

Lasso, Ridge and Elastic Net Regressions

This notebook presents a simple implementation of Lasso and elastic net regressions.

Load Packages and Extra Functions

```
1 using DelimitedFiles, LinearAlgebra, Statistics, Plots, Convex, SCS
```

Loading Data

We use the diabetes data from Efron et al, downloaded from https://web.stanford.edu/~hastie/StatLearnSparsity_files/DATA/diabetes.html and then converted from a tab to a comma delimited file.

All data series are standardised (see below) to have zero means and unit standard deviation, which improves the numerical stability. (Efron et al do not standardise the scale of the response variable.)

```
["AGE", "SEX", "BMI", "BP", "S1", "S2", "S3", "S4", "S5", "S6"]
```

```
1 begin
2   (x, header) = readdlm("base.txt", '\t', header = true)
3   #display(header)
4   #display(x)
5
6   x = (x .- mean(x, dims = 1)) ./ std(x, dims = 1)           #standardise
7
8   (Y, X) = (x[:, end], x[:, 1:end-1]);                      #to get traditional names
9   xNames = header[1:end-1];
10 end
```

Lasso, Ridge and Elastic Net Regressions

(a) The regression is $Y = Xb + u$, where Y and u are $T \times 1$, X is $T \times K$, and b is the K -vector of regression coefficients.

(b) We want to minimize $(Y - Xb)'(Y - Xb)/T + \gamma \sum |b_i| + \lambda \sum b_i^2$.

(c) We can equally well minimise $b'Qb - 2c'b + \gamma \sum |b_i| + \lambda \sum b_i^2$, where $Q = X'X/T$ and $c = X'Y/T$.

(d) Lasso: $\gamma > 0, \lambda = 0$; Ridge: $\gamma = 0, \lambda > 0$; elastic net: $\gamma > 0, \lambda > 0$.

LassoEN

LassoEN(Y,X,γ,λ)

Do Lasso (set $\gamma > 0, \lambda = 0$), ridge (set $\gamma = 0, \lambda > 0$) or elastic net regression (set $\gamma > 0, \lambda > 0$).

Input

- $Y::\text{Vector}$: T-vector with the response (dependent) variable
- $X::\text{VecOrMat}$: TxK matrix of covariates (regressors)
- $\gamma::\text{Number}$: penalty on $\text{sum}(\text{abs}(\mathbf{b}))$
- $\lambda::\text{Number}$: penalty on $\text{sum}(\mathbf{b}^2)$

The next cell makes a Lasso regression for a single value of γ .

```
1 begin
2     K = size(X, 2)
3     γ = 0.25
4
5     (b, b_ls) = LassoEN(Y, X, γ)
6
7     println("OLS and Lasso coeffs (with γ=$γ)")
8     display([[ "OLS" "Lasso"]; xNames b_ls b])
9 end
```

```
OLS and Lasso coeffs (with γ=0.25)
11×3 Matrix{Any}:
 " "      "OLS"      "Lasso"
"AGE"    -0.00618293  8.28707e-5
"SEX"    -0.14813     5.65438e-5
"BMI"     0.3211      0.295132
"BP"      0.200367     0.091037
"S1"     -0.489314     0.000171273
"S2"      0.294474     0.000187834
"S3"      0.0624127    -0.042918
"S4"      0.109369     9.97689e-5
"S5"      0.464049     0.255667
"S6"      0.0417719    0.000106045
```

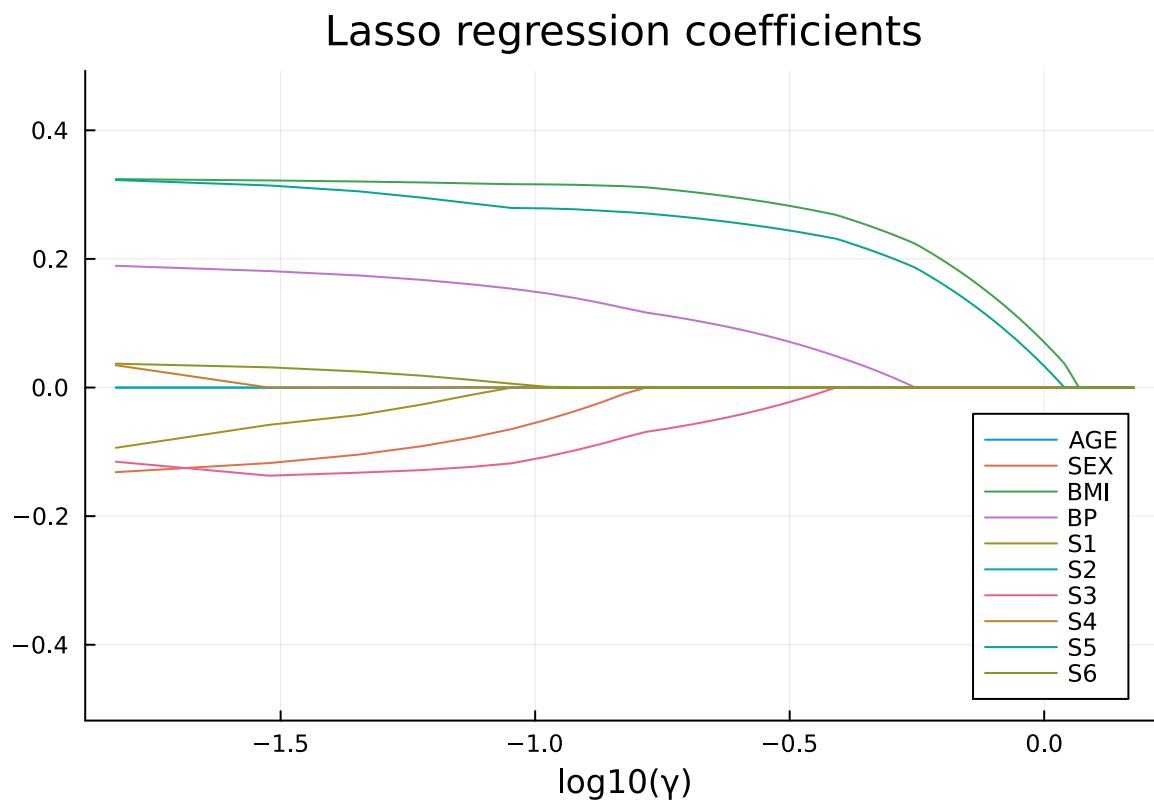


Redo the Lasso Regression with Different Gamma Values

We now loop over γ values.

Remark: it would be quicker to put this loop inside the `LassoEN()` function so as to not recreate L1-L4.

```
1 begin
2   ny = 101
3   yM = range(0; stop = 1.5, length = ny)           #different  $\gamma$  values
4
5   bLasso = fill(NaN, size(X, 2), ny)               #results for  $\gamma[i]$  are in  $bLasso[:,i]$ 
6   for i in 1:ny
7     sol, _ = LassoEN(Y, X, yM[i])
8     bLasso[:, i] .= sol
9   end
10 end
```



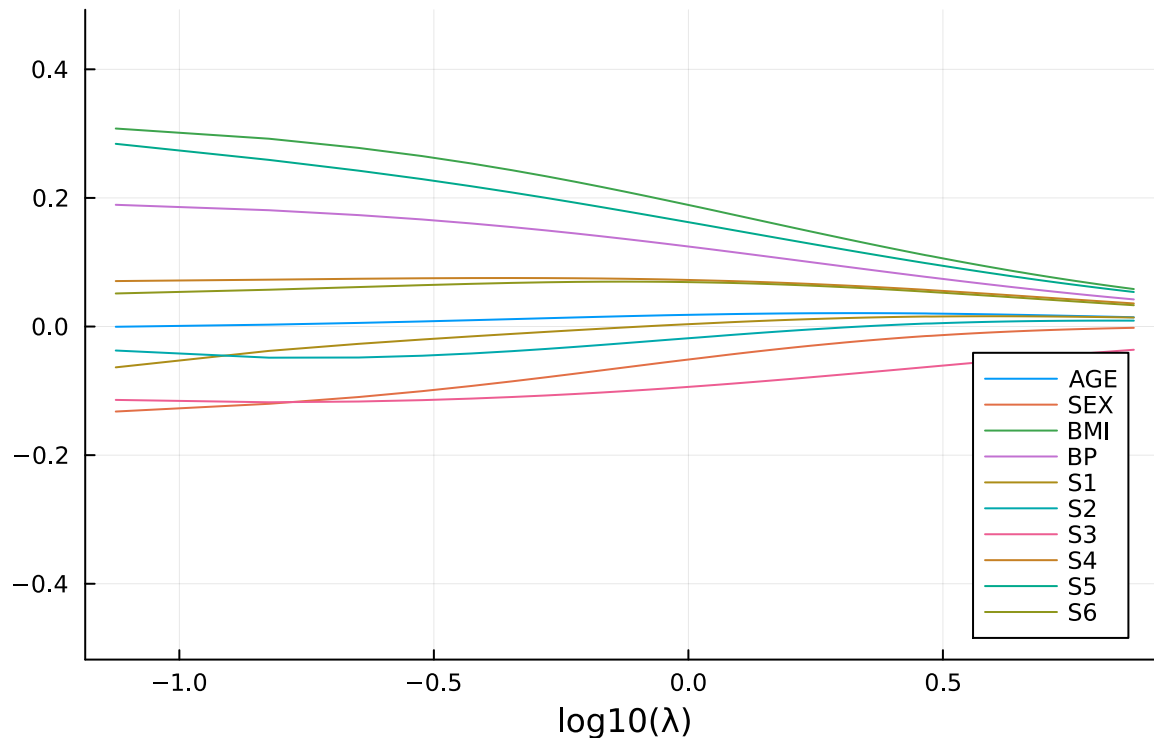
```
1 plot(
2   log10.(yM),
3   bLasso',
4   title = "Lasso regression coefficients",
5   xlabel = "log10( $\gamma$ )",
6   label = permutedims(xNames),
7   size = (600, 400),
8 )
```

Ridge Regression

We use the same function to do a ridge regression. Alternatively, do $b = \text{inv}(X'X/T + \lambda * I) * X'Y/T$.

```
1 begin
2   nλ = 101
3   λM = range(0; stop = 7.5, length = nλ)
4
5   bRidge = fill(NaN, size(X, 2), nλ)
6   for i in 1:nλ
7     sol, _ = LassoEN(Y, X, 0, λM[i])
8     bRidge[:, i] .= sol
9   end
10 end
```

Ridge regression coefficients



```
1 plot(
2   log10.(λM),
3   bRidge',
4   title = "Ridge regression coefficients",
5   xlabel = "log10(λ)",
6   label = permutedims(xNames),
7   size = (600, 400),
8 )
```

Elastic Net Regression

```
1 md" ## Elastic Net Regression"
```

```

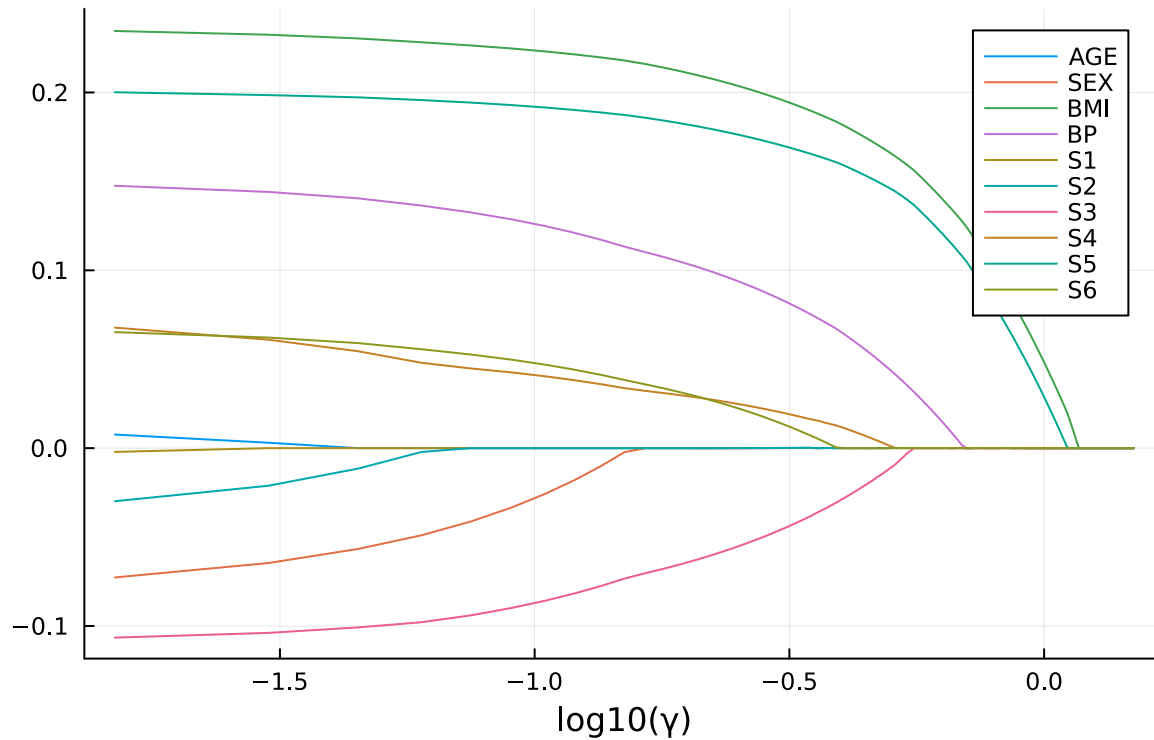
1 begin
2     λ = 0.5
3     println("redo the Lasso regression, but with λ=$λ: an elastic net regression")
4
5     bEN = fill(NaN, size(X, 2), ny)
6     for i in 1:ny
7         sol, _ = LassoEN(Y, X, yM[i], λ)
8         bEN[:, i] .= sol
9     end
10 end

```

redo the Lasso regression, but with $\lambda=0.5$: an elastic net regression



Elastic Net regression coefficients



```

1 plot(
2     log10.(yM),
3     bEN',
4     title = "Elastic Net regression coefficients",
5     xlabel = "log10(γ)",
6     label = permutedims(xNames),
7     size = (600, 400),
8 )

```

