(\beta_0, \beta \in \mathbb{R}^{p+1})} R_\lambda(\beta_0, \beta) = \min_{(\beta_0, \beta \in \mathbb{R}^{p+1})} \left[\frac{1}{2N} \sum_{i=1}^{N} (y_i - \beta_0 - x^T\beta)^2 + \lambda P_\alpha(\beta) \right]$$

That is, suppose there are estimates $\tilde{\beta}_0$ and $\tilde{\beta}_l$ for

$$l \neq j$$

and one wishes to partially optimize with respect to β_j

21. Computing the Gradient at $\tilde{\beta}_j$: One would like to compute the gradient at

$$\beta_j = \tilde{\beta}_j$$

which only exists if

$$\tilde{\beta}_j \neq 0$$

If

$$\beta_j > 0$$

then

$$\left.\frac{\partial R_\lambda(\beta_0, \beta)}{\partial \beta_j}\right|_{\beta=\tilde{\beta}} = -\frac{1}{N}\sum_{i=1}^{N} x_{ij}\left(y_i - \tilde{\beta}_0 - x^T\tilde{\beta}\right) + \lambda(1-\alpha)\beta_j + \lambda\alpha$$

22. Form of the Coordinate Update: A similar expression exists if

$$\tilde{\beta}_j < 0$$

$$\tilde{\beta}_j = 0$$

is treated separately. Simple calculus (Donoho and Johnstone (1994)) shows that the coordinate-wise update has the form

$$\tilde{\beta}_j \leftarrow \frac{S\left(\frac{1}{N}\sum_{i=1}^{j} x_{ij}\left[y_i - \tilde{y}_i^{(j)}\right], \lambda\alpha\right)}{1 + \lambda(1-\alpha)}$$

23. Fitted Contributions Exclusive of x_{ij}:

$$\tilde{y}_i^{(j)} = \tilde{\beta}_0 + \sum_{l \neq j} x_{il}\tilde{\beta}_l$$

is the fitted value excluding the contributions from x_{ij}, and hence $y_i - \tilde{y}_i^{(j)}$ the *partial residual* for fitting β_j. Because of the standardization

$$\frac{1}{N}\sum_{i=1}^{j} x_{ij}\left[y_i - \tilde{y}_i^{(j)}\right]$$

is the simple least-squares coefficient when fitting this product residual to x_{ij}.

24. Soft-Thresholding Operator: $S(z,\gamma)$ is the soft-thresholding operator with the value

$$sign(z)(|z|-\gamma)_+ = \begin{cases} z-\gamma & if\ z>0\ and\ \gamma<|z| \\ z+\gamma & if\ z<0\ and\ \gamma<|z| \\ 0 & if\ \gamma \geq |z| \end{cases}$$

The details of this derivation are spelled out in Friedman, Hastie, Hoefling, and Tibshirani (2007).

25. Overview of Coordinate Descent Procedure: Thus, the simple least-squares coefficient on the partial residual are computed, soft-thresholding is applied to take care of the lasso contribution to the penalty, and then a proportional shrinkage is applied for the ridge penalty. This algorithm was suggested by van der Kooij (2007).

Naïve Updates

1. Closer Examination of Coordinate Update: Looking more closely at

$$\tilde{\beta}_j \leftarrow \frac{S\left(\frac{1}{N}\sum_{i=1}^{j} x_{ij}\left[y_i - \tilde{y}_i^{(j)}\right], \lambda\alpha\right)}{1+\lambda(1-\alpha)}$$

one can see that

$$y_i - \tilde{y}_i^{(j)} = y_i - \hat{y}_i + x_{ij}\tilde{\beta}_j = r_i + x_{ij}\tilde{\beta}_j$$

where $\hat{y}_i$ is the current fit of the model for observation i and hence r_i the current residual.

2. Simplification for Naïve Update Sequence: Thus

$$\frac{1}{N}\sum_{i=1}^{j} x_{ij}\left[y_i - \tilde{y}_i^{(j)}\right] = \frac{1}{N}\sum_{i=1}^{j} x_{ij} r_i + \tilde{\beta}_j$$

because the x_j are standardized.

3. Sum over x_{ij} and r_i: The first term on the right-hand side is the gradient of the loss with respect to β_j.
4. Origin of Coordinate Descent Efficiency: It is clear from

$$\frac{1}{N}\sum_{i=1}^{j} x_{ij}\left[y_i - \tilde{y}_i^{(j)}\right] = \frac{1}{N}\sum_{i=1}^{j} x_{ij} r_i + \tilde{\beta}_j$$

why coordinate descent is computationally efficient. Many coefficients are zero, remain zero after thresholding, and so nothing needs to be changed. Such a step costs $\mathcal{O}(N)$ operations – the sum to compute the gradient.

5. Coefficient Change after the Thresholding: On the other hand, if a coefficient does change after the thresholding, r_i is changed in $\mathcal{O}(N)$ and the step costs $\mathcal{O}(2N)$. Thus, a complete cycle through all p variables costs $\mathcal{O}(Np)$ operations.
6. Naïve vs. Covariance Updating Algorithm: Friedman, Hastie, and Tibshirani (2010) refer to this as the *naïve algorithm*, since this is generally less efficient than the *covariance updating* algorithm to follow.
7. IRLS - Iteratively Reweighted Least Squares: Later, these algorithms are used in the context of iteratively reweighted least squares IRLS, where the observation weights change frequently; there the naïve algorithm dominates.

Covariance Updates

1. Revisiting the x_{ij}/r_i Sum: Further efficiencies can be achieved in computing the updates in the equation above. One can write the first term on the right – up to a factor $\frac{1}{N}$ – as

$$\sum_{i=1}^{j} x_{ij} r_i = \langle x_j, y \rangle - \sum_{k:|\tilde{\beta}_k|>0} \langle x_j, x_k \rangle \tilde{\beta}_k$$

   where

$$\langle x_j, y \rangle = \sum_{i=1}^{j} x_{ij} y_i$$

2. Inner Products across each Feature: Hence, there is a need to compute inner products of each feature with y initially, and then each time a new feature x_k enters the model – for the first time – one needs to compute and store its inner product with all the rest of the features - $\mathcal{O}(Np)$ operations.
3. Update of each Gradient Coefficient: The p gradient components in

$$\sum_{i=1}^{j} x_{ij} r_i = \langle x_j, y \rangle - \sum_{k:|\tilde{\beta}_k|>0} \langle x_j, x_k \rangle \tilde{\beta}_k$$

are stored. If one of the coefficients currently in the model changes, one can update each gradient in $\mathcal{O}(p)$ operations.

4. Non-zero Terms in the Model: Hence with m non-zero terms in the model, a complete cycle costs $\mathcal{O}(pm)$ operations if no new variables become non-zero, and costs $\mathcal{O}(Np)$ for each new variable entered.
5. Number of Calculations avoided: Importantly, $\mathcal{O}(N)$ calculations do not have to be made at every step. This is the case for all penalized procedures with squared error loss.

Sparse Updates

1. Feature-Matrix that is Extremely Sparse: One is sometimes faced with problems where the $N \times p$ feature-matrix X is extremely sparse.
2. Example Document Classification Feature Vector: A leading example is from document classification, where the feature vector uses the so-called "bag-of-words" model. Each document is scored for the presence/absence of the words in the entire dictionary under consideration – sometimes counts are used, or some transformation of counts.
3. Efficient Storage using Sparse Format: Since most words are absent, the feature vector for each document is mostly zero, and so the entire matrix is mostly zero. Such matrices are stored efficiently in the *sparse matrix format*, where only non-zero entries and the coordinates where they occur are stored.
4. Sparsity Exploitation using Coordinate Descent: Coordinate descent is ideally setup to exploit such sparsity, in an obvious way. The $\mathcal{O}(N)$ inner-product operations in either the native or covariance updates can exploit the sparsity, by summing over only the non-zero entities.
5. Scaling vs. Centering of Variables: Note that in this case scaling of the variable will not alter the sparsity, but centering will. So scaling is performed up front, but centering is incorporated in the algorithm in an efficient and obvious manner.

Weighted Updates

1. Weight Associated with each Observation: Often a weight w_i – other than $\frac{1}{N}$ – is associated with each observation. This will arise naturally in later sections where observations receive weights in the IRLS algorithm.
2. Complications in the Update Step: In this case the update step

$$\tilde{\beta}_j \leftarrow \frac{S\left(\frac{1}{N}\sum_{i=1}^{j} x_{ij}\left[y_i - \tilde{y}_i^{(j)}\right], \lambda\alpha\right)}{1 + \lambda(1-\alpha)}$$

   becomes only slightly more complicated

$$\tilde{\beta}_j \leftarrow \frac{S\left(\sum_{i=1}^{j} w_i x_{ij}\left[y_i - \tilde{y}_i^{(j)}\right], \lambda\alpha\right)}{\sum_{i=1}^{j} w_i {x_{ij}}^2 + \lambda(1-\alpha)}$$

3. Impact of Weights on Computation: If the x_j are not standardized, there is a similar sum-of-squares term in the denominator – even without weights. The presence of weights not change the computational costs of either algorithm much, as long as the weights remain fixed.

Path-wise Coordinate Descent

1. Sequence of Values for λ: Friedman, Hastie, and Tibshirani (2010) compute the solutions for a decreasing sequence of values for λ, starting at the smallest value λ_{max} for which the entire vector

$$\hat{\beta} = 0$$

2. Exploitation of the Warm Starts: Apart from giving it a path of solutions, this scheme exploits *warm starts*, and leads to a more stable algorithm.
3. Path Given by Corresponding λ: In certain situations, it is faster to compute the path down to λ – for small λ – than the solution only at that value for λ.
4. λ_{max} – Starting Value for λ: When

$$\tilde{\beta} = 0$$

it can be seen from

$$\tilde{\beta}_j \leftarrow \frac{S\left(\frac{1}{N}\sum_{i=1}^{j} x_{ij}\left[y_i - \tilde{y}_i^{(j)}\right], \lambda\alpha\right)}{1 + \lambda(1-\alpha)}$$

that $\tilde{\beta}_j$ will stay zero if

$$\frac{1}{N}\left|\langle x_j, y\rangle\right| < \lambda\alpha$$

Hence

$$N\alpha\lambda_{max} = \max_{l} |\langle x_l, y \rangle|$$

5. λ_{min}, K, and ϵ Choices: The strategy is to select a minimum value

$$\lambda_{min} = \epsilon\lambda_{max}$$

and construct a sequence of K values of λ decreasing from λ_{max} to λ_{min} on the log scale. Typical values are

$$\epsilon = 0.001$$

and

$$K = 100$$

Other Details

1. Centering of each Predictor Variable: Irrespective of whether the variables are standardized to have variance 1, each predictor variable is always centered. Since the intercept is not regularized, this means that

$$\hat{\beta}_0 = \tilde{y}$$

the mean of the y_i, for all values of α and λ.

2. Different Penalties for Each Variable: It is easy to allow different penalties λ_j for each of the variables. This is implemented via the penalty scaling parameter

$$\gamma_j \geq 0$$

If

$$\gamma_j > 0$$

the penalty is then applied to β_j is

$$\lambda_j = \lambda\gamma_j$$

3. Penalty Rescaling as Adaptive Lasso: If

$$\gamma_j = 0$$

that variable does not get penalized, and always enters the model unrestricted at the first step and remains in the model. Penalty rescaling would also allow, for example, the software to be used to implement the *adaptive lasso* (Zou (2006)).

4. Iterations around the Active Set: Considerable speedup is obtained by organizing the iterations around the *active set* of features – those with non-zero coefficients.
5. Iterating Active Set until Convergence: After a complete cycle through all the variables, one iterates only on the active set till convergence. If another complete cycle does not change the active set, it is done, otherwise the process is repeated. Active set convergence is also mentioned in Krishnapuram and Hartemink (2005) and Meier, van de Geer, and Buhlmann (2008).

Regularized Logistic Regression

1. Usage when Response Variable is Binary: When the response variable is binary, the linear logistic regression model is often used. Denote by $\mathcal{G}$ the response variable, taking values in

$$\mathcal{G} = \{1, 2\}$$

   – the labeling of the elements is arbitrary.
2. Class Probabilities as Linear Functions: The logistic regression model represents the class-conditional probabilities through a linear function of the predictors

$$\mathbb{P}[\mathcal{G} = 1|x] = \frac{1}{1 + e^{-(\beta_0 + x^T\beta)}}$$

$$\mathbb{P}[\mathcal{G} = 2|x] = \frac{1}{1 + e^{\beta_0 + x^T\beta}} = 1 - \mathbb{P}[\mathcal{G} = 1|x]$$

3. Alternate Expression for Class Probabilities: Alternatively, the above implies that

$$\log \frac{\mathbb{P}[\mathcal{G} = 1|x]}{\mathbb{P}[\mathcal{G} = 2|x]} = \beta_0 + x^T\beta$$

4. Regularized Maximum Binomial Likelihood Fit: Here, this model is fit by the regularized maximum binomial likelihood model. Let

$$p(x_i) = \mathbb{P}[\mathcal{G} = 1|x_i]$$

be the probability for observation i at a particular value for the parameters (β_0, β), the penalized log likelihood is maximized from

$$\underset{(\beta_0, \beta) \in \mathbb{R}^{p+1}}{max} \left\{ \frac{1}{N} \sum_{i=1}^{N} \left[I(g_i = 1) \log p(x_i) + I(g_i = 2) \log\big(1 - p(x_i)\big) \right] - \lambda P_\alpha(\beta) \right\}$$

5. Log-likelihood Part Above: Denoting

$$y_i = I(g_i = 1)$$

the log-likelihood part above can be written in the most explicit form

$$l(\beta_0, \beta) = \frac{1}{N} \sum_{i=1}^{N} \left[y_i \cdot (\beta_0 + {x_i}^T\beta) - \log\big(1 + e^{\beta_0 + {x_i}^T\beta}\big) \right]$$

a concave function of the parameters.

6. Algorithm for Maximizing Log-likelihood: The Newton algorithm for maximizing log-likelihood above amounts to iteratively re-weighted least squares.
7. Quadratic Approximation to Log-likelihood: Hence, if the current estimates of the parameters are $(\tilde{\beta}_0, \tilde{\beta})$, the quadratic approximation to log-likelihood, i.e., the Taylor expansion about the current estimates, may be formed as

$$l_Q(\beta_0, \beta) = -\frac{1}{2N}\sum_{i=1}^{N} w_i(z_i - \beta_0 - {x_i}^T\beta)^2 + C^2(\tilde{\beta}_0, \tilde{\beta})$$

where

$$z_i = \tilde{\beta}_0 + {x_i}^T\tilde{\beta} + \frac{y_i - \tilde{p}(x_i)}{\tilde{p}(x_i)[1 - \tilde{p}(x_i)]}$$

which is the working response, and

$$w_i = \tilde{p}(x_i)[1 - \tilde{p}(x_i)]$$

8. Newton Update by Minimizing l_Q: The last term above is a constant. The Newton update is obtained by minimizing l_Q.
9. l_Q Quadratic Approximation around $(\tilde{\beta}_0, \tilde{\beta})$: The current approach is similar. For each value λ, an outer loop is created that computes the quadratic approximation l_Q about the current parameters $(\tilde{\beta}_0, \tilde{\beta})$.

10. Penalized Weighted Least-Squares Problem: Coordinate descent is then used to solve the penalized weighted least-squares problem

$$\min_{(\beta_0, \beta) \in \mathbb{R}^{p+1}} \{-l_Q(\beta_0, \beta) + \lambda P_\alpha(\beta)\}$$

This approach to a sequence of nested loops.

11. Outer Loop: Decrement λ
12. Middle Loop: Update the quadratic approximate l_Q using the current parameters $(\tilde{\beta}_0, \tilde{\beta})$
13. Inner Loop: Run the coordinate descent algorithm on the penalized weighted least-squares problem above.
14. Details of the Algorithm Implementation: There are several important details in the implementation of this algorithm.
15. Issues with Logistic Regression Fit: When

$$p \gg N$$

one cannot run λ all the way to zero, because the saturated logistic regression fir is undefined, i.e., the parameters wander off to $\pm\infty$ in order to achieve probabilities of 0 or 1.

16. Impact on λ Sequence Range: Hence, the λ sequence runs down to

$$\lambda_{min} = \epsilon \lambda_{max} > 0$$

17. Eliminating Blowup of Fitted Probabilities: Care is taken to avoid coefficients diverging in order to achieve fitted probabilities of 0 or 1. When a probability is within

$$\varepsilon = 10^{-5}$$

it is set to 1, and the weights are set to ε. 0 is treated similarly.

18. Upper Bound on the Horizon: Friedman, Hastie, and Tibshirani (2010) allow for the option to approximate the Hessian terms by an exact upper bound. This is obtained by setting

$$w_i = \tilde{p}(x_i)[1 - \tilde{p}(x_i)]$$

all equal to 0.25 (Krishnapuram and Hartemink (2005)).

19. Representation of Grouped Response Data: Further, the response data is allowed to be supplied in the form of a two-column matrix of counts, sometimes referred to as *grouped* data.
20. Convergence Properties of Newton Properties: The Newton algorithm is not guaranteed to converge without step-size optimization (Lee, Lee, Abbeel, and Ng (2006)).
21. Checks for Detecting Newton Convergence: Checks for divergence are not implemented here; this would slow it down, and when used as recommended, it is not necessary.
22. Closed Form Expression for Starters: A closed form expression has been used for the starting solutions, and each subsequent expression is warm started from the previous close-by solution, which generally makes the quadratic convergence very accurate. No divergence problems have been encountered so far.

Regularized Multinomial Regression

1. Generalization to Multinomial Logistic Regression: When the categorical response $\mathcal{G}$ has

$$K > 2$$

levels, the linear logistic regression model can be generalized to a multi-logit model.

2. Extension to K Logit Categories: The traditional approach is to extend

$$\log \frac{\mathbb{P}[\mathcal{G} = 1|x]}{\mathbb{P}[\mathcal{G} = 2|x]} = \beta_0 + x^T\beta$$

to $K - 1$ logits

$$\log \frac{\mathbb{P}[\mathcal{G} = l|x]}{\mathbb{P}[\mathcal{G} = K|x]} = \beta_{0l} + x^T\beta_l$$

$$l = 1, \dots, K - 1$$

3. β_l - the p-vector of Coefficients: Here, β_l is a p-vector of coefficients. As in Zhu and Hastie (2004), a symmetric approach is chosen here.
4. Modeling the Logit Category Probability: Thus, one models

$$\mathbb{P}[\mathcal{G} = l|x] = \frac{e^{\beta_{0l} + x^T\beta_l}}{\sum_{k=1}^{K} e^{\beta_{0k} + x^T\beta_k}}$$

5. Role of Constraints on Parameters: The above parametrization is not estimable without constraints, because for any values of the parameters $\{\beta_{0l}, \beta_l\}_1^K, \{\beta_{0l} - c_0, \beta_l - c\}_1^K$, the

above logit probability gives identical probabilities. Regularization deals with that ambiguity in a natural way.

6. Regularized Maximum Multinomial Likelihood Fit: The above model is fit by regularized maximum multinomial likelihood fit. Using a similar notation as before, let

$$p_l(x_i) = \mathbb{P}[\mathcal{G} = l|x_i]$$

and let

$$g_i \in \{1, \dots, K\}$$

be the i[th] response.

7. Maximizing the Log-likelihood Penalization:

$$\underset{(\beta_{0l}, \beta_l) \in \mathbb{R}^{K(p+1)}}{max} \left\{ \frac{1}{N} \sum_{i=1}^{N} \log p_{g_i}(x_i) - \lambda \sum_{l=1}^{K} P_\alpha(\beta_l) \right\}$$

8. Usage of Indicator Response Matrix: Denoting by Y the $N \times K$ *indicator* response matrix, with elements

$$y_{il} = I(g_i = l)$$

one can write the log-likelihood of the above maximization in the more explicit form

$$l(\{\beta_{0l}, \beta_l\}_1^K) = \frac{1}{N}\sum_{i=1}^{N}\left[\sum_{l=1}^{K} y_{il}(\beta_{0l} + x_i^T\beta_l) - \log\left(\sum_{l=1}^{K} e^{\beta_{0l}+x_i^T\beta_l}\right)\right]$$

9. Newton Algorithm for Multinomial Regression: The Newton algorithm for multinomial regression can be tedious, because of the vector nature of the response observations. Instead of the weights w_i as in

$$w_i = \tilde{p}(x_i)[1 - \tilde{p}(x_i)]$$

   one gets the weight *matrices*, for example.
10. Avoidance of Newton Algorithm Intricacies: However, in the spirit of coordinate descent, one can avoid these complexities.
11. Log likelihood Partial Quadratic Approximation: One can perform *partial Newton steps* by forming a partial quadratic approximation to the log-likelihood, allowing only $\{\beta_{0l}, \beta_l\}$ to vary for a single class at a time.
12. Expression for Log-likelihood Probability: It is not hard show that this is

$$l_{Ql}(\beta_{0l}, \beta_l) = -\frac{1}{2N}\sum_{i=1}^{N} w_{il}(z_{il} - \beta_{0l} - {x_i}^T\beta_l)^2 + C^2(\{\beta_{0l}, \beta_l\}_1^K)$$

   where, as before

$$z_{il} = \tilde{\beta}_{0l} + {x_i}^T\tilde{\beta}_l + \frac{y_{il} - \tilde{p}_l(x_i)}{\tilde{p}_l(x_i)[1 - \tilde{p}_l(x_i)]}$$

$$w_i = \tilde{p}(x_i)[1 - \tilde{p}(x_i)]$$

13. Loop over the Category Classes: The approach is similar to the two-class case, except one has to cycle over the classes as well as in the out loop.
14. Outer Loop Cycle over l: For each value of λ, one creates an outer loop that cycles over l and computes the partial quadratic approximation l_{Ql} about the current parameters $(\tilde{\beta}_0, \tilde{\beta})$.
15. Penalized Weighted Least-Squares Formulation: The coordinate descent can be used to solve the penalized weighted least-squares problem

$$\underset{(\beta_{0l}, \beta_l) \in \mathbb{R}^{p+1}}{min} \{-l_{Ql}(\beta_{0l}, \beta_l) + \lambda P_\alpha(\beta_l)\}$$

This amounts to the following nested loops:

16. Outer Loop: Decrement λ
17. Outer Middle Loop: Cycle over

$$l \in \{1, \cdots, K, 1, \cdots, K, \cdots\}$$

18. Inner Middle Loop: Update the quadratic approximation l_{Ql} using the current parameters $(\tilde{\beta}_{0k}, \tilde{\beta}_k)_1^K$
19. Inner Loop: Run the coordinate descent algorithm on the penalized weighted least-squares problem.

Regularization and Parameter Ambiguity

1. Parameter Degeneracy to an Offset: As was pointed out earlier, if $(\beta_{0l}, \beta_l)_1^K$ characterizes a fitted model for

$$\mathbb{P}[\mathcal{G} = l|x] = \frac{e^{\beta_{0l}+x^T\beta_l}}{\sum_{k=1}^K e^{\beta_{0k}+x^T\beta_k}}$$

   then $(\beta_{0l} - c_0, \beta_l - c)_1^K$ where c is a p-vector – gives an identical fit.
2. Sensitivity of Penalty to Offset: Although this means that the log-likelihood part of

$$\underset{(\beta_{0l}, \beta_l) \in \mathbb{R}^{K(p+1)}}{max} \left\{ \frac{1}{N} \sum_{i=1}^N \log p_{g_i}(x_i) - \lambda \sum_{l=1}^K P_\alpha(\beta_l) \right\}$$

   is insensitive to (c, c_0), the penalty is not.
3. Imposing the Estimation of Coefficients: In particular, one can always improve the estimate $(\beta_{0l}, \beta_l)_1^K$ with respect to

$$\underset{(\beta_{0l}, \beta_l) \in \mathbb{R}^{K(p+1)}}{max} \left\{ \frac{1}{N} \sum_{i=1}^N \log p_{g_i}(x_i) - \lambda \sum_{l=1}^K P_\alpha(\beta_l) \right\}$$

   by solving

$$\underset{c \in \mathbb{R}^p}{max} \left\{ \sum_{l=1}^K P_\alpha(\beta_l - c) \right\}$$

4. Separate Estimate for each Estimate: This can be done separately for each coordinate, hence

$$c_j = \arg\min_t \sum_{l=1}^{K} \left[\frac{1}{2}(1-\alpha)\left(\beta_{jt} - t\right)^2 + \alpha\left|\beta_{jt} - t\right| \right]$$

5. Coefficient Offset Parameter Bound Theorem #1: Consider the above expression c_j for values

$$\alpha \in [0, 1]$$

Let $\beta_{j,AVERAGE}$ be the mean of β_{jl}, and $\beta_{j,MEDIAN}$ a median of β_{jl} – and for simplicity assume

$$\beta_{j,AVERAGE} \leq \beta_{j,MEDIAN}$$

6. Coefficient Offset Parameter Bound Theorem #2: Thus, one has

$$c_j \in \left[\beta_{j,AVERAGE}, \beta_{j,MEDIAN}\right]$$

with the left endpoint achieved if

$$\alpha = 0$$

and the right if

$$\alpha = 1$$

7. Coefficient Offset Parameter Bound Proof #1: One has

$$c_j = \underset{t}{\arg min} \sum_{l=1}^{K} \left[\frac{1}{2}(1-\alpha)\left(\beta_{jt} - t\right)^2 + \alpha\left|\beta_{jt} - t\right| \right]$$

Suppose

$$\alpha \in (0,1)$$

8. Coefficient Offset Parameter Bound Proof #2: Differentiating with respect to t – using a sub-gradient representation – one has

$$\sum_{l=1}^{K} \left[-(1-\alpha)(\beta_{kl} - t) - \alpha s_{jl} \right] = 0$$

where

$$s_{jl} = sign(\beta_{jl} - t)$$

if

$$\beta_{jl} \neq t$$

and

$$s_{jl} \in [-1, 1]$$

otherwise.

9. Coefficient Offset Parameter Bound Proof #3: This gives

$$t = \beta_{j,AVERAGE} + \frac{I}{K}\frac{\alpha}{1-\alpha}\sum_{l=1}^{K} s_{jl}$$

10. Coefficient Offset Parameter Bound Proof #4: It follows that t cannot be larger than $\beta_{j,MEDIAN}$ since then the second term above could be negative and this would imply that t is less than $\beta_{j,AVERAGE}$
11. Coefficient Offset Parameter Bound Proof #5: Similarly, t cannot be less than $\beta_{j,AVERAGE}$ since then the second term would have to be negative, implying that t is larger than $\beta_{j,MEDIAN}$
12. Simplified Newton Search Scheme: The mean and the median endpoints are obvious. A consequence of the theorem is that a very simple search algorithm can be used to solve

$$c_j = \underset{t}{\arg min} \sum_{l=1}^{K} \left[\frac{1}{2}(1-\alpha)\left(\beta_{jl} - t\right)^2 + \alpha\left|\beta_{jt} - t\right|\right]$$

13. Piece wise Quadratic Objective – with Knots: The objective is piece-wise quadratic, with knots defined by β_{jl}. One needs only evaluate solutions in the intervals including mean and the median, and those in between.
14. Parameter Recentering after Inner Middle Loop: One *recenters* the parameters in each index set j after each *inner middle loop* step, using the solution c_j after each j.
15. Parameters that are not Regularized: Note that not all the parameters in the models are regularized. The intercept β_{0l} are not, and with the penalty modifiers γ_l others need not be as well. For these parameters, one uses mean centering.

Grouped and Matrix Responses

1. Data Matrix of Non-negative Numbers: As in the two-class case, the data can be presented in the form of $N \times K$ matrix m_{il} of non-negative numbers.
2. m_i as the Observation Weight: For example, if the data are grouped, at each x_i we have a number of multinomial samples, with m_{il} falling into the category l. In this case each row is divided by the row sum

$$m_i = \sum_l m_{il}$$

and produce the response matrix

$$y_{il} = \frac{m_{il}}{m_i}$$

m_i thus becomes an observation weight.

3. Changes to Working Response/Weights: Thus, the penalized maximum likelihood changes in a trivial way. The working response

$$z_{il} = \tilde{\beta}_{0l} + x_i{}^T\tilde{\beta}_l + \frac{y_{il} - \tilde{p}_l(x_i)}{\tilde{p}_l(x_i)[1 - \tilde{p}_l(x_i)]}$$

is defined exactly the same way, using y_{il} just defined. The weights in

$$w_i = \tilde{p}(x_i)[1 - \tilde{p}(x_i)]$$

get augmented with the observation m_i

$$w_i = m_i\tilde{p}(x_i)[1 - \tilde{p}(x_i)]$$

4. Matrix of Class Weight Properties: Equivalently, the data can be presented directly as a matrix of class proportions, along with a weight vector. From the point of view of the algorithms, any matrix of positive vectors and any non-negative weight vector will be treated the same way.

Timings

1. Comparing Coordinate-wise Algorithms with Competitors: This section compares the run times of the coordinate-wise algorithm to certain competing algorithms. These use the lasso penalty

$$\alpha = 1$$

in both the regression and the logistic regression settings. All timings were carried out on an Intel Xeon 2.8 GHz processor.
2. Unavailability of Elastic Net Software: No comparisons are performed on the elastic net versions of the penalties, since there is not much software available.
3. glmnet - lars vs. elasticnet Packages: Comparisons of glmnet with the R package elasticnet will mimic the comparisons with lars (Hastie and Efron (2007)) for the lasso, since elasticnet (Zou and Hastie (2004)) is built on the lars package.

Regression with the Lasso

1. N Gaussian Observations / p Predictors: Friedman, Hastie, and Tibshirani (2010) generated Gaussian data with N observations and p predictors X_j/X_j' having the same population correlation ρ. They tried a number of different values for N and p, with ρ varying from 0 to 0.95.
2. Correlation of the Response Values: The outcome values were generated by

$$Y = \sum_{j=1}^{p} X_j \beta_j + k \cdot Z$$

where

$$\beta_j = (-1)^j e^{-\frac{2(j-1)}{20}}$$

$$Z \sim \mathcal{N}(0,1)$$

and k is chosen such that the signal-to-noise ratio is 3.0

3. Strategy behind Construction of Coefficients: The coefficients are constructed to have alternating signs and to be exponentially decreasing.
4. CPU Times – Coordinate-wise vs. lars:

	Linear regression – Dense features					
	Correlation					
	0	0.1	0.2	0.5	0.9	0.95
	$N = 1000,\ p = 100$					
`glmnet (type = "naive")`	0.05	0.06	0.06	0.09	0.08	0.07
`glmnet (type = "cov")`	0.02	0.02	0.02	0.02	0.02	0.02
`lars`	0.11	0.11	0.11	0.11	0.11	0.11
	$N = 5000,\ p = 100$					
`glmnet (type = "naive")`	0.24	0.25	0.26	0.34	0.32	0.31
`glmnet (type = "cov")`	0.05	0.05	0.05	0.05	0.05	0.05
`lars`	0.29	0.29	0.29	0.30	0.29	0.29
	$N = 100,\ p = 1000$					
`glmnet (type = "naive")`	0.04	0.05	0.04	0.05	0.04	0.03
`glmnet (type = "cov")`	0.07	0.08	0.07	0.08	0.04	0.03
`lars`	0.73	0.72	0.68	0.71	0.71	0.67
	$N = 100,\ p = 5000$					
`glmnet (type = "naive")`	0.20	0.18	0.21	0.23	0.21	0.14
`glmnet (type = "cov")`	0.46	0.42	0.51	0.48	0.25	0.10
`lars`	3.73	3.53	3.59	3.47	3.90	3.52
	$N = 100,\ p = 20000$					
`glmnet (type = "naive")`	1.00	0.99	1.06	1.29	1.17	0.97
`glmnet (type = "cov")`	1.86	2.26	2.34	2.59	1.24	0.79
`lars`	18.30	17.90	16.90	18.03	17.91	16.39
	$N = 100,\ p = 50000$					
`glmnet (type = "naive")`	2.66	2.46	2.84	3.53	3.39	2.43
`glmnet (type = "cov")`	5.50	4.92	6.13	7.35	4.52	2.53
`lars`	58.68	64.00	64.79	58.20	66.39	79.79

Table 1: Timings (in seconds) for **glmnet** and **lars** algorithms for linear regression with lasso penalty. The first line is **glmnet** using `naive` updating while the second uses `covariance` updating. Total time for 100 λ values, averaged over 3 runs.

The table shows the average CPU timings for the coordinate-wise algorithm, and the lars procedure (Efron, Hastie, Johnstone, and Tibshirani (2004)). All algorithms are implemented as R functions.

5. R/Fortran in Numerical Work: The coordinate-wise algorithm does all of its numerical work in Fortran, while lars (Hastie and Efron (2007)) does much of its work in R, calling Fortran routines for some matrix responses.
6. Lars in Fortran vs. Mixed: However, comparisons in Friedman, Hastie, Hoefling, and Tibshirani (2007) showed that lars was actually faster than a version entirely in Fortran.
7. Lars Computes the Full Path: Comparisons between different programs are always tricky: in particular, the lars procedures computes the entire path of solutions, while the coordinate-wise procedure solves the path for a set pre-defined locations along the solution path.
8. Number of Steps in lars: In the orthogonal case, lars takes $\min(N, p)$ steps: hence to make things roughly comparable, the latter two algorithms are called to solve a total of $\min(N, p)$ problems along the path.
9. Glmnet Considerably Faster than lars: The table above shows timing in seconds averaged over three runs. It can be seen that glmnet is considerably faster than lars; the covariance-updating version of the algorithm is a little faster than the naïve version when

$$N > p$$

   and a little slower when

$$p > N$$

10. High Correlation between the Features: It would be expected that high correlation between the features would increase the runtime of glmnet, but this does not seem to be the case.

Lasso-Logistic Regression

1. Outcomes – Continuous and Two-Class: The same simulation setup as above has been used, except that the continuous outcome y, defined

$$p = \frac{1}{1 + e^{-y}}$$

is used to generate a two-class outcome z with

$$\mathbb{P}[z = 1] = p$$

$$\mathbb{P}[z = 0] = 1 - p$$

2. Comparison with the Following Algorithms: The speed of the glmnet was compared to the interior point l1logreg (Koh, Kim, and Boyd (2007a)), Bayesian Binary Regression BBR (Genkin, Lewis, and Madigan (2007), Madigan and Lewis (2007)), and the Lasso penalized logistic regression LPL (Wu and Lange (2008)). The latter two methods also use a coordinate descent approach.
3. Ten-fold Cross-validation across λ: The BBR software automatically performs ten-fold cross-methodology when given a set of λ values. Hence this chapter reports the total time for ten-fold cross-validation for all methods using the same 100 λ values for all.
4. Speed Improvement in Coordinate Descent:

	Logistic regression – Dense features					
	Correlation					
	0	0.1	0.2	0.5	0.9	0.95
	$N = 1000,\ p = 100$					
glmnet	1.65	1.81	2.31	3.87	5.99	8.48
l1logreg	31.475	31.86	34.35	32.21	31.85	31.81
BBR	40.70	47.57	54.18	70.06	106.72	121.41
LPL	24.68	31.64	47.99	170.77	741.00	1448.25
	$N = 5000,\ p = 100$					
glmnet	7.89	8.48	9.01	13.39	26.68	26.36
l1logreg	239.88	232.00	229.62	229.49	22.19	223.09
	$N = 100,000,\ p = 100$					
glmnet	78.56	178.45	205.94	274.33	552.48	638.50
	$N = 100,\ p = 1000$					
glmnet	1.06	1.07	1.09	1.45	1.72	1.37
l1logreg	25.99	26.40	25.67	26.49	24.34	20.16
BBR	70.19	71.19	78.40	103.77	149.05	113.87
LPL	11.02	10.87	10.76	16.34	41.84	70.50
	$N = 100,\ p = 5000$					
glmnet	5.24	4.43	5.12	7.05	7.87	6.05
l1logreg	165.02	161.90	163.25	166.50	151.91	135.28
	$N = 100,\ p = 100,000$					
glmnet	137.27	139.40	146.55	197.98	219.65	201.93

Table 2: Timings (seconds) for logistic models with lasso penalty. Total time for tenfold cross-validation over a grid of 100 λ values.

The table shows the result; in some respects, a method was omitted when it was seen to be very slow at smaller values for N or p. Again, glmnet is the clear winner; it slows down a little under high correlation.

5. Computation Time Function of N/p: The computation seems to be roughly linear in N, but grows faster than linear in p.

	Logistic regression – Sparse features					
	Correlation					
	0	0.1	0.2	0.5	0.9	0.95
	$N = 1000,\ p = 100$					
glmnet	0.77	0.74	0.72	0.73	0.84	0.88
l1logreg	5.19	5.21	5.14	5.40	6.14	6.26
BBR	2.01	1.95	1.98	2.06	2.73	2.88
	$N = 100,\ p = 1000$					
glmnet	1.81	1.73	1.55	1.70	1.63	1.55
l1logreg	7.67	7.72	7.64	9.04	9.81	9.40
BBR	4.66	4.58	4.68	5.15	5.78	5.53
	$N = 10,000,\ p = 100$					
glmnet	3.21	3.02	2.95	3.25	4.58	5.08
l1logreg	45.87	46.63	44.33	43.99	45.60	43.16
BBR	11.80	11.64	11.58	13.30	12.46	11.83
	$N = 100,\ p = 10,000$					
glmnet	10.18	10.35	9.93	10.04	9.02	8.91
l1logreg	130.27	124.88	124.18	129.84	137.21	159.54
BBR	45.72	47.50	47.46	48.49	56.29	60.21

Table 3: Timings (seconds) for logistic model with lasso penalty and sparse features (95% zero). Total time for ten-fold cross-validation over a grid of 100 λ values.

The table shows some results when the feature matrix is sparse: randomly 95% of the feature values to zero. Again, the glmnet procedure is significantly faster than l1logreg.

Real Data

1. Timing Results for Four Datasets:

Name	Type	N	p	**glmnet**	**l1logreg**	**BBR/BMR**
			Dense			
`Cancer`	14 class	144	16,063	2.5 mins		2.1 hrs
`Leukemia`	2 class	72	3571	2.50	55.0	450
			Sparse			
`InternetAd`	2 class	2359	1430	5.0	20.9	34.7
`NewsGroup`	2 class	11,314	777,811	2 mins	3.5 hrs	

Table 4: Timings (seconds, unless stated otherwise) for some real datasets. For the `Cancer`, `Leukemia` and `InternetAd` datasets, times are for ten-fold cross-validation using 100 values of λ; for `NewsGroup` we performed a single run with 100 values of λ, with $\lambda_{\min} = 0.05\lambda_{\max}$.

2. Cancer: Ramaswamy, Tamayo, Rifkin, Mukherjee, Yeang, Angelo, Ladd, Reich, Latulippe, Mesirov, Poggio, Gerald, Loda, Lander, and Golub (2002) – gene-expression data with 14 cancer classes. The comparison here is between glmnet and BMR (Genkin, Lewis, and Madigan (2007)), a multinomial version of **BBR**.
3. Leukemia: Golub, Slonim, Tamayo, Huard, Gaasenbeek, Mesirov, Coller, Loh, Downing, Caligiuri, Bloomfield, and Lander (1999) – gene-expression data with a binary response indicating type of leukemia – AML vs. ALL. The preprocessed data of Dettling (2004) was used.
4. Internet Ad: Kushmerick (1999) – document classification problem with mostly binary features. The response is binary, and indicates whether the document is an advertisement. Only 1.2% non-zero values in the predictor matrix.
5. NewsGroup: Lang (1995) – document classification problem. The training set is cultured from the data provided by Koh, Kim, and Boyd (2007a). The response is binary, and indicates a subclass of topics; the predictors are binary, and indicate the presence of particular tri-gram sequences. The predictor matrix has 0.05% non-zero values.
6. Online Availability of Chapter's Datasets: All datasets are available online with this chapter as saved R data objects – the latter two in sparse format using the *Matrix* package (Bates and Maechler (2022)).
7. BBR used Fewer λ Values: For the Leukemia and the InternetAd datasets, the BBR program used fewer than 100 λ values so the total time was estimated by scaling by the time for smaller number of values.

8. Datasets that are very Sparse: The InternetAd and NewsGroup datasets are both sparse: 1% for the former, and 0.05% for the latter. Again, the glmnet is considerably faster than the competing models.

Other Comparisons

1. Some other Models not Compared: When making comparisons, one invariably leaves out someone's favorite method. glmpath (Park and Hastie (2007, 2022)) extension of lars for GLMs has been left out, since it does not scale well to the size of problems considered here.
2. Sparse Linear Models - Alternate Schemes: Two referees of an earlier draft of Friedman, Hastie, and Tibshirani (2010) suggested two other methods. A single benchmark again each of these using the Leukemia data has been run, fitting models of 100 values of λ in each case.
3. OWL-QN: Orthant-wise Limited-memory Quasi-Newton Optimizer of l_1-regularized objectives (Andrew and Gao (2007a, 2007b)).
4. penalized Package: The R package penalized (Goeman (2010, 2022)) fits GLMs using a flat implementation of gradient ascent.
5. Comparisons across all these Models:

	MacBook Pro	HP Linux server
glmnet	0.34	0.13
penalized	10.31	
OWL-QN		314.35

Table 5: Timings (seconds) for the `Leukemia` dataset, using 100 λ values. These timings were performed on two different platforms, which were different again from those used in the earlier timings in this paper.

The shows these comparisons on two different machines glmnet is considerably faster in both cases.

Selecting the Tuning Parameters

1. Entire Solution Path across λ: The algorithms described in this chapter compute an entire path of solution – across λ – for any particular model, leaving the user to select a particular solution from the ensemble.
2. Choice Guided by Prediction Error: One general approach is to use prediction error to guide this choice. If a user is data rich, they can set aside some fraction – say a third – of their data for this purpose. They would then evaluate the predictive performance at each value of λ, and pick the model with the best performance.
3. K-fold Cross-validation: Alternatively, they can use K-fold cross-validation (Hastie, Tibshirani, and Friedman (2009)), where the user data is used for both training and testing in an unbiased way.
4. Binomial Deviance vs. Misclassification Error:

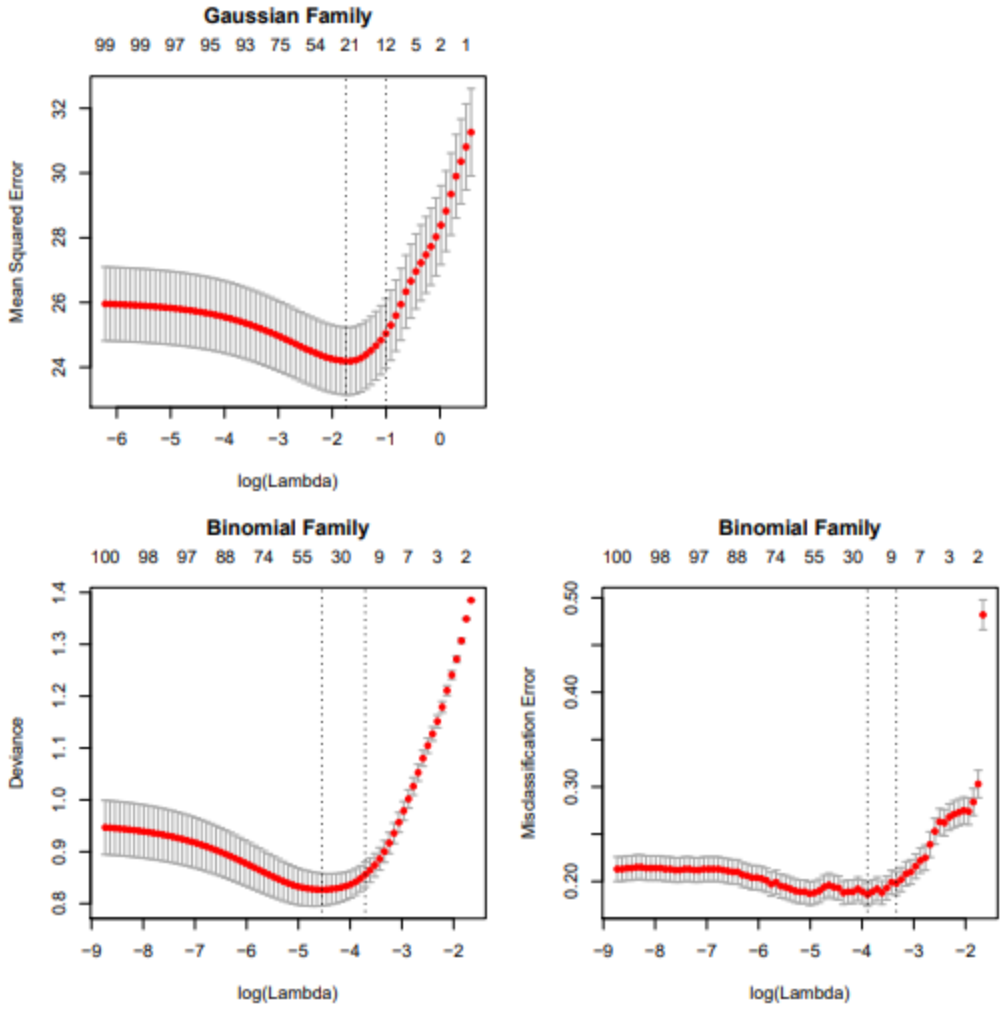


Figure 2: Ten-fold cross-validation on simulated data. The first row is for regression with a Gaussian response, the second row logistic regression with a binomial response. In both cases we have 1000 observations and 100 predictors, but the response depends on only 10 predictors. For regression we use mean-squared prediction error as the measure of risk. For logistic regression, the left panel shows the mean deviance (minus twice the log-likelihood on the left-out data), while the right panel shows misclassification error, which is a rougher measure. In all cases we show the mean cross-validated error curve, as well as a one-standard-deviation band. In each figure the left vertical line corresponds to the minimum error, while the right vertical line the largest value of lambda such that the error is within one standard-error of the minimum—the so called "one-standard-error" rule. The top of each plot is annotated with the size of the models.

The figure illustrates cross-validation on a simulated dataset. For logistic regression, one sometimes uses the binomial deviance rather than misclassification error, since the latter is smoother.

5. The "one-standard-error" Rule: One of ten uses the "one-standard-error" rule when selecting the best model; this acknowledges the fact that the risk curves are estimated with error, so errs on the side of parsimony (Hastie, Tibshirani, and Friedman (2009)).
6. Cross-validation to select α: Cross-validation can be used to select α as well, although it is often viewed as a higher-level parameter and chosen on more subjective grounds.

Discussion

1. Cyclical Coordinate Descent Methods: These methods are a natural approach for solving convex problems with l_1 or l_2 constraints, or mixtures of the two - the elastic net.
2. Coordinate-wise Minimization Explicit Formula: Each coordinate-descent step is fast, with an explicit formula for each coordinate-wise minimization.
3. Exploitation of the Model Sparsity: The method also exploits the sparsity of the model, spending much of the time evaluating only inner products for the variables with non-zero coefficients.
4. Computational Speed for Large N/p: Its computational speed for both large N and p are quite remarkable.
5. R-language Package Implementation of Lasso: An R-language package glmnet is available under general public license GPL-2 from the Comprehensive R Archive Network at http://CRAN.R-project.org/package=glmnet.
6. Data Inputs and MATLAB Functions: Sparse data inputs are handled by the *Matrix* package. MATLAB functions (Jiang (2009)) are available from http://www-stat.stanford.edu/~tibs/glmnet-matlab.

References

- Andrew, G., and J. Gao (2007a): Orthant-Wise Limited-memory Quasi-Newton Optimizer for L1-regularized Objectives
- Andrew, G., and J. Gao (2007b): Scalable Training of L1-Regularized Log-Linear Models
- Bates, D., and M. Maechler (2022): Sparse and Dense Matrix Classes and Methods
- Candes, E., and T. Tao (2007): The Dantzig Selector: Statistical Estimation when p is much Larger than n *The Annals of Statistics* **35 (6)** 2313-2351
- Chen, S. S., D. Donoho, and M. Saunders (1998): Atomic Decomposition by Basis Pursuit *SIAM Journal of Scientific Computing* **20 (1)** 33-61
- Daubechies, I., M. Defrise, and C. De Mol (2004): An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constraint *Communications on Pure and Applied Mathematics* **57 (11)** 1413-1457
- Dettling, M. (2004): Bag Boosting for Tumor Classification with Gene Expression Data *SIAM Journal of Scientific Computing* **20 (18)** 3583-3593
- Donoho, D. L., and I. M. Johnstone (1994): Ideal Spatial Adaptation by Wavelet Shrinkage *Biometrika* **81 (3)** 425-455
- Efron, B., T. Hastie, I. Johnstone, and R. Tibshirani (2004): Least Angle Regression *The Annals of Statistics* **32 (2)** 407-499
- Fan, J., and R. Li (2005): Variable Selection via Non-concave Penalized Likelihood and its Oracle Properties *Journal of the American Statistical Association* **96 (456)** 1348-1360
- Friedman, J., T. Hastie, H. Hoefling, and R. Tibshirani (2007): Path-wise Coordinate Optimization *The Annals of Applied Statistics* **2 (1)** 302-332
- Friedman, J. (2008): *Fast Sparse Regression and Classification* **Department of Statistics, Stanford University**
- Friedman, J., T. Hastie, and R. Tibshirani (2009): glmnet: Lasso and Elastic-Net Regularized Generalized Linear Models
- Friedman, J., T. Hastie, and R. Tibshirani (2010): Regularization Paths of Generalized Linear Models via Coordinate Descent *Journal of Statistical Software* **33 (1)** 1-22
- Fu, W. (1998): Penalized Regressions: The Bridge vs. the Lasso *Journal of Computational and Graphical Studies* **7 (3)** 397-416

- Genkin, A., D. Lewis, and D. Madigan (2007): Large-scale Bayesian Logistic Regression for Text Categorization *Technometrics* **49 (3)** 291-304
- Goeman, J. (2010): l_1 Penalized Estimation in the Cox Proportional Hazards Model *Biometrical Journal* **52 (1)** 70-84
- Goeman, J. (2022): L1 and L2 Penalized Regression Models
- Golub, T., D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander (1999): Molecular Classification of Cancer: Cancer Discovery and Class Prediction by Gene Expression Modeling *Science* **286 (5439)** 531-536
- Hastie, T., S. Rosset, R. Tibshirani, and J. Zhu (2004): The Entire Regularization Path for the Support Vector Machine *Journal of Machine Learning Research* **5** 1391-1415
- Hastie, T., and B. Efron (2007): lars: Least Angle Regression, Lasso, and Forward Stagewise
- Hastie, T., R. Tibshirani, and J. Friedman (2009): *The Elements of Statistical Learning: Prediction, Inference, and Data Mining 2nd Edition* **Springer-Verlag** New York
- Krishnapuram, B., and A. J. Hartemink (2005): Sparse Multinomial Logistic Regression: Fast Algorithms and Generalized Bounds *IEEE Transactions on Pattern Analysis and Machine Intelligence* **27 (6)** 957-968
- Koh, K., S. J. Kim, and S. Boyd (2007a): An Interior-point Method for Large Scale L_1-regularized Logistic Regression *Journal of Machine Learning Research* **8** 1519-1555
- Koh, K., S. J. Kim, and S. Boyd (2007b): l1_logreg: A Large-scale Solver for L_1-regularized Logistic Regression Problems
- Kushmerick, N. (1995): Learning to Remove Internet Advertisements
- Lang, K. (1995): NewsWeeder: Learning to Filter News, in: *Proceedings of the 12th International Conference on Machine Learning (editors: Preditis, A., and S. Russell)* 331-339
- Lee, S., H. Lee, P. Abbeel, and A. Ng (2006): Efficient L1 Regularized Logistic Regression
- Madigan, D., and D. Lewis (2007): Bayesian Logistic Regression (BBR, BMR, BXR)
- Meier, L., S. van de Deer, and P. Buhlmann (2008): The Group Lasso for Logistic Regression *Journal of the Royal Statistical Society B* **70 (1)** 53-71

- Osborne, M., B. Presnell, and B. Turlach (2000): A New Approach to Variable Selection in Least Squares Problems *IMA Journal of Numerical Analysis* **20 (3)** 389-404
- Park, M. Y., and T. Hastie (2007): L_1 Regularization Path Algorithm for Generalized Linear Models *Journal of the Royal Statistical Society B* **69 (4)** 659-677
- Park, M. Y., and T. Hastie (2018): glmpath: L1 Regularization Path for Generalized Linear Models and Cox Proportional Hazards Model
- Ramaswamy, S., P. Tamayo, R. Rifkin, S. Mukherjee, C. Yeang, M. Angelo, C. Ladd, M. Reich, E. Latulippe, J. Mesirov, G. Poggio, M. Loda, E. Lander, and T. Golub (2002): Multiclass Cancer Diagnosis using Tumor Gene Expression Signature *Proceedings of the National Academy of Sciences* **98 (26)** 15149-15154
- Rosset, S., and J. Zhu (2007): Piece-wise Linear Regularized Solution Paths *The Annals of Statistics* **35 (3)** 1012-1030
- Shevade, K., and S. Keerthi (2003): A Simple and Efficient Algorithm for Gene Selection using Sparse Logistic Regression *Bioinformatics* **19 (17)** 2246-2253
- Tibshirani, R. (1996): Regression Shrinkage and Selection with the Lasso *Journal of the Royal Statistical Society B* **58 (1)** 267-288
- Tibshirani, R. (1997): The Lasso Method for Variable Selection in the Cox Model *Statistics in Medicine* **16 (4)** 385-395
- Tseng, P. (2001): Convergence of a Block Coordinate Descent Method for Non-differentiable Minimization *Journal of Optimization Theory and Applications* **109 (3)** 475-494
- Van der Kooij, A. (2007): *Prediction Accuracy and Stability of Regression with Optimal Scaling Transformations* **Department of Data Theory, University of Leiden**
- Wu, T., and K. Lange (2008): Coordinate Descent Procedures for Lasso Penalized Regression *The Annals of Applied Statistics* **2 (1)** 224-244
- Wu, T., Y. Chen, T. Hastie, E. Sobel, and K. Lange (2008): Genome-wide Association Analysis by Penalized Logistic Regression *Bioinformatics* **25 (6)** 714-721
- Yuan, M., and Y. Lin (2007): Model Selection and Estimation in Regression with Grouped Variables *Journal of the Royal Statistical Society B* **68 (1)** 49-67
- Zhu, J. and T. Hastie (2004): Classification of Expression Arrays by Penalized Logistic Regression *Biostatistics* **5 (3)** 427-443

- Zou, H. and T. Hastie (2005): Regularization and Variable Selection vis the Elastic Net *Journal of the Royal Statistical Society B* **67 (2)** 301-320
- Zou, H. (2006): The Adaptive Lasso and its Oracle Properties *Journal of the American Statistical Association* **101 (476)** 1418-1429