# EC999: Variable Selection

Thiemo Fetzer

University of Chicago & University of Warwick

February 7, 2017

# Note on Variable Selection

- Last week we spoke about Logistic regression and the possibility to do a penalized estimation, e.g. using Lasso.
- I wanted to briefly revisit this here.

$$p(X\beta) = \frac{e^{\beta_0 + \sum_{k=1}^{p} \beta_k X_k}}{1 + e^{\beta_0 + \sum_{k=1}^{p} \beta_k X_k}} = \frac{e^{X\beta}}{1 + e^{X\beta}}$$

# Best Subset Selection

In estimating the $\hat{P}(Y = y|X)$, we typically have a whole range of regressor variables available to us.
Not all variables $X$, here, some word features are good predictors. In fact, the occurrence of some word features may be specific due to the random draw of the training data set.
Ideally, we "try out" all possible combinations of features $X$ - this approach is known as *Best Subset Selection*.

# Best subset selection

- We can express the best subset selection problem as a nonconvex and combinatorial optimization problem.
- The objective is to find the optimal $s$

$$max_{\beta} \sum_{i=1}^{n} y_i \log(p(x_i'\beta)) + (1-y_i)\log((1-p(x_i'\beta))) \text{ subject to } \sum_{j=1}^{p} \mathbf{I}(\beta_j \tag{1}$$

- This requires that the optimial solution involves finding a vector $\beta$ such that the maximum likelihood is minimal and no more than $s$ coefficients are non-zero.
- The best subset selection approach would try out all possible combinations of regressors such that the above is maximized. This can result in *non nested* models.

# Regularized Lasso

Lasso approximates this optimization problem

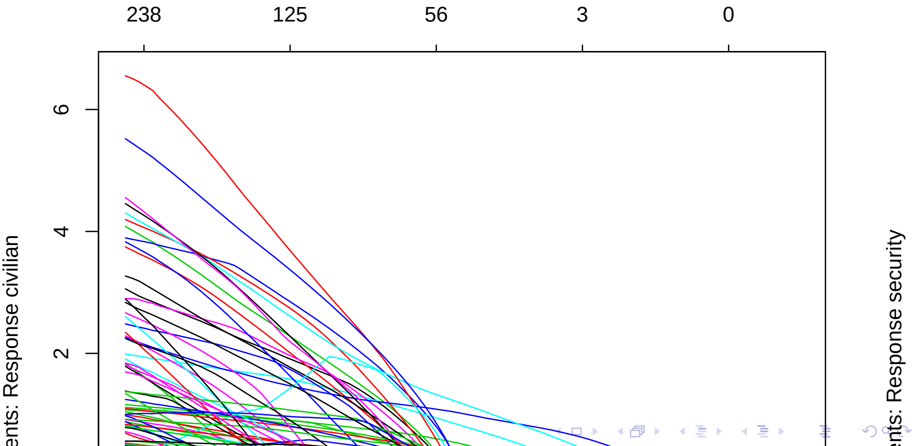
$$min_{\beta} \underbrace{\sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2}_{\text{Residual Sum of Squares}} \text{ subject to } \sum_{j=1}^{p} \mid \beta_j \mid \leq s \tag{2}$$

# An illustration

```
##just look at one word feature
L1 <- create_matrix(c(TTIM[,paste(objectcleanpp,sep=" ")]),
                   language="english",stemWords=FALSE)
##CREATION OF NON SPARSE MATRIX
DTM<-as.matrix(L1)

dim(DTM)


## [1] 1397 2460


##changing column names
colnames(DTM) <- paste("stem_", colnames(DTM),sep="")

##turn this into a document-term-incidence matrix
DTM<-apply(DTM, 2, function(x) as.factor(x>0))
```

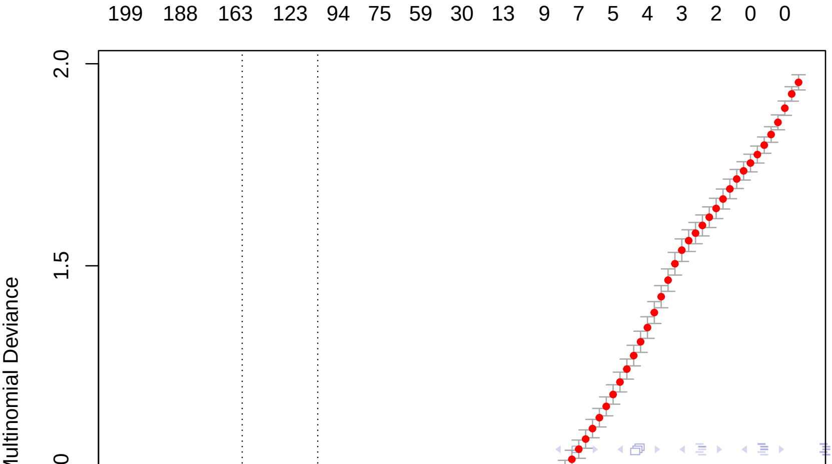
# An illustration

```
##just look at one word feature
library(glmnet)
x<-model.matrix(label1~., data=data.frame(label1=as.factor(TTIM$label1), DTM))[,-1]

lasso.mod=glmnet(x, as.factor(TTIM$label1),alpha=1,standardize=TRUE,family='multinomial')

plot(lasso.mod, xvar="lambda")
```



# An illustration

```
##just look at one word feature

cv.glmmod<-cv.glmnet(x,y=as.factor(TTIM$label1),alpha=1,standardize=TRUE,family="multinomial")


plot(cv.glmmod)
```


Multinomial Deviance