Misc Topics

Jacob M. Montgomery

2018

Cleaning up a few topics you should know

Summary

- Lasso and related
- Handling missing data
- Practicum on R (if time)

LASSO and it's family

- LASSO (least absolute shrinkage and selection operator)
- LASSO Vs. Ridge
- Aside on Elastic Net
- LASSO in Practice

What is it for?

- Variable selection
- Too many co-variates $\rightarrow$ overfitting
- It is another shrinkage method

The LASSO estimate $\hat{\beta}^L_\lambda$ is just regression with an L1 norm penalty:

$$\text{argmin}\left\{\sum_{i=1}(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \sum_{j=1}^{p} \lambda_1|\beta_j|\right\}$$

The LASSO estimate $\hat{\beta}^L_\lambda$ is just regression with an L1 norm penalty:

$$\operatorname{argmin}\left\{\sum_{i=1}(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \sum_{j=1}^{p}\lambda_1|\beta_j|\right\}$$

$$\operatorname{argmin}\left\{RSS + \sum_{j=1}^{p}\lambda_1|\beta_j|\right\}$$

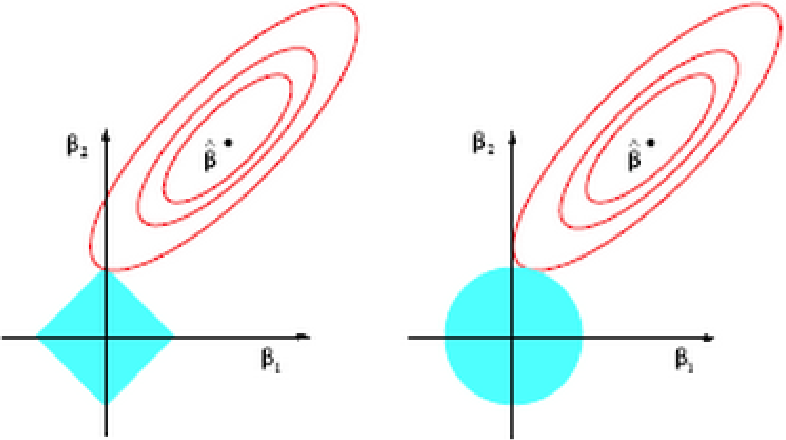
This contrasts with ridge regression . . .

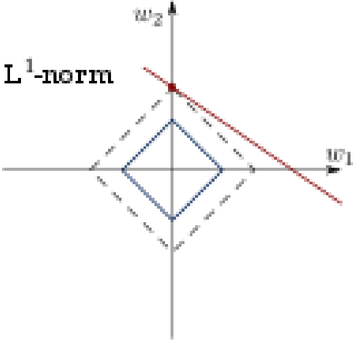
$$\text{argmin}\left\{\sum_{i=1}(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \sum_{j=1}^{p} \lambda_2 |\beta_j|^2\right\}$$

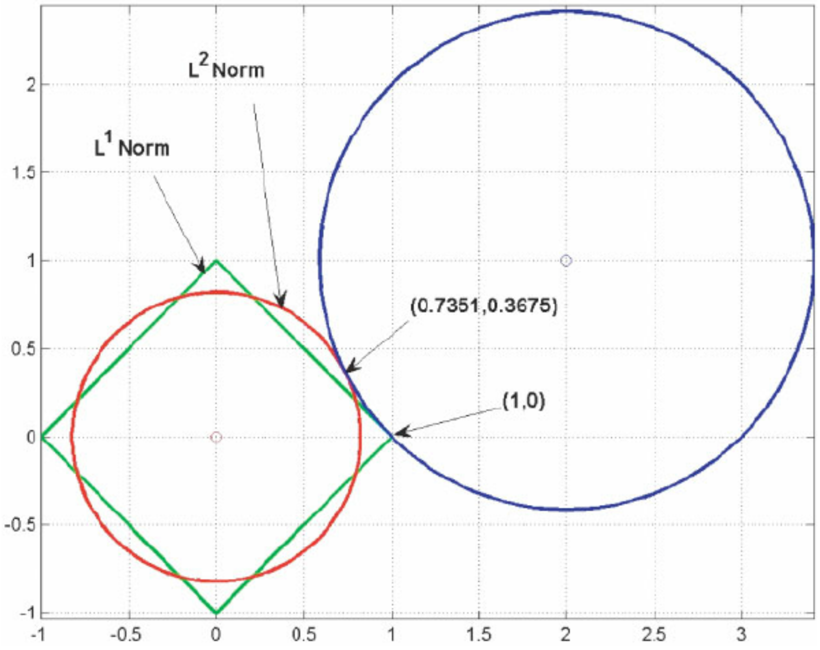
.. and elastic net.

$$\text{argmin}\left\{\sum_{i=1}(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \sum_{j=1}^{p} \lambda_2 |\beta_j|^2 + \sum_{j=1}^{p} \lambda_1 |\beta_j|\right\}$$

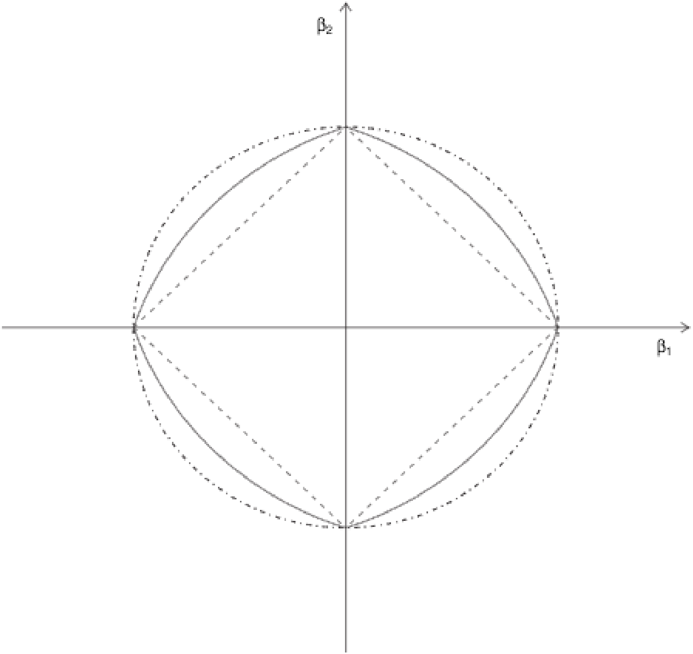
What does this mean conceptually?


β_2
$\hat{\beta}$
β_1
β_2
$\hat{\beta}$
β_1


L¹-norm
w_2
w_1
L²-norm
w_2
w_1



L^2 Norm
L^1 Norm
(0.7351,0.3675)
(1,0)


β2
β1

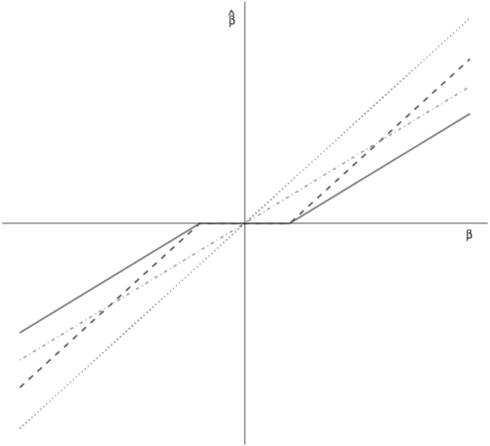


Fig. 2. Exact solutions for the lasso (-------), ridge regression (· - · - · -) and the naïve elastic net (———) in an orthogonal design (·······, OLS): the shrinkage parameters are $\lambda_1 = 2$ and $\lambda_2 = 1$

```r
library(lars)
```

```
## Loaded lars 1.2
```

```r
library(glmnet)
```

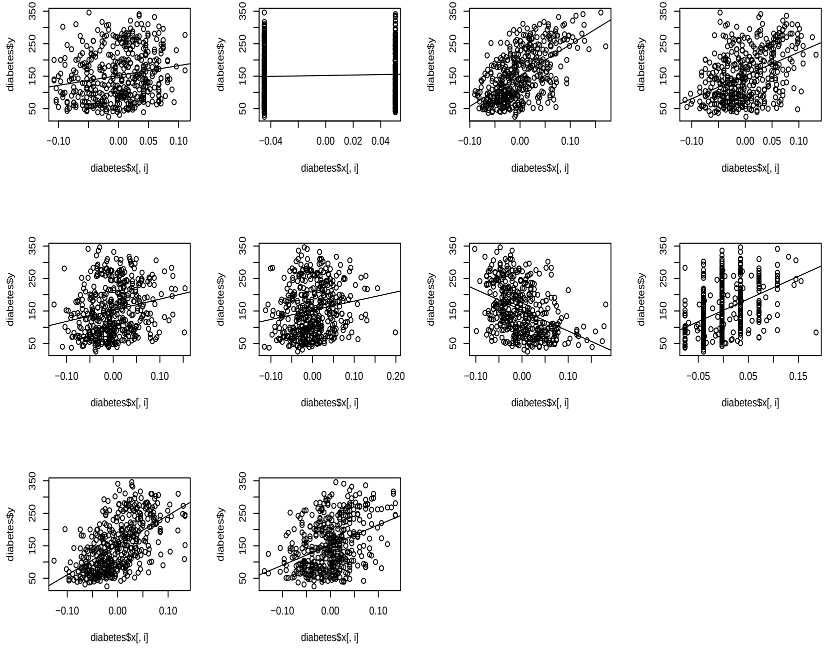
```
## Loading required package: Matrix
```

```
## Loading required package: foreach
```

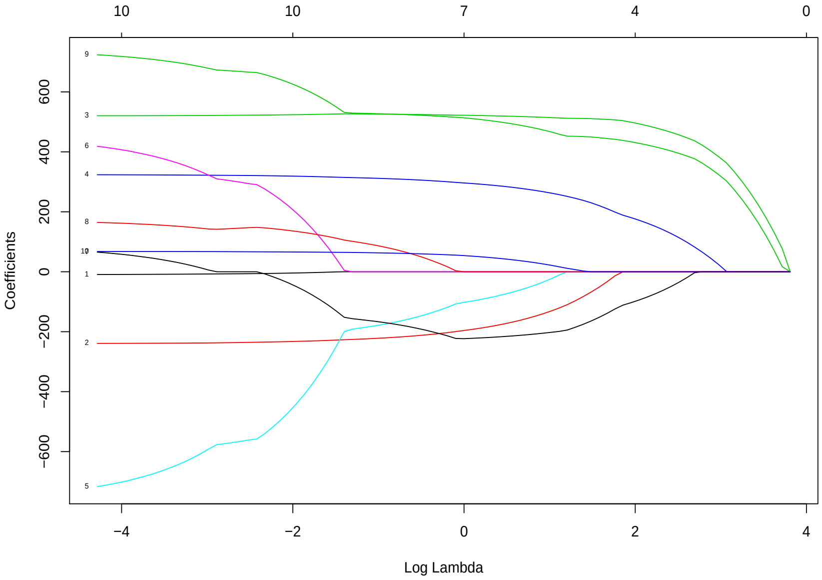
```
## Loaded glmnet 2.0-16
```

```r
data(diabetes)
colnames(diabetes$x)
```

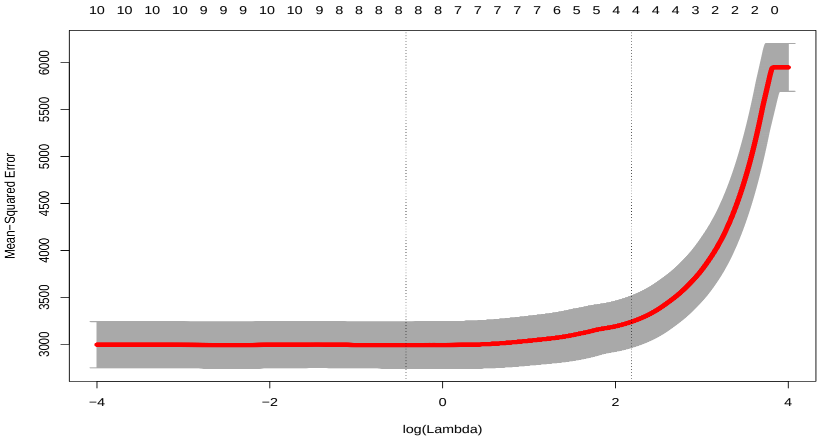
```
##  [1] "age" "sex" "bmi" "map" "tc"  "ldl" "hdl" "tch" "ltg" "glu"
```


diabetes$y
diabetes$x[, i]
50 150 250 350
−0.10 0.00 0.05 0.10
−0.04 0.00 0.02 0.04
−0.10 0.00 0.10
−0.10 0.00 0.05 0.10
−0.10 0.00 0.10
−0.10 0.00 0.10 0.20
−0.10 0.00 0.10
−0.05 0.05 0.15
−0.10 0.00 0.10
−0.10 0.00 0.10

```r
model_lasso <- glmnet(diabetes$x, diabetes$y, alpha=1)
plot.glmnet(model_lasso, xvar="lambda", label=TRUE)
```



```
cv_fit <- cv.glmnet(x=diabetes$x, y=diabetes$y, alpha = 1, nfolds = 10,
                    lambda = exp(seq(-4, 4, by=.001)))
plot.cv.glmnet(cv_fit)
```



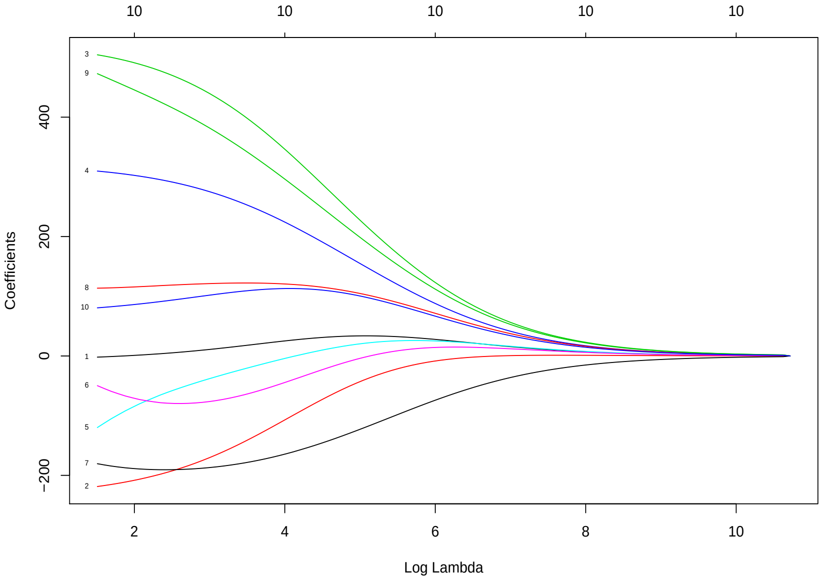

```
cv_fit$lambda.min
```

```
## [1] 0.6544239
```

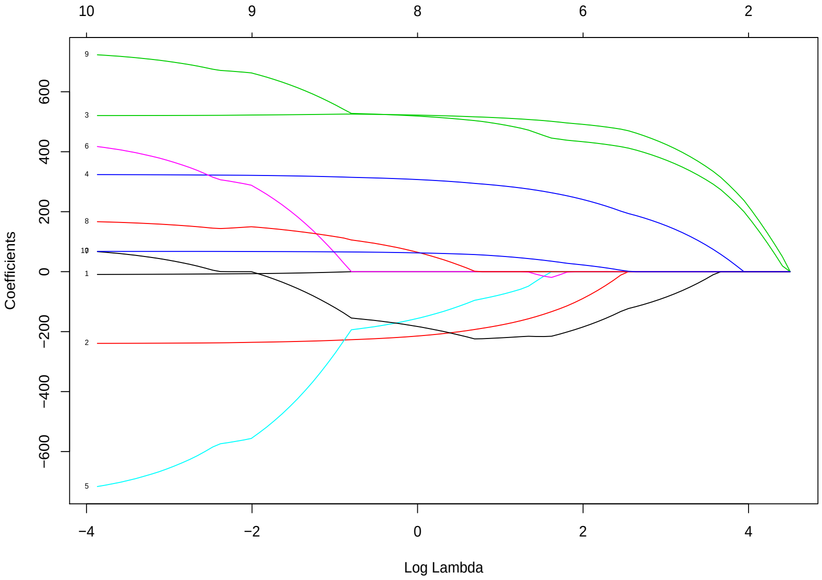
```r
fit <- glmnet(x=diabetes$x, y=diabetes$y, alpha = 1, lambda=cv_fit$lambda.min)
fit$beta
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##             s0
## age    .
## sex -210.05147
## bmi  523.99371
## map  304.47474
## tc  -140.65802
## ldl    .
## hdl -196.33782
## tch   42.46257
## ltg  520.90490
## glu   58.95380
```

```r
model_lasso <- glmnet(diabetes$x, diabetes$y, alpha=0)
plot.glmnet(model_lasso, xvar="lambda", label=TRUE)
```



```r
model_lasso <- glmnet(diabetes$x, diabetes$y, alpha=.5)
plot.glmnet(model_lasso, xvar="lambda", label=TRUE)
```



A bit more

- Algorithm is a version of LARS (Least Angle Regression) and works as:
  1. Start with all coefficients at zero
  2. Find the predictor most correlated with the outcome, and increase coefficient estimate until some other variable more correlated with the residuals.
  3. Proceeds in a direction “equiangular” between the two until a third added.
  4. Repeat.
- Thus, in a weird way is related to the the other procedures

A bit more

- Algorithm is a version of LARS (Least Angle Regression) and works as:
  1. Start with all coefficients at zero
  2. Find the predictor most correlated with the outcome, and increase coefficient estimate until some other variable more correlated with the residuals.
  3. Proceeds in a direction "equiangular" between the two until a third added.
  4. Repeat.
- Thus, in a weird way is related to the the other procedures

Missing data

The vocabulary of missing

R is a matrix that with a dichotomous indicator valued 1 if a datum in **X** is missing and 0 if it is not. The missing data generating mechanism is described by ϕ (Little and Rubin, 2002, p.12)

$$Z_{mis} = (X_{mis}, Y_{mis})$$

$$Z_{obs} = (X_{obs}, Y_{obs})$$

The vocabulary of missing

R is a matrix that with a dichotomous indicator valued 1 if a datum in **X** is missing and 0 if it is not. The missing data generating mechanism is described by ϕ (Little and Rubin, 2002, p.12)

$$Z_{mis} = (X_{mis}, Y_{mis})$$

$$Z_{obs} = (X_{obs}, Y_{obs})$$

Missing completely at random (MCAR) Missingness not related to any observed or unobserved data

$$P(\mathbf{R}|Z_{obs}, Z_{mis}) = P(R|\phi)$$

Missing at random (MAR) Missingness depends only on observed data

$$P(\mathbf{R}|Z_{obs}, Z_{mis}) = P(\mathbf{R}|Z_{obs}, \phi)$$

Non-ignorable (NI) Missingness depends on unobserved data

$$P(\mathbf{R}|Z_{obs}, Z_{mis}) = P(\mathbf{R}|Z_{obs}, Z_{mis}, \phi)$$

Examples

- Missing Completely at Random (MCAR)
  - Respondents accidentally skip questions.

Examples

- Missing Completely at Random (MCAR)
  - Respondents accidentally skip questions. Very unlikely.
- Missing at Random (MAR)
  - Respondents with low political sophistication, low levels of information and education, do not know how to place themselves of the liberal/conservative dimension.
  - Respondents with lower income do not answer questions about their income.

Examples

- Missing Completely at Random (MCAR)
  - Respondents accidentally skip questions. Very unlikely.
- Missing at Random (MAR)
  - Respondents with low political sophistication, low levels of information and education, do not know how to place themselves of the liberal/conservative dimension.
  - Respondents with lower income do not answer questions about their income.
- Non-Ignorable
  - Really . . . everything.

The problem with just dropping

Consider the computation of a mean μ from data $\mathbf{y}$ where some data are non-randomly missing.

When μ_R is the mean of respondents and μ_M is the mean of missing data, we write the overall mean as:

$$\mu = \pi_R \mu_R + (1 - \pi_R)\mu_M$$

where π_R is the *proportion* of observed responses.

The problem with just dropping

Consider the computation of a mean μ from data $\mathbf{y}$ where some data are non-randomly missing.

When μ_R is the mean of respondents and μ_M is the mean of missing data, we write the overall mean as:

$$\mu = \pi_R \mu_R + (1 - \pi_R)\mu_M$$

where π_R is the *proportion* of observed responses.

The bias produced by casewise deletion is the expected fraction of missing data times the difference in means for observed and missing data (Little and Rubin, 2002, p.43):

$$\mu_R - \mu = (1 - \pi_R)(\mu_R - \mu_M)$$

In the special case MCAR, $\mu_R = \mu_M$ and the statistic is unbiased, but this is commonly violated in the social sciences.

Alternative approach – impute(Rubin 1979)

Steps:

1. generate reasonable values for the missing the data (Z_{mis}) m times to get m replicate datasets,

Alternative approach – impute(Rubin 1979)

Steps:

1. generate reasonable values for the missing the data (Z_{mis}) m times to get m replicate datasets,
2. analyze/regress each dataset separately,

Alternative approach – impute(Rubin 1979)

Steps:

1. generate reasonable values for the missing the data (Z_{mis}) m times to get m replicate datasets,
2. analyze/regress each dataset separately,
3. combine results with summary process.

- Imputation step assumes missing data is conditioned on observed values.
- Oddly, enough m $=$ 5 to 10 is sufficient.
- Combining process uses means for coefficients and an intuitive approach for standard errors.

Practicel issues

- Pre-specify if a variable is a factor, numeric, etc. Read manuals.

Practicel issues

- Pre-specify if a variable is a factor, numeric, etc. Read manuals.
- Dataset cannot have perfectly collinear variables. e.g, A variable with country names and another variable with country IDs.

Practicel issues

- Pre-specify if a variable is a factor, numeric, etc. Read manuals.
- Dataset cannot have perfectly collinear variables. e.g, A variable with country names and another variable with country IDs.
- Dataset for imputation usually has more variables than the model we want to specify.

Practicel issues

- Pre-specify if a variable is a factor, numeric, etc. Read manuals.
- Dataset cannot have perfectly collinear variables. e.g, A variable with country names and another variable with country IDs.
- Dataset for imputation usually has more variables than the model we want to specify.
- “An imputation model does not represent causal relationships among the data." (Young and Johnson 2010)

How to actually do this in R

- `mice` (van Buuren et al. 2006, van Buuren 2007)
- `Amelia` (King and others)
- `mi` for multilevel data (Gelman, others)
- `hot.deck` for categorical variables (Cranmer and Gill 2013)

How to actually do this in R

- `mice` (van Buuren et al. 2006, van Buuren 2007)
- `Amelia` (King and others)
- `mi` for multilevel data (Gelman, others)
- `hot.deck` for categorical variables (Cranmer and Gill 2013)
- Lot's of predictive algorithms
- `rfImpute` using random forests for imputation