MA8701 Advanced methods in statistical inference and learning

L3: Shrinkage - algorithm, variants, GLM

Mette Langaas IMF/NTNU

23 January, 2021

Shrinkage - second act

Literature L3

- [ELS] The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics, 2009) by Trevor Hastie, Robert Tibshirani, and Jerome Friedman. Ebook. Chapter 4.4.1-4.4.3 (4.4.4 is covered in 3.2 of HTW).
- [HTW] Hastie, Tibshirani, Wainwrigh: "Statistical Learning with Sparsity: The Lasso and Generalizations". CRC press. Ebook. Chapter 2.4 (understanding from 5.4 will be presented - but not on reading list), 3.1-3.2,3.7 4.1-4.3,4.5-4.6 (but only from a practical view for ch 4)

Lasso

What do we know from L2?

We forgot to say that

- the acronym is *Least Absolute Shrinkage and Selection Operator*, and that the
- lasso was invented by Robert Tibshirani and published in an article in JRSSB in 1996

We still work on the linear regression case - with continuous response.

Computations of the lasso solutions

(HTW 2.4)

- Focus is on the Coordinate descent algorithm, where
- soft thresholding plays an important role.
- Also mention the concept of subgradients (from HTW 5.4 - not on the reading list).

See slides from guest lecturer Benjamin Dunn

Group discussion:

Write down in pseudo code the steps of the cyclic coordinate descent algorithm.

Coordinate descent

(Result HTW page 110): Additive function to minimize:

$$f(\beta) = g(\beta) + \sum_{j=1}^{p} h_j(\beta_j)$$

g differentiable and convex, h univariate and convex. It is found that the coordinate descent algorithm is *guaranteed to converge* to the global minimizer.

Generalizations of the lasso penalty

(HTW 4.1-4.3, 4.5-4.6: NB only from a practical point of view)

See slides from guest lecturer Benjamin Dunn

The main goal of this part is to

- know about these special versions of the lasso, and
- to see which practical data situation these can be smart to use.

Maybe one of these is suitable for Data analysis project 1?

Theoretical properties and algorithmic details are not on the reading list.

Group activity:

(choose one variant to work with)

For the lasso variants

- the elastic net [HTW 4.2]
- the group lasso [HTW 4.3]
- the fused lasso [HTW 4.5]

write down

- which variation on the classic lasso penalty is used (write down the penalty part of the minimization problem)
- make a drawing of the penalty (comparable to the sphere for ridge and the diamond for lasso)
- in which practical data analysis situation is this variation used (e.g. when many correlated variables are present, when the covariates have a natural group structure, . . .)
- anything else you found interesting?

Generalized linear models

(HTW 3.1, 3.2, and TMA4315 GLM background)

The model

The GLM model has three ingredients:

1) Random component
2) Systematic component
3) Link function

We look into that for the normal and binomial distribution - to get multiple linear regression and logistic regression.

- Write in class
- Poll on standardization and centering.

Explaining β in logistic regression

- The ratio $\frac{P(Y_i=1)}{P(Y_i=0)} = \frac{\pi_i}{1-\pi_1}$ is called the *odds*.
- If $\pi_i = \frac{1}{2}$ then the odds is 1, and if $\pi_i = \frac{1}{4}$ then the odds is $\frac{1}{3}$. We may make a table for probability vs. odds in R:

pivec	0.10	0.20	0.30	0.40	0.5	0.6	0.70	0.8	0.9
odds	0.11	0.25	0.43	0.67	1.0	1.5	2.33	4.0	9.0

- Odds may be seen to be a better scale than probability to represent chance, and is used in betting. In addition, odds are unbounded above.

We look at the link function (inverse of the response function). Let us assume that our linear predictor has k covariates present

$$\eta_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik}$$

$$\pi_i = \frac{\exp(\eta_i)}{1 + \exp(\eta_i)}$$

$$\eta_i = \ln(\frac{\pi_i}{1 - \pi_i})$$

$$\ln(\frac{\pi_i}{1 - \pi_i}) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik}$$

$$\frac{\pi_i}{1 - \pi_i} = \frac{P(Y_i = 1)}{P(Y_i = 0)} = \exp(\beta_0) \cdot \exp(\beta_1 x_{i1}) \cdots \exp(\beta_k x_{ik})$$

We have a *multiplicative model* for the odds.

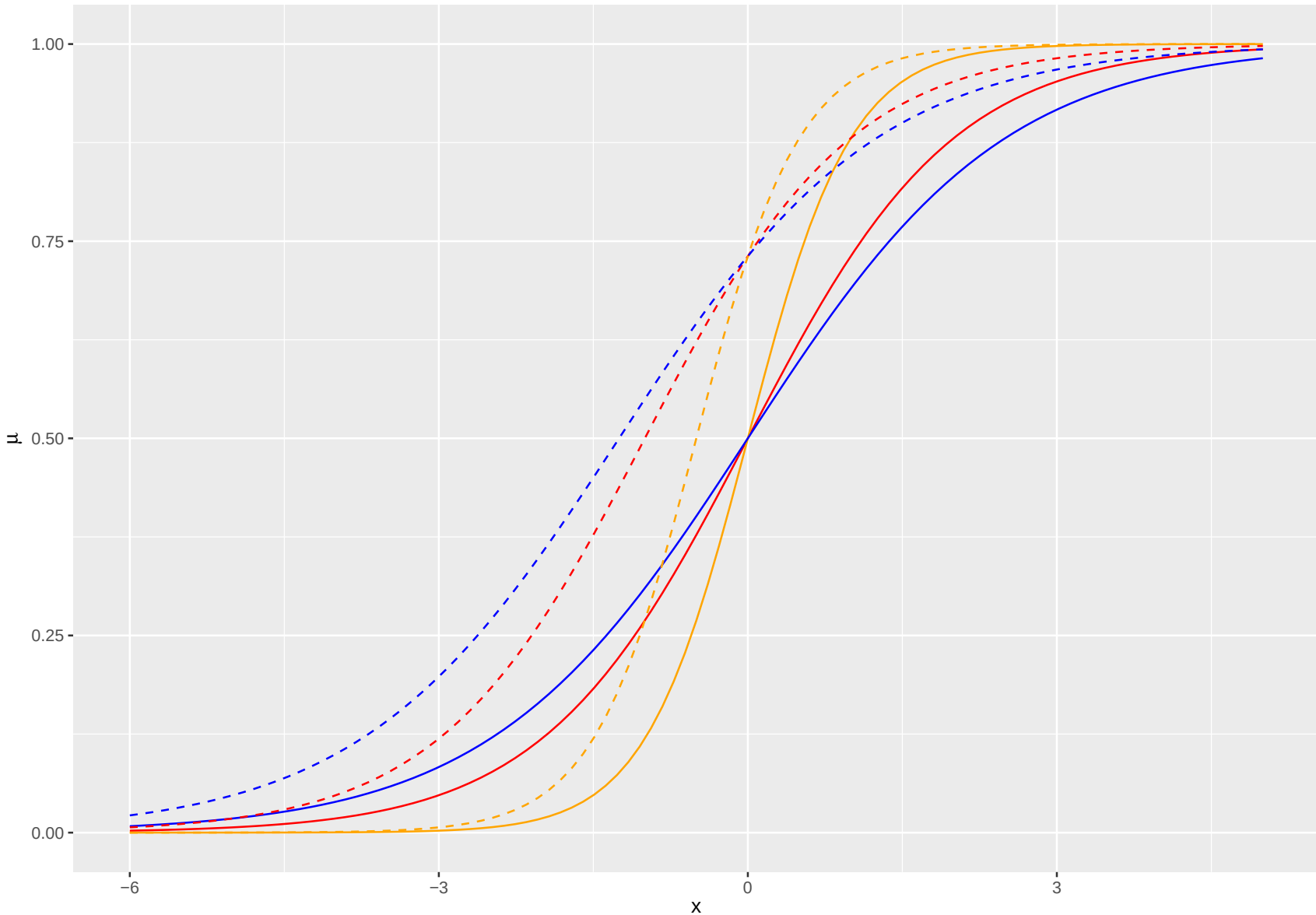
So, what if we increase x_{1i} to $x_{1i} + 1$?

If the covariate x_{1i} increases by one unit (while all other covariates are kept fixed) then the odds is multiplied by $\exp(\beta_1)$:

$$\begin{aligned}
\frac{P(Y_i = 1 \mid x_{i1} + 1)}{P(Y_i = 0) \mid x_{i1} + 1)} &= \exp(\beta_0) \cdot \exp(\beta_1(x_{i1} + 1)) \cdots \exp(\beta_k x_{ik}) \\
&= \exp(\beta_0) \cdot \exp(\beta_1 x_{i1}) \exp(\beta_1) \cdots \exp(\beta_k x_{ik}) \\
&= \frac{P(Y_i = 1 \mid x_{i1})}{P(Y_i = 0 \mid x_{i1})} \cdot \exp(\beta_1)
\end{aligned}$$

This means that if x_{i1} increases by 1 then: if $\beta_1 < 0$ we get a decrease in the odds, if $\beta_1 = 0$ no change, and if $\beta_1 > 0$ we have an increase. In the logit model $\exp(\beta_1)$ is easier to interpret than β_1.

The response function as a function of the covariate x and not of η. Solid lines: $\beta_0 = 0$ and β_1 is 0.8 (blue), 1 (red) and 2 (orange), and dashed lines with $\beta_0 = 1$.



Parameter estimation

- Maximum likelihood estimation $=$ maximize the likelihood of the data $\mathcal{L}(\beta_\prime, \beta; \mathbf{y}, \mathbf{X})$.
- For penalized method we instead minimize the negative loglikelihood scaled with $\frac{1}{N}$.
- The ridge and lasso penalty is added to the scaled negative loglikelihood.
- We write this out for the normal and binomial distribution.
- Write in class

Algorithms

- The likelihood for the GLM is differentiable, and the ridge and lasso objective functions are convex - and can be solved with socalled "standard convex optimization methods".
- But, by popular demand also special algorithms are available - building on the cyclic coordinate descent.

To understand the (ridge and) lasso logistic regression we first look at the *iteratively reweighted least squares* (IRLS) - as a result of the Newton Raphson method.

Lasso logistic regression fitting algoritm

(HTW page 116)

```
OUTER LOOP: start with lambdamax and decrement

      MIDDLE LOOP (with warm start)

         compute quadratic approximation Q(beta0,beta)
         for current beta-estimates



              INNER LOOP: cyclic coordinate descent
              to minimize Q added the lasso penalty
```

Criteria for choosing λ

We use cross-validation to choose λ.

For regression we choose λ by minimizing the (mean) squared error.

For (ridge and) lasso logistic regression we may choose:

- misclassification error rate on the validation set
- ROC-AUC
- binomial deviance

Confusion matrix, sensitivity, specificity

(from TMA4268)

In a two class problem - assume the classes are labelled "-" (non disease,0) and "+" (disease,1). In a population setting we define the following event and associated number of observations.

	Predicted -	Predicted +	Total
True -	True Negative TN	False Positive FP	N
True +	False Negative FN	True Positive TP	P
Total	N*	P*	

(N in this context not to be confused with our sample size. . .)

Sensitivity (recall) is the proportion of correctly classified positive observations: $\frac{\#\text{True Positive}}{\#\text{Condition Positive}} = \frac{\text{TP}}{\text{P}}$.

Specificity is the proportion of correctly classified negative observations: $\frac{\#\text{True Negative}}{\#\text{Condition Negative}} = \frac{\text{TN}}{\text{N}}$.

We would like that a classification rule have both a high sensitivity and a high specificity.

Other useful quantities:

Name	Definition	Synonoms
False positive rate	FP/N	Type I error, 1-specificity
True positive rate	TP/P	1-Type II error, power, sensitivity, recall
Positive predictive value (PPV)	TP/P*	Precision, 1-false discovery proportion
Negative predictive value (NPV)	TN/N*	

Where the PPV can be used together with the sensitivity to make a precision-recall curve more suitable for low case rates.

ROC curves

(also from TMA4268)

The receiver operating characteristics (ROC) curve gives a graphical display of the sensitivity against specificity, as the threshold value (cut-off on probability of success or disease) is moved over the range of all possible values. An ideal classifier will give a ROC curve which hugs the top left corner, while a straight line represents a classifier with a random guess of the outcome.

ROC-AUC

- The **ROC-AUC** score is the area under the ROC curve. It ranges between the values 0 and 1, where a higher value indicates a better classifier.
- The AUC score is useful for comparing the performance of different classifiers, as all possible threshold values are taken into account.
- The ROC-AUC is closely connected to the robust U statistics.
- If the prevalence (case proportion) is very low (0.01ish), the ROC-AUC may be misleading, and the PR-AUC is more commonly used.

Deviance

The *deviance* is based on the likelihood ratio test statistic.
The derivation assumes that data can be grouped into covariate patterns, with G groups (else interval solutions are used in practice).
Saturated model: If we were to provide a perfect fit to our data then we would estimate π_j by the observed frequency for the group, $\hat{y}_j = y_j$.
Candidate model: the model with the current choice of λ.

$$D_\lambda = 2(l(\text{saturated model}) - l(\text{candidate model}_\lambda))$$

The **null deviance** is replacing the candidate model with a model where $\hat{y}_i = \frac{1}{N}\sum_{i=1}^{N} y_i$ (the case proportion).

Example: South African heart disease

(ELS 4.4.2)

Group discussion: Comment on what is done and the results. Where are the CIs and p-values for the ridge and lasso version?

Data set

The data is presented in ELS Section 4.4.2, and downloaded from http://statweb.stanford.edu/~tibs/ElemStatLearn.1stEd/ with information in the file `SAheat.info` and data in `SAheart.data`.

- This is a retrospective sample of males in a heart-disease high-risk region in South Africa. *It consists of 462 observations on the 10 variables. All subjects are male in the age range 15-64.
- There are 160 cases (individuals who have suffered from a conorary heart disease) and 302 controls (individuals who have not suffered from a conorary heart disease).
- The overall prevalence in the region was 5.1%.

The response value (`chd`) and covariates

- `chd` : conorary heart disease {yes, no} coded by the numbers {1, 0}
- `sbp` : systolic blood pressure
- `tobacco` : cumulative tobacco (kg)
- `ldl` : low density lipoprotein cholesterol
- `adiposity` : a numeric vector
- `famhist` : family history of heart disease. Categorical variable

Data description

We start by loading and looking at the data:

```
ds=read.csv("./SAheart.data",sep=",")[,-1]
ds$chd=as.factor(ds$chd)
ds$famhist=as.factor(ds$famhist)
dim(ds)
```

```
## [1] 462  10
```

```
colnames(ds)
```

```
##  [1] "sbp"       "tobacco"   "ldl"       "adiposity" "fa
##  [7] "obesity"   "alcohol"   "age"       "chd"
```

```
head(ds)
```

```
##   sbp tobacco  ldl adiposity famhist typea obesity alcoh
## 1 160   12.00 5.73     23.11 Present    49   25.30    97.
## 2 144    0.01 4.41     28.61  Absent    55   28.87     2.
## 3 118    0.08 3.48     32.28 Present    52   29.14     3.
## 4 170    7.50 6.41     38.03 Present    51   31.99    24.
## 5 134   13.60 3.50     27.78 Present    60   25.99    57.
## 6 132    6.20 6.47     36.21 Present    62   30.77    14.
```

Logistic regression

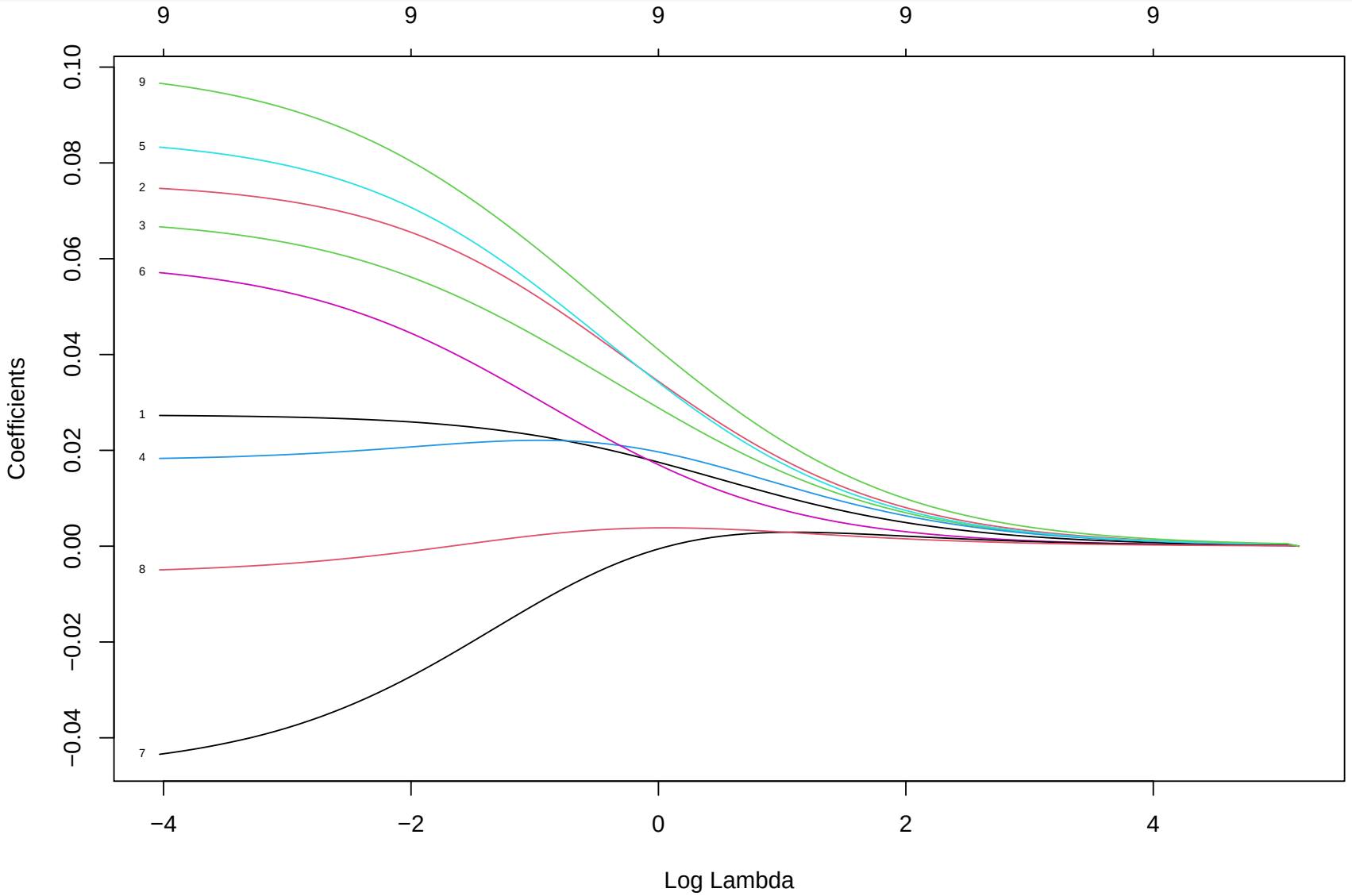
We now fit a (multiple) logistic regression model using the `glm` function and the full data set. In order to fit a logistic model, the `family` argument must be set equal to `="binomial"`. The `summary` function prints out the estimates of the coefficients, their standard errors and z-values. As for a linear regression model, the significant coefficients are indicated by stars where the significant codes are included in the `R` printout.

```
glm_heart = glm(chd~.,data=dss, family="binomial")
summary(glm_heart)
```

```
##
## Call:
## glm(formula = chd ~ ., family = "binomial", data = dss)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7781  -0.8213  -0.4387   0.8889   2.5435
##
## Coefficients:
```

Ridge logistic regression

```
ridgefit=glmnet(x=xss,y=ys,alpha=0,standardize=FALSE) # al
plot(ridgefit,xvar="lambda",label=TRUE)
```




```
cv.ridge=cv.glmnet(x=xss,y=ys,alpha=0)
```

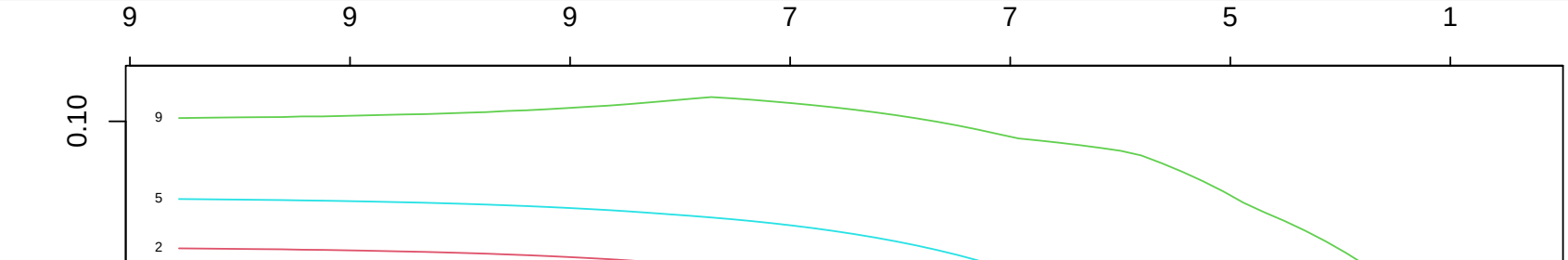
Lasso logistic regression

Numbering in plots is order of covariates, so:

```
cbind(1:9,colnames(xss))
```

```
##       [,1] [,2]
##  [1,] "1"  "sbp"
##  [2,] "2"  "tobacco"
##  [3,] "3"  "ldl"
##  [4,] "4"  "adiposity"
##  [5,] "5"  "famhistPresent"
##  [6,] "6"  "typea"
##  [7,] "7"  "obesity"
##  [8,] "8"  "alcohol"
##  [9,] "9"  "age"
```

```
lassofit=glmnet(x=xss,y=ys,alpha=1,standardize=FALSE) # al
plot(lassofit,xvar="lambda",label=TRUE)
```



Computational details for the glmnet implementation

(HTW 3.7)

`glmnet` is the implementation in R of the elastic net from HTW-book, and the package is maintained by Trevor Hastie.

The package fits generalized linear models using penalized maximum likelihood of elastic net type (lasso and ridge are special cases).

The logistic lasso is fitted using a quadratic approximation for the negative log-likelihood in a "proximal-Newton iterative approach".

Software links

- R glmnet on CRAN with resources.
  - Getting started
  - GLM with glmnet

For Python there are different options.

- Python glmnet is recommended by Hastie et al.
- scikit-learn (seems to mostly be for regression? is there lasso for classification here?)

glmnet inputs

```
glmnet(x, y,
 family = c("gaussian", "binomial", "poisson", "multinomial
 weights = NULL, offset = NULL, alpha = 1, nlambda = 100,
 lambda.min.ratio = ifelse(nobs < nvars, 0.01, 1e-04),
 lambda = NULL, standardize = TRUE, intercept = TRUE,
 thresh = 1e-07, dfmax = nvars + 1,
 pmax = min(dfmax * 2 + 20, nvars),
 exclude = NULL, penalty.factor = rep(1, nvars),
 lower.limits = -Inf, upper.limits = Inf, maxit = 1e+05,
 type.gaussian = ifelse(nvars < 500, "covariance", "naive"
 type.logistic = c("Newton", "modified.Newton"),
 standardize.response = FALSE,
 type.multinomial = c("ungrouped","grouped"),
 relax = FALSE, trace.it = 0, ...)
```

cv.glmnet inputs

```
cv.glmnet(x, y, weights = NULL, offset = NULL, lambda = NUL
  type.measure = c("default", "mse", "deviance", "class", "
  nfolds = 10, foldid = NULL,
  alignment = c("lambda", "fraction"), grouped = TRUE,
  keep = FALSE, parallel = FALSE,
  gamma = c(0, 0.25, 0.5, 0.75, 1), relax = FALSE, trace.it
```

type.measure defaults to deviance (accoring to help(cv.glmnet)).
The last is for Cox models.

Family

we have only covered `gaussian` (the default) and `binomial`. Each family has implemented the deviance measure. Poisson regression and Cox proportional hazard (survival analysis) is also implemented in glmnet.

Penalties

The elastic net is implemented, with three possible adjustment parameters.

$$\text{minimize}_{\beta_0,\beta}\{-\frac{1}{N}l(y;\beta_0,\beta)+\lambda\sum_{j=1}^{p}\gamma_j((1-\alpha)\beta_j^2+\alpha|\beta_j|)\}$$

- λ: the penalty, default a grid of 100 values is chosen, to cover the lasso path on the log scale.
- α: elastic net parameter $\in [0,1]$. This is usually manually selected by a grid search over 3-5 values. Default is $\alpha = 1$ (lasso), and with $\alpha = 0$ we get ridge.
- γ_j: penalty modifier for each covariate to be able to always include ($\gamma_j == 0$), or exclude ($\gamma_j = \text{Inf}$), or give individual penalty modifications. Default $\lambda_j = 1$.

For the λ penalty the maximal value is for

- linear regression: $\lambda_{max} = \max_j |\hat{\beta}_{LS,j}|$ (standardized coefficients) or, should there also be a factor 1/N?
- logistic regression: $\lambda_{max} = \max_j |\mathbf{x}_j^T(\mathbf{y} - \bar{p})|$ where $\bar{p}$ is the mean case rate.

Additional modifications

- Coefficient bounds can be set (possible since coordinate descent is used)
- Some coefficients can be excluded from the penalization (than thus forced in).
- Offset can be added (popular if rate models for Poisson is used)
- For binary and multinomial data factors or matrices can be input.
- Sparse matrices with covariates can be supplied.

Lasso variants

Elastic net is already in glmnet (alpha-parameter).

Other lasso variants have their own R packages:

- The group lasso https://cran.r-project.org/web/packages/grplasso/grplasso.pdf
- The fused lasso https://cran.r-project.org/web/packages/genlasso/genlasso.pdf
- Bayesian lasso blasso function for normal data in package monomvn https://rdrr.io/cran/monomvn/man/monomvn-package.html
- Elastic net for ordinal data: https://cran.r-project.org/web/packages/ordinalNet/ordinalNet.pdf

Exercises

This week the best way to spend the time is to work on the Data Analysis Project 1.

But, also good to study the R-code for the South African heart disease example, and make some changes.

Smart: save this file as an .Rmd file and then run `purl(file.Rmd)` to produce a file with only the R-commands. (At the html-version you choose Code-Download Rmd on the top of the file).

- Change the CV criterion to auc and to class. Are there changes?

References

- Robert Tibshirani Regression Shrinkage and Selection via the Lasso, Journal of the Royal Statistical Society. Series B (Methodological) Vol. 58, No. 1 (1996), pp. 267-288 (22 pages)
- Lecture notes on ridge regression: Welle N. van Wieringen