A Robust hybrid of Lasso and Ridge regresssion

Group 35

June 23, 2022

Overview

- Lasso and Ridge regression are the regularized versions of least squares regression using L1 and L2 penalties respectively, on the coefficient vector.
- To make these regressions more robust we may replace least squares with Huber's criterion which is a hybrid of squared error (for relatively small errors) and absolute error (for relatively large ones).
- A reversed version of Huber's criterion (Berhu criterion) can be used as a hybrid penalty function. Relatively small coefficients contribute their L1 norm to this penalty while larger ones cause it to grow quadratically.
- Both the Huber and reversed Huber penalty functions employ a scale parameter. We provide an objective function that is jointly convex in the regression coefficient vector and these two scale parameters.

Introduction

- Regularization is often necessary in the context of machine learning, especially, when we are looking at regression.
- Regularized version of least squares can be beneficial in the following two situations:
  - When the number of variables in the linear system exceeds the number of observations.
  - When the model suffers from lack of generalization ability.
- Lasso and ridge regression are two famous versions of the RLS problem. However, they do have some disadvantages.
- To overcome them, we try to formulate a robust hybrid of lasso and ridge regression.

Linear Regression

Consider the regression problem of predicting $y \in \mathbb{R}$ based on $z \in \mathbb{R}^d$. Hence the training data in this case is given by the data points (z_i, y_i) for $i = 1, 2, \ldots n$. Suppose that each predictor vector z_i is converted into a **feature** vector $x_i \in \mathbb{R}^p$ using a fixed function ϕ as follows :

$$x_i = \phi(z_i) \tag{1}$$

Then the predictor for y, which is denoted by $\hat{y}$ is given by :

$$\hat{y} = \mu + x^T \beta \tag{2}$$

where $\beta \in \mathbb{R}^p$. The loss function in Ridge regression is given by :

$$L(\theta) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - \mu - x_i^T \beta)^2 \tag{3}$$

where θ denotes all the parameters involved in the above formulation.

Ridge Regression

In Ridge regression, we regularize the loss by adding a $L2$ penalty term. The objective function in Ridge regression is given by :

$$L(\theta) = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \tag{4}$$

$$L(\theta) = \sum_{i=1}^{n}(y_i - \mu - x_i^T \beta)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \tag{5}$$

where θ denotes all the parameters involved in the above formulation. The ridge parameter $\lambda \in [0, \infty]$. Defining $\epsilon_i = y_i - \mu - x_i^T \beta$, the objective function becomes :

$$L(\theta) = \|\epsilon\|_2^2 + \lambda \|\beta\|_2^2 \tag{6}$$

We can solve the Ridge Regression problem by **minimizing** the above function over θ.

Lasso Regression

In Lasso regression, we replace the $L2$ penalty term with $L1$ penalty term. The objective function in Lasso regression is given by :

$$L(\theta) = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{7}$$

$$L(\theta) = \sum_{i=1}^{n}(y_i - \mu - x_i^T \beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{8}$$

where θ denotes all the parameters involved in the above formulation. The lasso parameter $\lambda \in [0, \infty]$. Defining $\epsilon_i = y_i - \mu - x_i^T \beta$, the objective function becomes :

$$L(\theta) = \|\epsilon\|_2^2 + \lambda \|\beta\|_1 \tag{9}$$

We can solve the Lasso Regression problem by **minimizing** the above objective function over θ.

Comparison between Lasso and Ridge Regression

- Lasso regression can produce sparse solutions, i.e, the solutions with some or most of the β_j's equal to zero. Sparsity is desirable for interpretation.
- Ridge regression cannot produce such sparsity, but can shrink the weights in the solution, leading to generalization ability.
- Lasso forces sparsity into solutions. In case of several correlated features with large effect on the output, lasso regression tries to zero out some, perhaps all but one of them.
- Ridge regression, on the other hand, shares the coefficient value among the correlated features, without any selection.
- A belief exists that the solutions from $L1$ penalty are less accurate than from $L2$ penalty.

Huber Function

The least squares criterion is well suited to y_i with Gaussian distribution, but can give poor performance when y_i has a heavier tailed distribution, or what is almost the same, when there are outliers. To overcome this Huber introduced a robust estimator with a loss function, which is unaffected by very large residuals. **Huber function** is defined as:

$$\mathcal{H}(z) = \begin{cases} z^2 & |z| \leq 1 \\ 2|z| - 1 & |z| \geq 1 \end{cases} \tag{10}$$

The function has quadratic nature for smaller values of z and is linear for larger values of z.

Huber Function Cont.

Using this Huber function, the robust estimator can be defined as:

$$\sum_{i=1}^{n} \mathcal{H}_M \left(\frac{y_i - \mu - x_i^T \beta}{\sigma} \right) \tag{11}$$

Where, $\mathcal{H}_M(z)$ is:

$$\mathcal{H}_M(z) = M^2 \mathcal{H}(z/M) = \begin{cases} z^2 & |z| \leq M \\ 2M|z| - M^2 & |z| \geq M \end{cases} \tag{12}$$

Here, M is a shape parameter and σ is a scale parameter. M describes the point of transition from quadratic to linear and controls the amount of robustness. Therefore, if the error is small than $M\sigma$ it gets squared. Else it comes under linear regime.

Huber Function Cont.

The Huber criteria resembles least squares as M increases, making β more efficient for normally distributed data but less robust. The criterion is more like L1 regression for small values of M, making it more robust against outliers but less efficient for normally distributed data. Huber proposes that taking M = 1.35 gives more robustness while retaining 95% statistical efficiency for normally distributed data. For fixed values of M and σ and a convex penalty P(.), the following will be convex in $(\mu, \beta) \in \mathcal{R}^{p+1}$.

$$\sum_{i=1}^{n} \mathcal{H}_M \left(\frac{y_i - \mu - {x_i}^T \beta}{\sigma} \right) + \lambda \sum_{j=1}^{p} P(\beta_j) \tag{13}$$

Huber Function Cont.

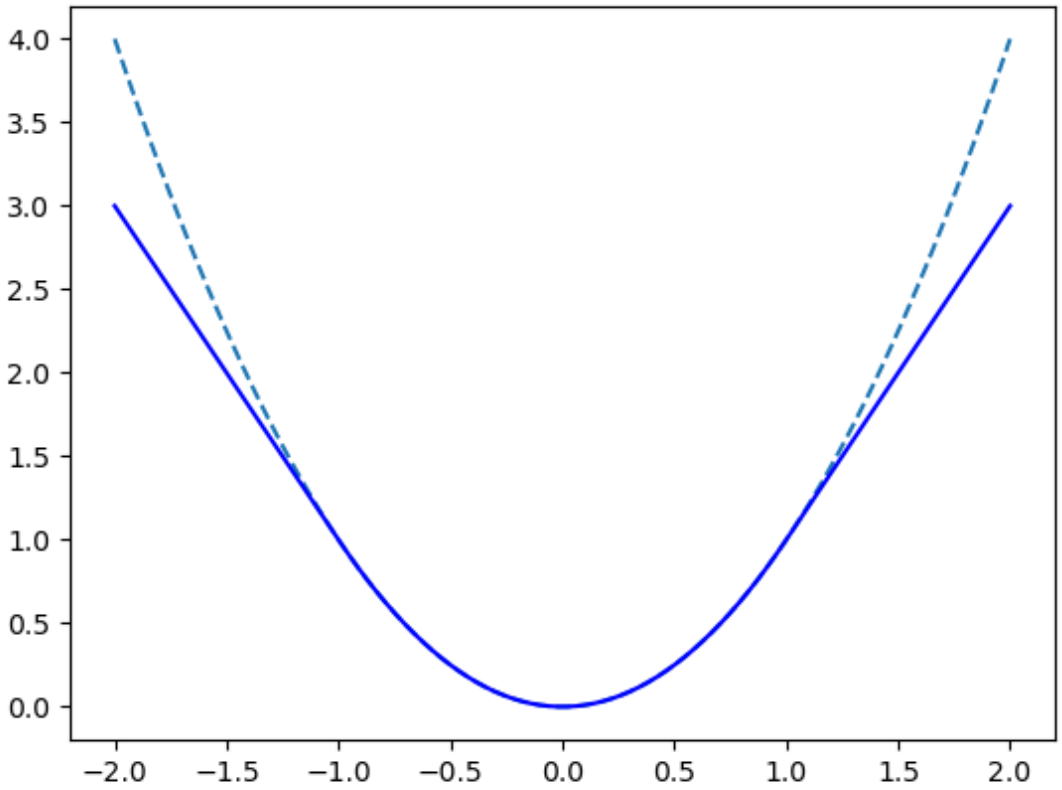

Figure: Huber function H(z)

Reverse Huber Function

We know that Huber function is quadratic near zero and linear for larger values. It's also a good fit for approximately normal distributed errors with heavier tails. But it is not well suited as a regularization term on regression coefficients. It is generally preferred to use L_1 regularization as it sets some β_j to zero making β sparse. This sparsity leads to savings in computation and storage. But there are some disadvantages associated with it. Hence, to overcome them, we propose a new penalty function Berhu, which uses L_1 penalty for small values and L_2 penalty for larger values. **Berhu function** is defined as:

$$\mathcal{B}(z) = \begin{cases} |z| & |z| \leq 1 \\ \frac{z^2+1}{2} & |z| \geq 1 \end{cases} \tag{14}$$

Reverse Huber Function Cont.

Then the penalty term using Berhu will be:

$$\mathcal{B}_M(z) = M\mathcal{B}_M\left(\frac{z}{M}\right) = \begin{cases} |z| & |z| \leq M \\ \frac{z^2+M^2}{2M} & |z| \geq M \end{cases} \tag{15}$$

Here M describes where the transition from linear to quadratic should occur. Note that the function $\mathcal{B}_M(z)$ is convex in z.

Reverse Huber Function Cont.

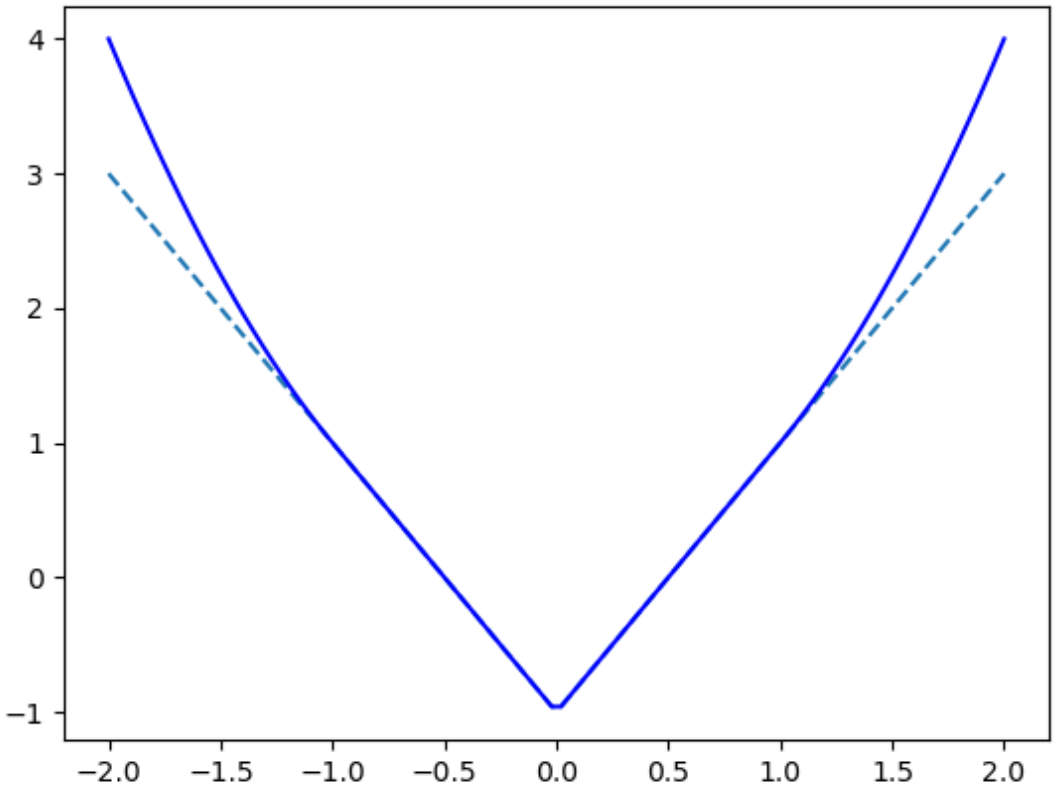

Figure: Berhu function B(z)

Proposed hybrid of lasso and ridge regression

Using Huber function for loss and Berhu function for penalty, we propose a robust hybrid of Lasso and Ridge regression. The robustness of the function is decided by value of M. The final hybrid function is:

$$\sum_{i=1}^{n} \mathcal{H}_M \left(\frac{y_i - \mu - {x_i}^T \beta}{\sigma} \right) + \lambda \sum_{j=1}^{p} \mathcal{B}_M \left(\frac{\beta_j}{\tau} \right) \tag{16}$$

equation (16) is jointly convex in μ and β given M, τ, and σ.

Concomitant scale estimation

The parameter M can be fixed to some value like 1.35. But there remain 3 other tuning parameters to consider: λ, σ, and τ. Instead of doing a 3 dimensional parameter search, we try to frame a objective which is jointly convex in $(\mu, \beta, \sigma, \tau)$, leaving only one dimensional search over λ. In practice it is necessary to estimate σ from the data, simultaneously with β. Huber proposed several ways to jointly estimate σ and β. One of his ideas for robust regression is to minimize

$$n\sigma + \sum_{i=1}^{n} \mathcal{H}_M\left(\frac{y_i - \mu - x_i^T \beta}{\sigma}\right)\sigma \tag{17}$$

over β and σ. For any fixed value of $\sigma \in (0, \infty)$, the minimizer β of (17) is the same as that of (11). The criterion (17) is however jointly convex as a function of $(\beta, \sigma) \in \mathbb{R}^p \times (0, \infty)$. Therefore convex optimization can be applied to estimate β and σ together. This removes the need for ad hoc algorithms to estimate β for fixed σ and σ for fixed β.

Concomitant scale estimation Cont.

The same idea can be used for τ. We can reframe the penalty term as $\tau + \mathcal{B}_M(\beta/\tau)\tau$ which is jointly convex in both β and τ. Using Huber's penalty function on the regression errors and the Berhu function on the coefficients leads to a criterion of the form

$$n\sigma + \sum_{i=1}^{n} \mathcal{H}_M\left(\frac{y_i - \mu - x_i^T\beta}{\sigma}\right)\sigma + \lambda\left[\rho\tau + \left\{\sum_{j=1}^{p} \mathcal{B}_M\left(\frac{\beta_j}{\tau}\right)\tau\right\}\right] \quad (18)$$

where $\lambda \in [0, \infty)$ governs the amount of regularization applied. The expression in (18) is jointly convex in $(\mu, \beta, \sigma, \tau) \in \mathbb{R}^{p+1} \times (0, \infty)^2$. We show this in the upcoming section. So we can apply convex optimization to estimate μ, β, σ and τ together.

Theory of concomitant scale estimation

Lemma

let ρ be a convex and twice differential function on an interval $\mathcal{I} \subseteq \mathbb{R}$. Then $\rho\left(\frac{\eta}{\sigma}\right)\sigma$ is a convex function of $(\eta, \sigma) \in \mathcal{I} \times (0, \infty)$.

Theory of concomitant scale estimation Cont.

Proof.

Let $\eta_0 \in \mathcal{I}$ and $\sigma_0 \in (0, \infty)$ and parameterise η and σ linearly as $\eta = \eta_0 + \cos(\theta) \times t$ and $\sigma = \sigma_0 + sin(\theta) \times t$ over t in an open interval with 0 and $\theta \in [0,\ 2\pi)$. Where θ denotes the direction.
Now let us try to calculate the double derivative of $\rho\left(\frac{\eta}{\sigma}\right)\sigma$ w.r.t t.
First derivative:

$$\frac{d}{dt}\rho\left(\frac{\eta}{\sigma}\right)\sigma = \rho'\left(\frac{\eta}{\sigma}\right)\frac{cos(\theta)\sigma - sin(\theta)\eta}{\sigma} + \rho\left(\frac{\eta}{\sigma}\right)sin(\theta) \tag{19}$$

Second derivative:

$$\frac{d^2}{dt^2}\rho\left(\frac{\eta}{\sigma}\right)\sigma = \rho''\left(\frac{\eta}{\sigma}\right)\frac{(cos(\theta)\sigma - sin(\theta)\eta)^2}{\sigma^3}. \tag{20}$$

which is always ≥ 0. i.e. the curvature of $\rho\left(\frac{\eta}{\sigma}\right)\sigma$ is always non negative.
Hence proved. □

Theory of concomitant scale estimation Cont.

Lemma

Let ρ be a convex function on an interval $\mathcal{I} \subseteq \mathbb{R}$ Then $\rho\left(\frac{\eta}{\sigma}\right)\sigma$ is a convex function of $(\eta, \sigma) \in \mathcal{I} \times (0, \infty)$.

Theory of concomitant scale estimation Cont.

Proof.

Take two points (η_0, σ_0) and (η_1, σ_1) both in $\mathcal{I} \times (0, \infty)$. Now take a point (η, σ) cutting the line segment joining the two points in $\frac{\lambda}{1-\lambda}$ ratio internally. Where $0 < \lambda < 1$ .Then,

$$\eta = \lambda\eta_1 + (1 - \lambda)\eta_0$$
$$\sigma = \lambda\sigma_1 + (1 - \lambda)\sigma_0$$

Theory of concomitant scale estimation Cont.

Proof.

For $\epsilon > 0$. Let ρ_ϵ be a convex and twice differentiable function on $\mathcal{I}$ that is everywhere within ϵ of ρ. Then,

$$\begin{aligned}\rho\left(\frac{\eta}{\sigma}\right)\sigma &\geq \rho_\epsilon\left(\frac{\eta}{\sigma}\right)\sigma - \epsilon\sigma \\ &\geq \lambda\rho_\epsilon\left(\frac{\eta_1}{\sigma_1}\right)\sigma_1 + (1-\lambda)\rho_\epsilon\left(\frac{\eta_0}{\sigma_0}\right)\sigma_0 - \epsilon\sigma \\ &\geq \lambda\rho\left(\frac{\eta_1}{\sigma_1}\right)\sigma_1 + (1-\lambda)\rho\left(\frac{\eta_0}{\sigma_0}\right)\sigma_0 - \epsilon(\lambda\sigma_1 + (1-\lambda)\sigma_0 - \sigma) \quad (21)\end{aligned}$$

By taking ϵ arbitrarily small we can say that

$$\rho\left(\frac{\eta}{\sigma}\right)\sigma \geq \lambda\rho\left(\frac{\eta_1}{\sigma_1}\right)\sigma_1 + (1-\lambda)\rho\left(\frac{\eta_0}{\sigma_0}\right)\sigma_0 \quad (22)$$

Hence proved that $\rho\left(\frac{\eta}{\sigma}\right)\sigma$ is convex. □

Theory of concomitant scale estimation Cont.

Lemma

Let $y_i \in \mathbb{R}$ and $x_i \in \mathbb{R}^p$ for i=1,2,. . . ,n. Let ρ be a convex function on $\mathbb{R}$. Then

$$n\sigma + \sum_{i=1}^{n} \rho\left(\frac{y_i - \mu - x_i^T \beta}{\sigma}\right)\sigma \tag{23}$$

is convex in $(\mu, \beta, \sigma) \in \mathbb{R}^{p+1} \times (0, \infty)$.

Theory of concomitant scale estimation Cont.

Proof.

The first term i.e $n\sigma$ is linear so it is convex. Now we need to prove that the second term $\sum_{i=1}^{n} \rho\left(\frac{y_i-\mu-x_i^T\beta}{\sigma}\right)\sigma$ is convex.

From 2 $\rho\left(\frac{\eta}{\sigma}\right)\sigma$ is convex on $(\eta,\sigma) \in \mathcal{I} \times (0,\infty)$. But the mapping of $\eta \to (y_i - \mu - x_i^T\beta)$ is affine so $\rho\left(\frac{y_i-x_i^T\beta}{\sigma}\right)\sigma$ is convex over $(\beta,\sigma) \in \mathbb{R}^p \times (0,\infty)$. And sum over i preserves the convexity. □

Now let us take $\rho(z) = z^2$.

From 1 we can tell that $\rho(z) = z^2$ is convex. so by applying it to the above 3 we can say that

$$n\sigma + \sum_{i=1}^{n} \rho\left(\frac{y_i - \mu - x_i^T\beta}{\sigma}\right)\sigma = n\sigma + \sum_{i=1}^{n} \frac{\left(y_i - \mu - x_i^T\beta\right)^2}{\sigma} \tag{24}$$

is convex in $(\mu,\beta,\sigma) \in \mathbb{R}^p \times (0,\infty)$.

Theory of concomitant scale estimation Cont.

Now the equation 24 is minimized by taking μ and β as their least square estimates and $\sigma = \hat{\sigma}$. Where,

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{\mu} - x_i^T \hat{\beta} \right)^2 \tag{25}$$

This gives rise to the usual normal distribution maximum likelihood estimates. It turns out that the equation (24) is not a simple monotone transformation of the negative log likelihood.

$$\frac{n}{2} \log(2\pi) + n \log(\sigma) + \frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(y_i - \mu - x_i^T \beta \right)^2 \tag{26}$$

The negative log-likelihood of the above equation is not convex in σ (for any μ, β). Huber's technique has convexified the Gaussian Log Likelihood 26 into equation (24).

Theory of concomitant scale estimation Cont.

Turning to the least squares case, if we substitute equation (25) in the criterion we get $2\left(\sum_{i=1}^{n}\left(y_i-\mu-x_i^T\beta\right)^2\right)^{1/2}$. A ridge regression for both residuals and coefficients then minimizes

$$\mathcal{R}(\mu,\beta,\tau,\sigma;\lambda)=n\sigma+\sum_{i=1}^{n}\left(\frac{(y_i-\mu-x_i^T\beta)^2}{\sigma}\right)+\lambda\left(p\tau+\sum_{j=1}^{p}\frac{\beta_j^2}{\tau}\right) \tag{27}$$

over $(\mu,\beta,\sigma,\tau)\in\mathbb{R}^{p+1}\times[0,\infty)^2$ for fixed $\lambda\in[0,\infty)$.

Theory of concomitant scale estimation Cont.

Minimizing over σ and τ in closed form leaving

$$\min_{\sigma,\tau} \mathcal{R}(\beta, \tau, \sigma; \lambda) = 2\left(\sum_{i=1}^{n}(y_i - \mu - x_i^T\beta)^2\right)^{1/2} + 2\lambda\left(\sum_{j=1}^{p}\beta_j^2\right)^{1/2} \quad (28)$$

to be minimized on β given λ. In other words, after putting Huber's concomitant scale estimators into ridge regression we recover ridge regression again. The criterion is $||y - \mu - x^T\beta||_2 + \lambda||\beta||_2$ which gives the same trace as $||y - \mu - x^T\beta||_2^2 + \lambda||\beta||_2^2$.

CVXPY implementation

Recall :

$$\mathcal{H}_M(x) = \begin{cases} x^2 & |x| \leq M \\ 2M|x| - M^2 & |x| \geq M \end{cases} \tag{29}$$

The optimisation of Huber function $\mathcal{H}_M(x)$ can be framed as the following quadratic program

$$\begin{aligned} \text{minimize} \quad & u^2 + 2Mv && (30) \\ \text{subject to} \quad & |x| \leq u + v && (31) \\ & u \leq M && (32) \\ & v \geq 0, && (33) \end{aligned}$$

where,

$$u = min(|x|, M) \tag{34}$$

$$v = max(|x| - M, 0) \tag{35}$$

CVXPY implementation contd.

CVXPY has a built-in Huber function, but as we are using concomitant scale estimation, the optimisation of the function $\sigma + \mathcal{H}_M(z/\sigma)\sigma$ is framed as following quadratic program

$$\begin{aligned} \text{minimize} \quad & \sigma + u^2/\sigma + 2Mv && (36) \\ \text{subject to} \quad & |z| \leq u + v && (37) \\ & u \leq M\sigma && (38) \\ & v \geq 0, && (39) \end{aligned}$$

after substituting and simplifying.

CVXPY implementation Cont.

Recall :

$$\mathcal{B}_M(z) = M\mathcal{B}_M\left(\frac{z}{M}\right) = \begin{cases} |z| & |z| \leq M \\ \frac{z^2+M^2}{2M} & |z| \geq M \end{cases} \tag{40}$$

The optimisation of Berhu function $\mathcal{B}_M(x)$ can be framed as the following quadratic program

$$\begin{aligned} \text{minimize} \quad & v + u^2/(2M) + u && (41)\\ \text{subject to} \quad & |x| \leq v + u && (42)\\ & v \leq M && (43)\\ & u \geq 0. && (44) \end{aligned}$$

where,

$$v = min(|x|, M) \tag{45}$$

$$u = max(|x| - M, 0) \tag{46}$$

CVXPY implementation Cont.

As we are using concomitant scale estimation, the optimisation of the function $\tau + \mathcal{B}_M(z/\tau)\tau$ is framed as following quadratic program

$$\begin{aligned} \text{minimize} \quad & \tau + v + u^2/(2M\tau) + u && (47) \\ \text{subject to} \quad & |z| \leq u + v && (48) \\ & v \leq M\tau && (49) \\ & u \geq 0, && (50) \end{aligned}$$

CVXPY implementation Cont.

```
M=1.35
def Robust_Hybrid_Solve(X,y,lam):
    n = y.shape[0]
    p = X.shape[1]
    sig = cp.Variable(1)
    tau = cp.Variable(1)
    mu = cp.Variable(1)
    res = cp.Variable((n,1))
    beta = cp.Variable((p,1))
    u1 = cp.Variable((n,1))
    v1 = cp.Variable((n,1))
    res = y-X@beta-mu
    u2 = cp.Variable((p,1))
    v2 = cp.Variable((p,1))
    objective = cp.Minimize(cp.quad_over_lin(u1,sig)+2*M*cp.sum(v1)+n*sig+lam*(cp.quad_over_lin(u2,2*M*tau)+cp.sum(u2)+cp.sum(v2)+p*tau))
    constraints = []
    constraints.append(cp.abs(res)<=u1+v1)
    constraints.append(u1<=M*sig)
    constraints.append(v1>=0)
    constraints.append(sig>=0)
    constraints.append(cp.abs(beta)<=u2+v2)
    constraints.append(v2<=M*tau)
    constraints.append(u2>=0)
    constraints.append(tau>=0)

    prob = cp.Problem(objective,constraints)
    prob.solve()

    return beta.value,mu.value
```

Figure: CVXPY implementation

Practical application - Diabetes example

We show and compare the results for penalized regression on the diabetes test data. The figures 4, 5 and 6 visualise the coefficient vectors for Lasso, ridge and robust hybrid regressions respectively. In each graph, the coefficient vector starts at $\beta(\infty) = (0, 0, .., 0)$ and grows as one moves from left to right. The code can be found here

Practical application - Diabetes example Cont.

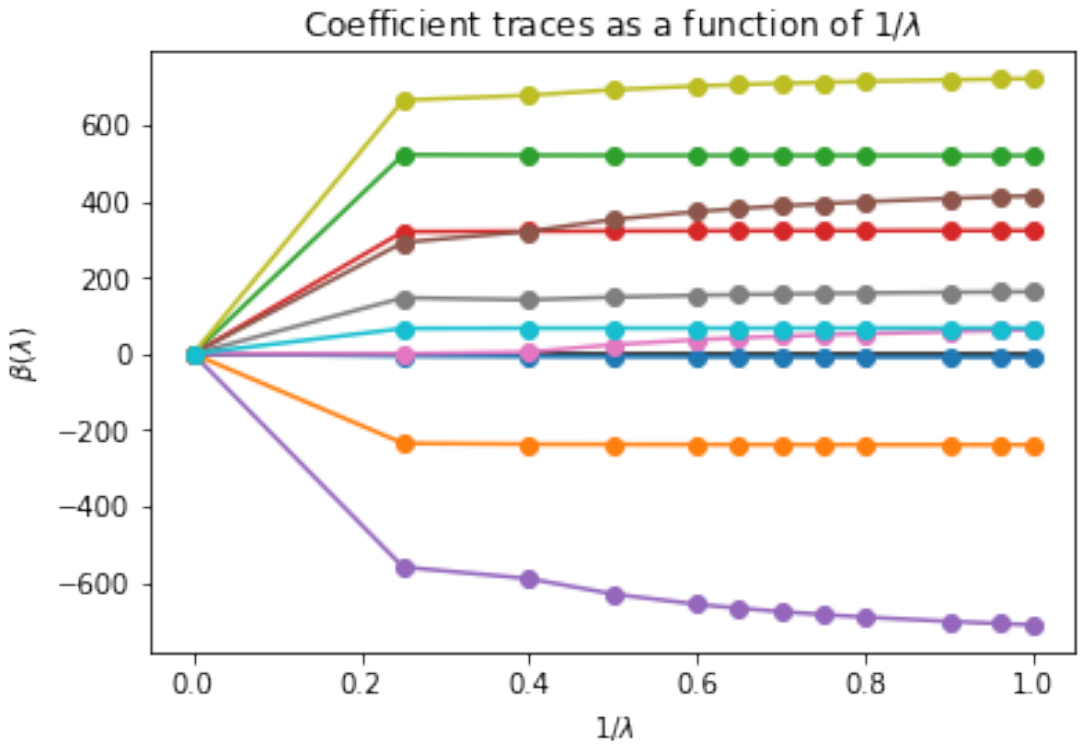

Figure: Lasso

Practical application - Diabetes example Cont.

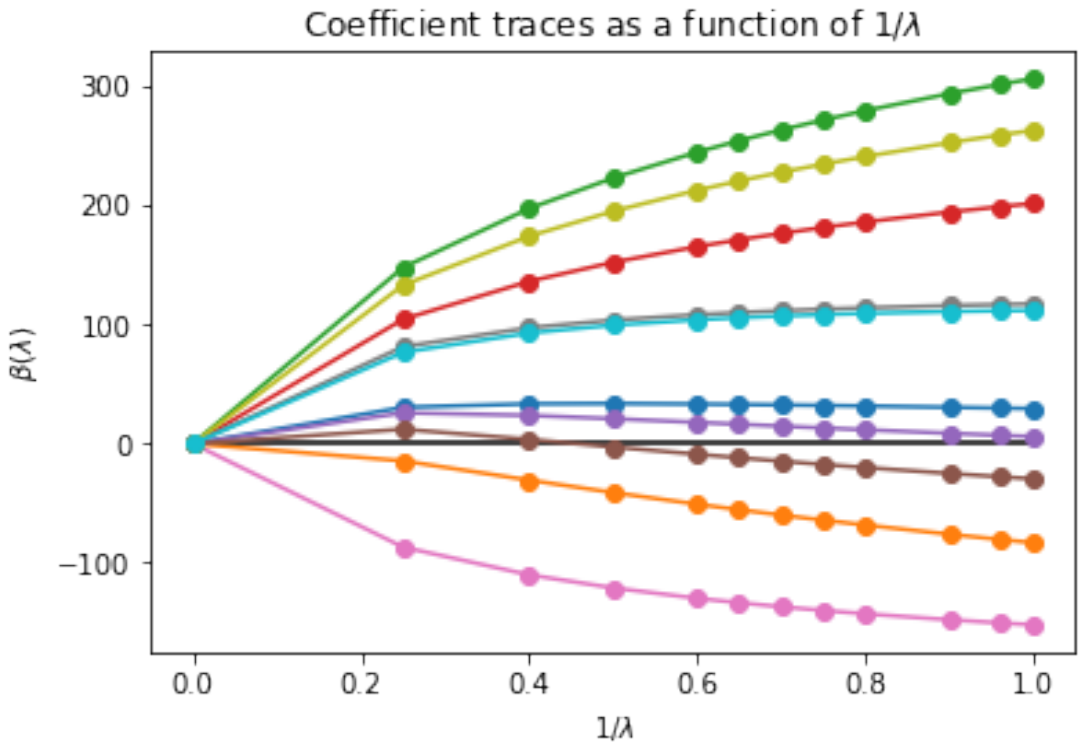

Figure: Ridge

Practical application - Diabetes example Cont.

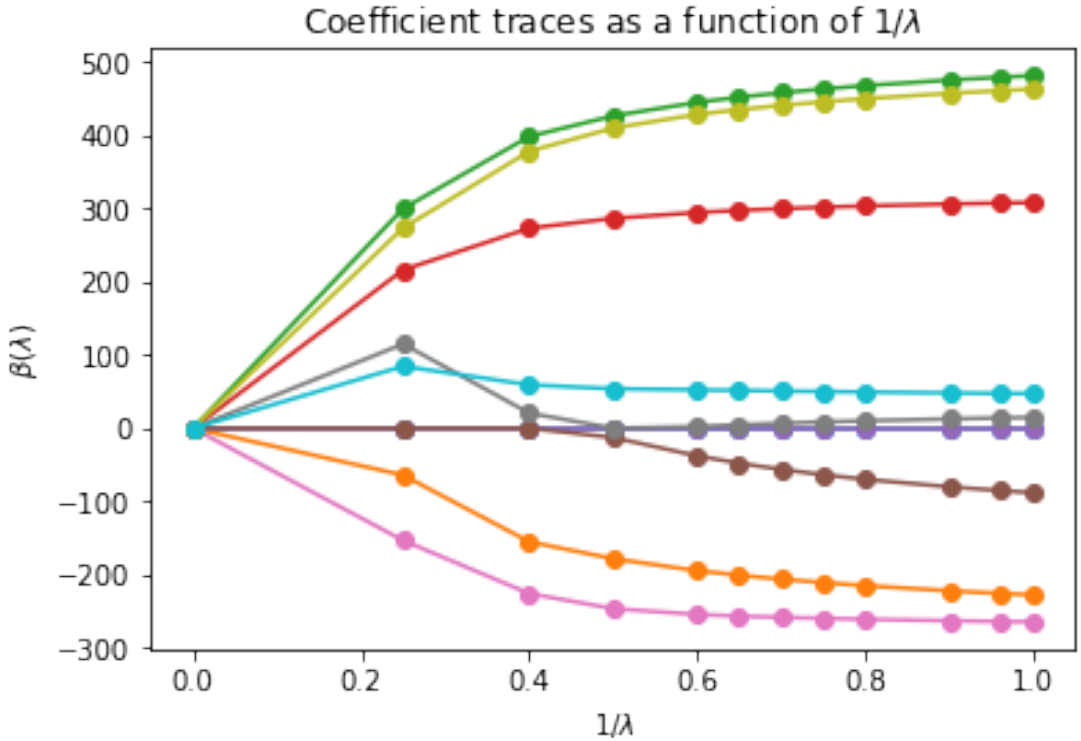

Figure: Robust Hybrid

Conclusion

We note the following observations:

- For lasso regression, we can observe more sparsity.
- For ridge regression, the coefficient vectors are mostly non-sparse
- With the robust hybrid, we get a hybrid behaviour. We get some sparsity, and and some diverse values for β's