UCLAAnderson
SCHOOL of MANAGEMENT

MGMTMFE 431:

Data Analytics and Machine Learning

Topic 4: Model Selection and Shrinkage

Spring 2019

Professor Lars A. Lochstoer

UCLAAnderson
SCHOOL of MANAGEMENT

Model Selection and Shrinkage

a. Overfitting and the data-mining problem in finance
b. Shrinkage
c. Ridge Regression
d. Cross-Validation and Model Selection
e. Sparsity: The Lasso
f. Bayesian Regression
g. An automated “Machine Learning”-based dynamic trading strategy

UCLAAnderson
SCHOOL of MANAGEMENT

a. Overfitting: The data-mining problem in finance

A major concern in finance is that data-mining can lead to overfitting

- Example: Perfect model of expected return, f(X), has properties:

$$E_t\left[R^e_{i,t+1}\right] = f\left(X_{i,t}\right) \; for \; all \; i$$

and

$$e_{i,t+1} \equiv R^e_{i,t+1} - E_t\left[R^e_{i,t+1}\right] \; where \; E\left[f\left(X_{i,t}\right)e_{i,t+1}\right] = 0 \; for \; all \; i$$

Overfitting means that we are predicting residuals in sample and so in-sample we actually have $E\left[f\left(X_{i,t}\right)e_{i,t+1}\right] \neq 0$

- This in turn implies that $f\left(X_{i,t}\right) \neq E_t\left[R^e_{i,t+1}\right]$
- Thus, out-of-sample we have a quite imperfect predictor

UCLAAnderson
SCHOOL of MANAGEMENT

a. Overfitting: A simple example

We want to estimate expected return on N stocks using a sample of length T .

- Assume expected returns are constant for each stock

$$E_t\left[R^e_{i,t+1}\right] = E\left[R^e_{i,t+1}\right] = \mu_i \quad for\ all\ i$$

Assume the proposed model is $f\left(X_{i,t}\right) = \frac{1}{T}\sum_{t=1}^{T} R^e_{i,t}$

- Very flexible. No restrictions imposed across stocks.
- Also a *consistent* estimator (as *T* goes to infinity)
- Fits perfectly in-sample!
- But, clearly we are overfitting as $\frac{1}{T}\sum_{t=1}^{T} R^e_{i,t} = \mu_i + \frac{1}{T}\sum_{t=1}^{T} e_{i,t}$

UCLAAnderson
SCHOOL of MANAGEMENT

a. Overfitting: A simple example (cont'd)

Assume every stock has a return standard deviation of 40%

- Standard error of estimate is 8% with 25 years of data
- So, 95% confidence interval of estimate is +/- 16%. Pretty big…
  - Poor fit out-of-sample!

In this case, the high standard error told us estimate is very noisy

- At the same time, the estimate does contain some information
- Stocks with high expected return should on average have higher return and vice versa
- So, we shouldn't throw away this information, should we?

UCLAAnderson
SCHOOL of MANAGEMENT

b. Shrinkage

Some of the expected return estimates will be unreasonable

- Expected excess returns of, say, 32% or -16% seem too extreme
- Why?

Perhaps you have some underlying model of investor preferences and beliefs in mind, for instance the CAPM

- Long-run market betas are typically between 0 and 2 (95% confidence interval)
- So, expected excess returns should be between 0% and 16%, if market risk premium is 8%
- That implies a standard deviation in the cross-section of true expected returns of 4%.

How can we incorporate such 'prior' beliefs?

b. Shrinkage (cont'd)

Solution: 'shrink' your estimates towards the prior distribution

$$\mu_i^{shrunk} = w_i\mu_i + (1 - w_i)\mu_{prior}$$

where μ_{prior} is the unconditional mean of your prior distribution

- Intuitively, the weight should be a function of how informative your prior is relative to the standard error of your estimate
- For instance, if your estimate has zero standard error, it is truth and we should have $w_i = 1$.
- Conversely, if the estimate has infinite standard error it is useless and we should have $w_i = 0$.

Bayesian inference formalizes this logic and tells you what w_i should be

UCLAAnderson
SCHOOL of MANAGEMENT

b. Shrinkage and Bayesian Inference

Bayesian inference starts from the proposition that we can use probability statements to assess our views about a model parameter like a regression coefficient.

The idea is very appealing:

> Given our information (data and, e.g., economic theory), where do we think it "likely" that the parameters are?

To translate "likely" into something concrete, we say we would like to make probability statements about a parameter.

For example, suppose we are trying to make inferences regarding expected returns. We would like to say, e.g., "the probability is .95 that the expected (annual) excess return is between 0% and 15%."

UCLAAnderson
SCHOOL of MANAGEMENT

b. Bayesian Inference and Bayes Theorem

We need to start with some basic ideas from probability. Let's start with the joint distribution of two random variables, (x,y). This is represented by the joint or bivariate distribution. For continuous random variables, we use a joint or bivariate density function.

$$p_{X,Y}(x,y)$$

This is a surface over the (x,y) plane and provides the probability rate per unit of volume. The joint distribution of two random variables can always be written as (conditional times marginal).

$$p_{X,Y}(x,y) = p_{Y|X}(y \mid x)p_X(x)$$

b. Bayesian Inference and Bayes Theorem

Here the marginal distribution of X is obtained by integrating out Y from the joint or averaging all the conditional distributions:

$$p_X(x) = \int p_{X,Y}(x,y)dy$$

Let's consider a simple case. Suppose X,Y are bivariate normal with a simple covariance matrix

$$\begin{pmatrix} X \\ Y \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_X \\ \mu_Y \end{pmatrix}, \Sigma = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \right)$$

$$X \sim N(\mu_X, 1)$$

$$Y \mid X = x \sim N\left(\mu_Y + \rho(x - \mu_X), 1 - \rho^2\right)$$

UCLAAnderson
SCHOOL of MANAGEMENT

b. Shrinkage and Bayesian Inference

Bayes theorem tells us how to do this.

We start with a prior distribution over the model parameters

$$p(\theta)$$

And a model (i.e. joint distribution for our data given the theta).

$$p(\text{data} \mid \theta)$$

In our case, the parameters are the true mean excess return (μ_i) and stock return variance for each stock (σ_i^2). This density is the likelihood function of the stock returns, viewed as a function of these parameters.

UCLAAnderson
SCHOOL of MANAGEMENT

b. Shrinkage and Bayesian Inference

Example: Let's assume stock returns are *i.i.d.* normally distributed and consider a particular stock's return series with *T* observations (skip *i*'s). The likelihood of the "data," conditional on the parameters, is then:

$$\theta = \left(\mu, \sigma^2\right)$$

$$p(R_1^e, R_2^e, \ldots, R_T^e | \mu, \sigma^2) = \prod_{t=1}^{T} p(R_t^e | \mu, \sigma^2)$$

$$= (2\pi\sigma^2)^{-T/2} exp\left(-\frac{1}{2\sigma^2}\sum_{t=1}^{T}(R_t^e - \mu)^2\right)$$

b. Bayesian Inference and Bayes Theorem

Bayes theorem.

Let D denote our "data."

$$p(\theta|D) = \frac{p(D,\theta)}{p(D)} = \frac{p(D|\theta)p(\theta)}{p(D)}$$

$$p(\theta|D) \propto p(D|\theta)p(\theta)$$

$$\text{Posterior} \propto \text{"Likelihood"} \times \text{Prior}$$

The distribution, $p(\theta|D)$, is called the posterior distribution (posterior for "after"). Bayes Theorem tells us how to update our Prior views after we see a set of data!

b. The posterior

So, let's find our posterior

Let the prior distribution be $\mu \sim N(\bar{\mu}, \sigma^2_{prior})$ and assume σ^2 is known.

Let, σ^2_{Post}, denote the variance of the posterior distribution (work this out).

Then the posterior for μ is normal (since the likelihood and the prior are both normal; prove this yourself through a complete-the-squares argument):

$$p(\mu|R^e_1, R^e_2, \dots, R^e_T, \sigma^2)$$

$$\propto exp\left(-\frac{1}{2\sigma^2_{Post}}\left(\mu - \left(w\frac{1}{T}\sum_{t=1}^{T} R^e_t + (1-w)\bar{\mu}\right)\right)^2\right)$$

$$where\ \ w = \frac{T\sigma^{-2}}{T\sigma^{-2}+\sigma^{-2}_{Prior}}$$

UCLAAnderson
SCHOOL of MANAGEMENT

b. The Shrinkage Estimator

In sum, the best estimates of the means, given the data and your prior beliefs, are given by a weighted average of the sample mean and the prior mean where the weights depend on the dispersion in the prior and the standard error of the sample average:

$$\mu_i^{shrunk} = w_i \frac{1}{T}\sum_{t=1}^{T} R_{i,t}^{e} + (1 - w_i)\mu_{prior}$$

$$where \;\; w_i = \frac{T\sigma_i^{-2}}{T\sigma_i^{-2} + \sigma_{Prior}^{-2}}$$

Incorporating some notion of ***shrinkage*** or ***regularization*** typically will improve out-of-sample fit

- Also, provides an estimate even for stocks that have absolutely no return history (e.g., new listing)

b. The Shrinkage Estimator

In our example, $\sigma_i = 40\%$ and $\sigma_{Prior} = 4\%$.

Thus: $w_i = \frac{25 \times 0.4^{-2}}{25 \times 0.4^{-2} + 0.04^{-2}} = 0.2$.

With $\frac{1}{T}\sum_{t=1}^{T} R_{i,t}^{e} = -16\%$ we get

$$\mu_i^{shrunk} = 0.2 \times (-16\%) + 0.8 \times 8\% = 3.2\%$$

Very strong shrinkage due to the fact that even 25 years of past returns provides a very noisy signal of the mean return when return volatility is high (40% p.a.)

UCLAAnderson
SCHOOL of MANAGEMENT

b. Estimating the prior from data

Note that the prior can in fact be estimated from the cross-section itself:

$$\sigma_I^2(\hat{\mu}_i) = \sigma_I^2(\mu_i) + \frac{1}{N}\sum_{i=1}^{N}[St.error(\hat{\mu}_i)]^2$$

Thus, our prior estimate, obtained using the cross-section is

$$\sigma_I^2(\mu_i) = \sigma_I^2(\hat{\mu}_i) - \frac{1}{N}\sum_{i=1}^{N}[St.error(\hat{\mu}_i)]^2$$

Note, using the average standard error squared makes assumptions about the covariance matrix of estimation errors that may not be valid. This procedure often leads to reasonable results, but does not need to, so as always exercise good judgment.

This is a *data-driven* prior that often is very useful

- Note: does not control for prior data-mining!

UCLAAnderson
SCHOOL of MANAGEMENT

c. Ridge Regressions

Let's carry this idea over to regressions.

In particular, overfitting means regression coefficients (betas) are too big in absolute value.

Let's ***penalize*** the objective function when coefficients get too big, so as to shrink the coefficients towards zero

c. Recall: OLS objective function

Consider the OLS regression:

$$y_i = \beta_0 + \sum_{k=1}^{K} \beta_k X_{ki} + \epsilon_i$$

The objective function is

$$\min_{\beta} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{k=1}^{K} \beta_k X_{ki} \right)^2$$

Also, recall that the fit (R2) is not affected if the right hand side variables (X) are demeaned and standardized to have standard deviation of 1.

In addition, we for convenience assume the left hand side variable (y) will be demeaned. Thus, the intercept can be dropped as it will equal zero.

UCLAAnderson
SCHOOL of MANAGEMENT

c. Ridge regression objective function

The objective function of the ridge regression (using variables normalized as described on the previous slide) is

$$\min_{\beta} \frac{1}{2N} \sum_{i=1}^{N} \left(y_i - \sum_{k=1}^{K} \beta_k X_{ki} \right)^2 \quad s.t. \sum_{k=1}^{K} \beta_k^2 \leq B$$

Here, *B* is the "coefficient budget" the regression is given.

It is the nature of the constraint that makes the standardization of the signals (X) useful; each signal is given equal importance in the objective function.

One can un-demean all variables (y and X) by setting $\widehat{\beta_0} = \bar{y} - \sum_{k=1}^{K} \widehat{\beta_k} \overline{X_k}$

c. Ridge regression objective function

The objective function in ***penalty form***:

$$\min_{\beta} \frac{1}{2N} \sum_{i=1}^{N} \left(y_i - \sum_{k=1}^{K} \beta_k X_{ki} \right)^2 + \lambda \sum_{k=1}^{K} \beta_k^2$$

There exists a one-to-one mapping between *B* and $\lambda > 0$.

Clearly, for a small enough *B*, the coefficients will be ***regularized***, effectively shrunk towards zero. Note that the coefficients will then be biased.

The coefficients can be found by simply solving the Lagrangian problem, which is nicely behaved since the problem is convex, continuous and differentiable everywhere.

Let's consider some logistic Ridge regressions using the default model from the last topic notes

UCLAAnderson
SCHOOL of MANAGEMENT

c. The glmnet package

In this part of the course, we will use the glmnet package

- Generalized linear models, elastic-net regularization
- So, similar to the glm package, similar syntax, but with regularization

We use the Lending Club data you worked with in your homework

- First, assess model using only dummies based on "grade"

```
# read in and prepare the Lending Club Data
LC_RawData <- as.data.table(read.dta("LendingClub_LoanStats3a_v12.dta"))
LC_Baseline <- LC_RawData[(loan_status == "Fully Paid") | (loan_status == "Charged Off")]
LC_Baseline <- LC_Baseline[,Default:=(loan_status=="Charged Off")]

LC_Baseline <- LC_Baseline[,grade_factor:=as.factor(grade)]
factorToDummy(LC_Baseline,"grade_factor")

# create regressor matrix (glmnet needs matrix as x-input and vector as y-input)
regressors <-
as.matrix(LC_Baseline[,list(grade_factor_A,grade_factor_B,grade_factor_C,grade_factor_D,grade_factor
_E,grade_factor_F,grade_factor_G)])

# run the logistic ridge regression. Ridge is done when alpha = 0.
outloans_ridge = cv.glmnet(regressors,as.vector(LC_Baseline$Default), family=c("binomial"),alpha =
0, standardize = TRUE)
```

UCLAAnderson
SCHOOL of MANAGEMENT

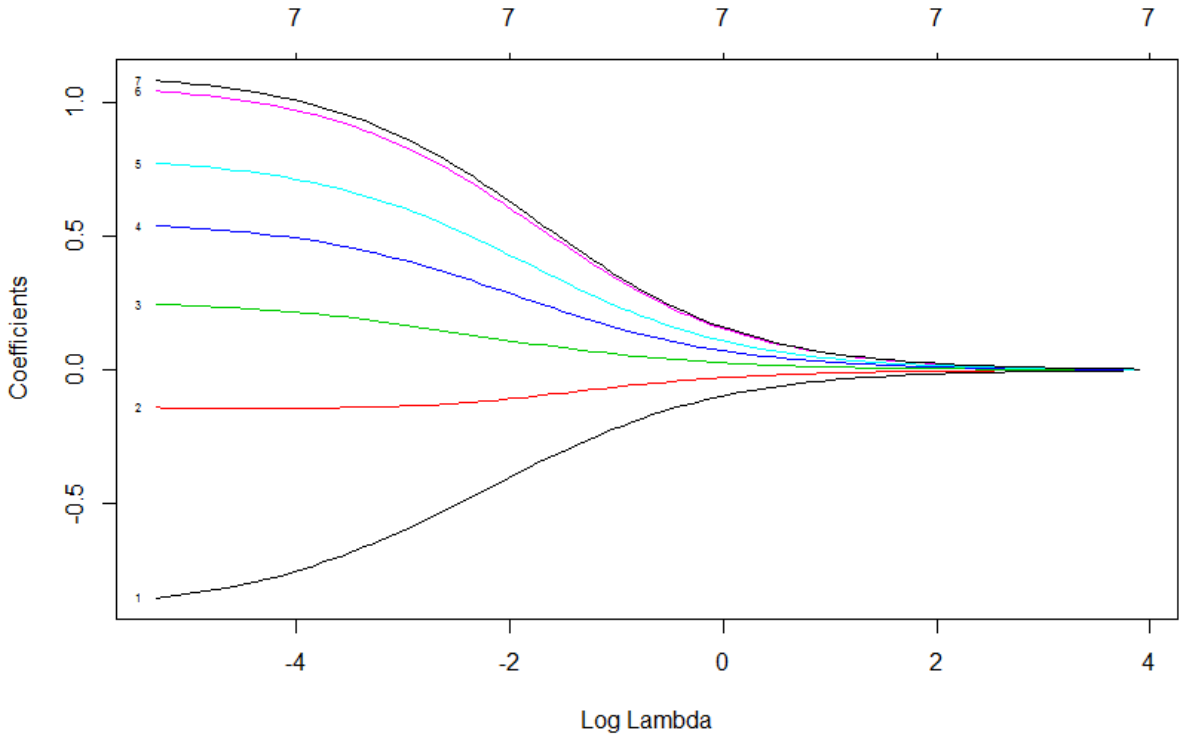
c. Ridge regression: Simple default model

The glmnet-function gives estimated coefficient for different values of "lambda" – a Lagrangian multiplier on the constraint

- A high value of *lambda* means the constraint is tight, a low value means it is not.
- Lambda = 0 means it does not bind at all and we have OLS coefficients
- The function plot.cv.glmnet plots the coefficients for different values of log(lambda)

```
> plot.glmnet(outloans_ridge$glmnet.fit,
"lambda", label = TRUE)
```

- Lines are numbered corresponding to coefficient place in regression (e.g., 7 equals "grade_factor_G")
- The 7 on top means that all 7 coefficients are non-zero



UCLAAnderson
SCHOOL of MANAGEMENT

c. Ridge regression: Which *B* (or *lambda*)?

Two questions are immediate:

1. How do you choose the level of the constraint *B* (or "*lambda*")?
2. How do you compute standard errors of the coefficients?

We will get back to 2. It is a little subtle as there is bias in the coefficients, so the standard error can severely understate the total uncertainty

Question 1. is handled through ***Cross-Validation***

- Cross-validation is an ***out-of-sample model selection criterion***
- Note that for each different B (lambda), we have a different model

UCLAAnderson
SCHOOL of MANAGEMENT

d. Cross-validation

Basic idea

- Split up the sample in random training and test sets and estimate performance on test sets.
- Choose the level of the constraint that gives best prediction

K-folds

- Split up the sample in K equal-sized groups (typically, 5 or 10; 10 is the default in cv.glmnet)
- For each of the K folds, estimate the model on the other K-1 folds and test on the K'th fold (so K 'out-of-sample' tests)
- The prediction error (typically, it's mean squared error, averaged across folds) is the basis for the choosing the best model

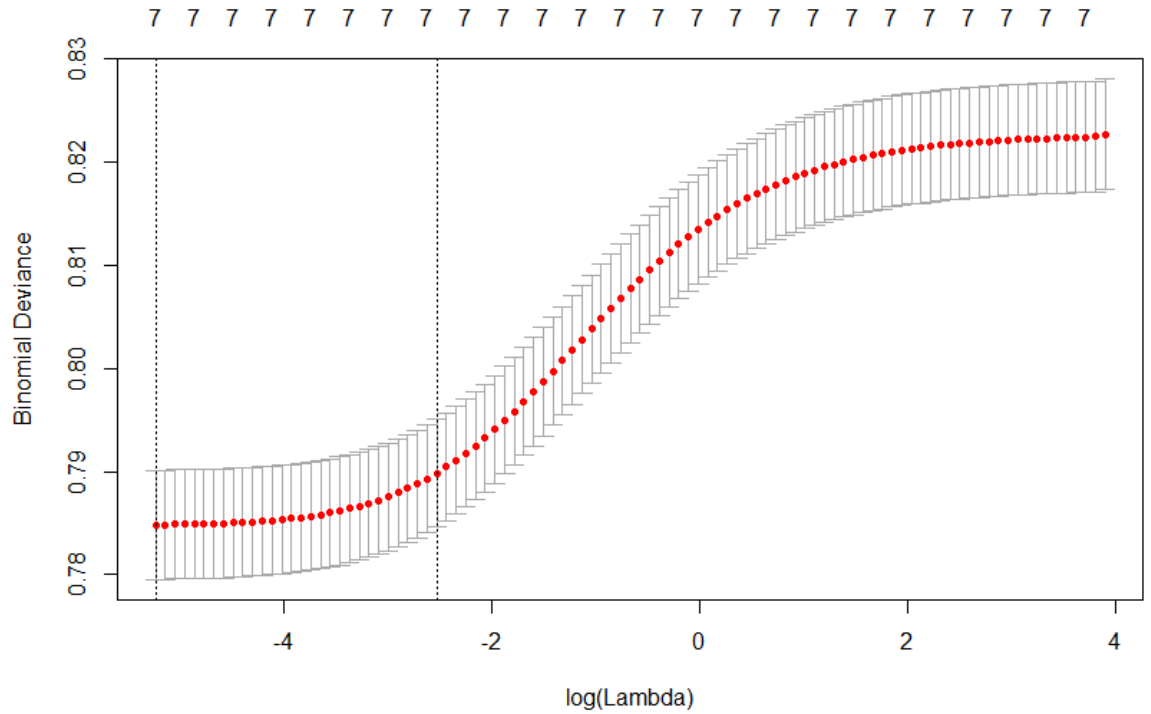
cv.glmnet has already done this!

- Use plot.cv.glmnet to see the outcome

UCLAAnderson
SCHOOL of MANAGEMENT

d. Cross-validation plot

```
> plot.cv.glmnet(outloans_ridge)
```



- The bounds are one standard error bounds based on the squared errors
- The leftmost line is the 'best' model with the least mean-squared-error (note that OLS corresponds to lambda = 0, so this is a somewhat restricted model)
- The more middle line is the 1-standard error away from least mean-squared error fit. This is often used as a more conservative fit. There is some overfitting also in the "out-of-sample" K-fold approach, which ultimately is an in-sample metric…

UCLAAnderson
SCHOOL of MANAGEMENT

d. An extended default model

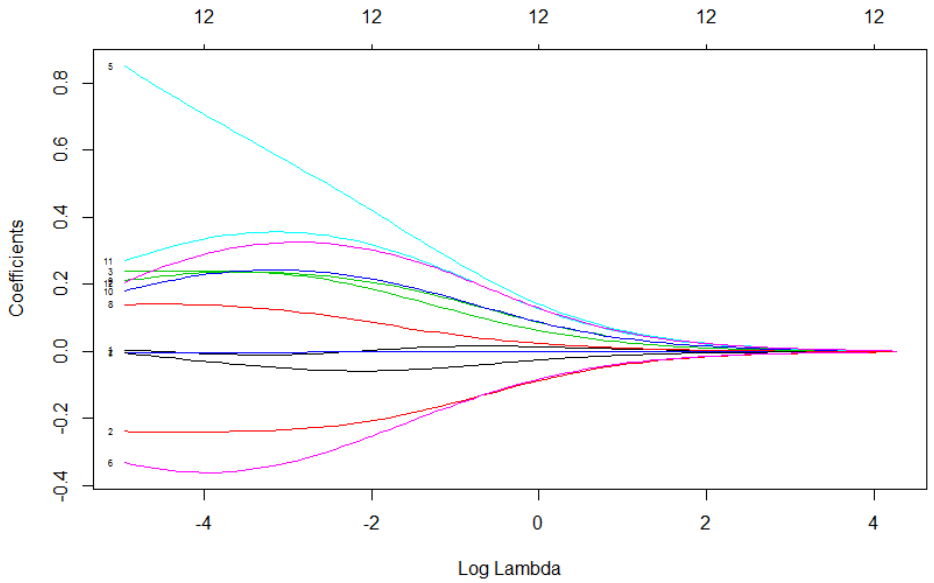
- In addition to Lending Club's "grade", we now add loan amount, annual income, term of loan (two dummies for up to 36 month, and then to 60 months), and the interest rate on the loan

```
regressors <- as.matrix(LC_Baseline[,list(loan_amnt/10000,`term_factor_ 36 months`, `term_factor_ 60 months`,
                      annual_inc/1000, 10*int_rate)])
regressors <- cbind(regressors, as.matrix(LC_Baseline[,list(grade_factor_A,grade_factor_B,grade_factor_C,
                      grade_factor_D,grade_factor_E,grade_factor_F,grade_factor_G)]))
outloans_ridge = cv.glmnet(regressors,as.vector(LC_Baseline$Default), family=c("binomial"),alpha = 0,
                      standardize = TRUE)
plot.glmnet(outloans_ridge$glmnet.fit, "lambda", label = TRUE)
```

Coefficient (5) is the interest rate

Note nonlinearities: interactions between regressors matter

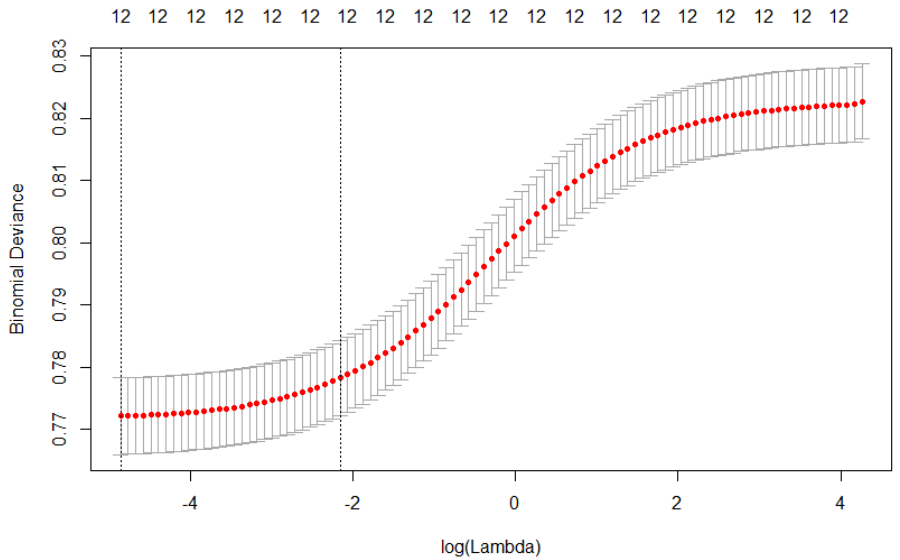
Note also that no coefficients are exactly zero



UCLAAnderson
SCHOOL of MANAGEMENT

d. An extended model: Prediction

```
> plot.cv.glmnet(outloans_ridge)
```



- Note deviance is lower for this model

```
➢ new_regs <- matrix(c(50000/10000,1,0,200000/10000,0.05*10,0,1,0,0,0,0,0),nrow = 1)
➢ predict(outloans_ridge, new_regs, s="lambda.1se", type = "response")
```

- Predicting the default probability for a person with the above attributes (loan of $50,000, <=36 month maturity, annual income of $200,000, etc.)
- Using “1 standard deviation above”-method (s = “lambda.1se” as opposed to s = “lambda.min”)
- Answer: `0.1028778`

e. The LASSO

LASSO: Least Absolute Selection and Shrinkage Operator

Same idea as Ridge regression, but penalty function is not sum of squared coefficients, but some of absolute values. Mathematically, the LASSO has an L_1 constraint, while Ridge has an L_2 constraint:

$$\min_{\beta} \frac{1}{2N} \sum_{i=1}^{N} \left(y_i - \sum_{k=1}^{K} \beta_k X_{ki} \right)^2 \quad s.t. \sum_{k=1}^{K} |\beta_k| \leq B$$

Here, *B* is the "coefficient budget" the regression is given.

It is the nature of the constraint that makes the standardization of the signals (X) useful; each signal is given equal importance in the objective function.

One can un-demean all variables (y and X) by setting $\widehat{\beta_0} = \bar{y} - \sum_{k=1}^{K} \widehat{\beta_k} \overline{X_k}$

UCLAAnderson
SCHOOL of MANAGEMENT

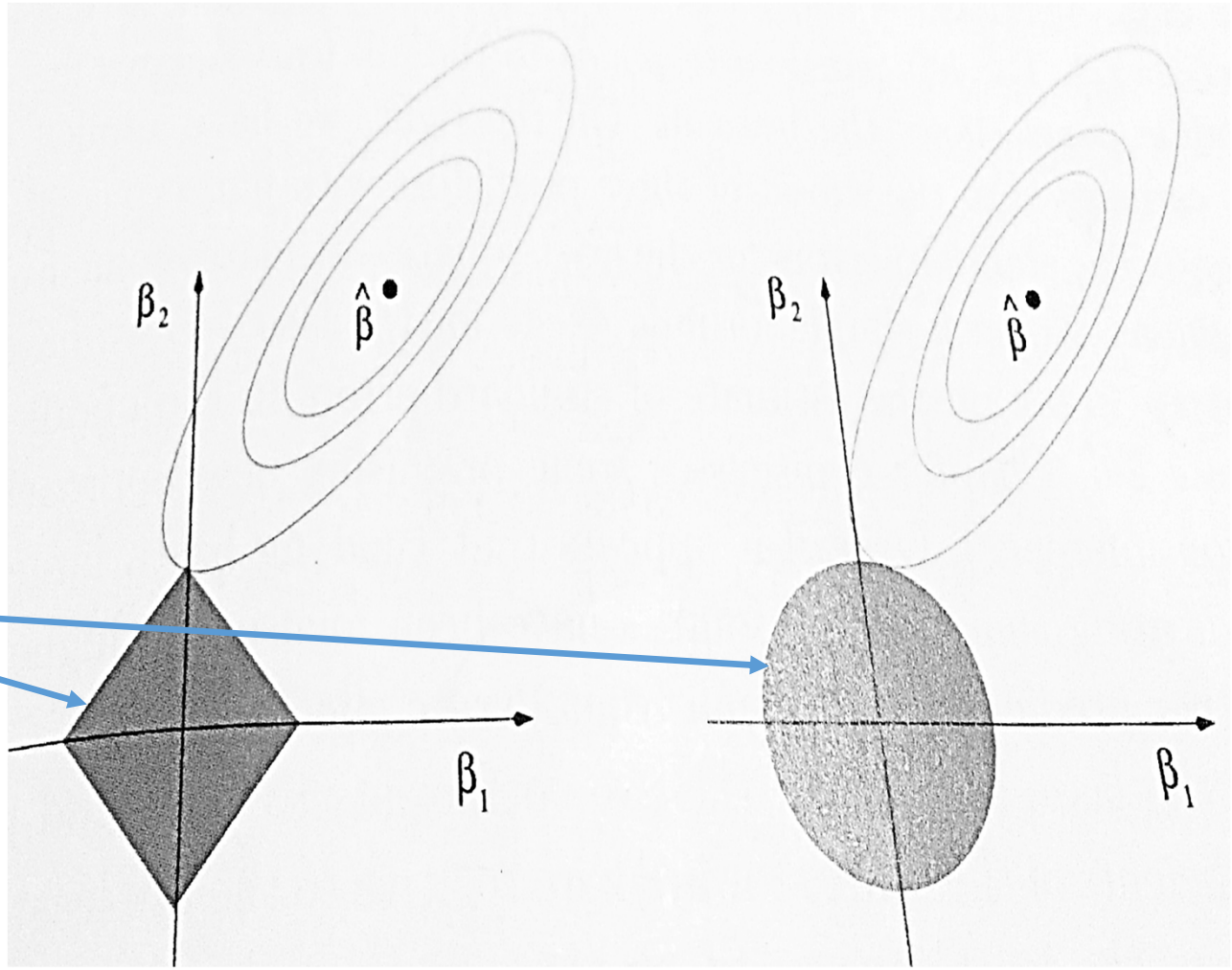
e. Lasso (L1) vs. Ridge (L2) penalty

Beta^hat is OLS estimate

Ellipses are contours for SSEs

Feasible set as given by constraint

Notice that LASSO is much more likely to land on a corner solution where one of the betas equal zero



UCLAAnderson
SCHOOL of MANAGEMENT

e. An extended default model

- Again, in addition to Lending Club's "grade", we now add loan amount, annual income, term of loan (two dummies for up to 36 month, and then to 60 months), and the interest rate on the loan
- We run the LASSO, by setting alpha = 1

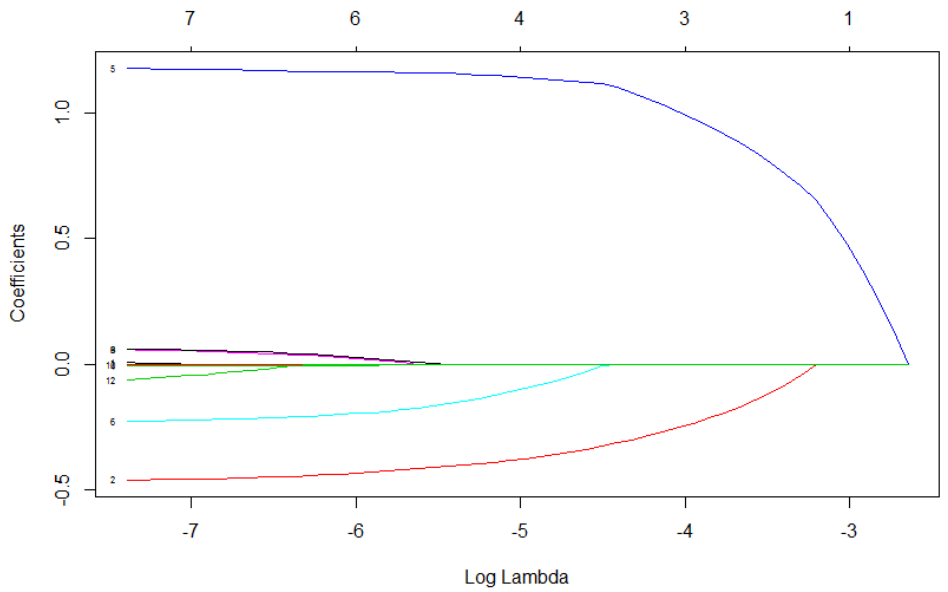
```
outloans_lasso = cv.glmnet(regressors,as.vector(LC_Baseline$Default), family=c("binomial"),alpha = 1,
                           standardize = TRUE)
plot.glmnet(outloans_ridge$glmnet.fit, "lambda", label = TRUE)
```

Coefficient (5) is the interest rate

Top axis gives number of non-zero coefficients

- Note corner-solutions, lots of zero betas

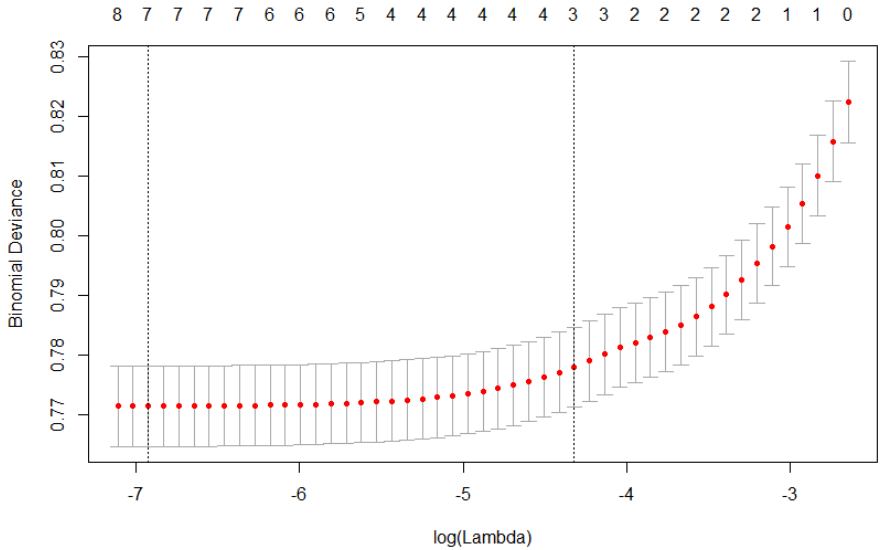
Interest rate (no. 5) is the most important, next is number 2, whether the loan is short-term or not, then 6, whether it is grade A or not, etc.



UCLAAnderson
SCHOOL of MANAGEMENT

e. An extended model: Prediction

```
> plot.cv.glmnet(outloans_lasso)
```



- Note that lambda.1se only has 3 predictors
  - A sparse model!
  - Bet-on-sparsity

```
➢ predict(outloans_lasso, new_regs, s="lambda.1se", type = "response")
```

- Predicting the default probability for a person with the above attributes (loan of $50,000, <=36 month maturity, annual income of $200,000, etc.)
- Using “1 standard deviation above”-method (s = “lambda.1se” as opposed to s = “lambda.min”)
- Answer: `0.06579284`, different from Ridge, more parsimonious model

UCLAAnderson
SCHOOL of MANAGEMENT

e. Elastic net

Finally, glmnet also allows alpha values between 0 and 1.

- This corresponds to the Elastic net estimator:

$$\min_{\beta} \sum_{i=1}^{N} \left(y_i - \sum_{k=1}^{K} \beta_k X_{ki} \right)^2 \quad s.t. \sum_{k=1}^{K} (1-\alpha)\beta_k^2 + \alpha|\beta_k| \leq B$$

The LASSO has problems with highly correlated X values. Erratic solution (switches between which coefficients are non-zero with large swings)

Adding a little of the Ridge penalty makes the problem strictly convex, which in cases with such high correlations between variables is a desirable property

One can un-demean all variables (y and X) by setting $\widehat{\beta_0} = \bar{y} - \sum_{k=1}^{K} \widehat{\beta_k}\overline{X_k}$

UCLAAnderson
SCHOOL of MANAGEMENT

f. Bayesian Inference and Bayes Theorem

Bayesian methods are becoming increasingly popular in a wide variety of industry contexts. The reason for their popularity is that Bayesian estimators impose a form of what statisticians call "regularization" or some call "shrinkage."

To obtain an intuition for this, consider the standard Least Squares estimator for a linear regression model.

$$b = \left(X'X\right)^{-1} X'y$$

Consider the case of one independent variable, suppose there is no variation in that variable (or very little). Then you simply can't run the regression (true also for logistic regression models).

$$X = \begin{bmatrix} \iota & x_1 \end{bmatrix}$$

UCLAAnderson
SCHOOL of MANAGEMENT

f. Bayesian Inference and Bayes Theorem

In the case of a very tiny amount of x variation, you will get a huge standard error and your estimates can be all over the place. If you are using the coefficients from these models to make allocation decisions or attribution inferences, then your model will produce absurd results and you will soon lose credibility.

A “regularization” procedure will take the least squares (or MLEs) and modify them so that they are more reasonable as well as have better sampling properties. Bayesian methods provide one type of regularization called shrinkage.

Barriers to using Bayesian methods: you have to understand more about distributions and densities and you have to use simulation as your method of computing!

UCLAAnderson
SCHOOL of MANAGEMENT

f. Bayesian References

Chapter 19 of Lander (very brief and with almost no explanation).

Bayesian Statistics and Marketing, Rossi, Allenby and McCulloch (detailed explanations along with accompanying R package, bayesm).

Bayesian Data Analysis, Gelman et al (good reference but weak on computing and priors).

Markov Chain Monte Carlo, Gammerman and Lopes (good more advanced treatment of Bayesian Simulation Methods).

UCLAAnderson
SCHOOL of MANAGEMENT

f. Bayes Regression

Now that we have the idea of a Bayesian method, let's consider regression from a Bayesian point of view.

What is the regression model likelihood function or distribution of the observed data?

$$y \mid X \sim N\left(X\beta, \sigma^2 I_n\right)$$

$$p\left(y_1,\ldots,y_n \mid \beta,\sigma^2\right) = \left(2\pi\sigma^2\right)^{-n/2} \exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i - x_i'\beta\right)^2\right]$$

Now, we will need a prior on both beta and sigma-squared. To obtain the intuition, we will ignore the prior on sigma-squared for now.

f. Bayes Regression – Prior on Beta

What prior distribution should we use for beta? A particularly convenient one is

$$\beta \sim N\left(\bar{\beta}, A^{-1}\right)$$

Obviously, beta-bar is the prior mean. A is called the prior "precision" matrix. Variance is the opposite of precision. That is, if the elements of A are very large then we have a lot of "precision" or "tightness" in the prior or a small prior variance.

How should we pick or "assess" the prior parameters? This is hard to do. One default is to use a relatively uninformative prior:

$$\beta \sim N\left(0, \left[.01 I_n\right]^{-1}\right)$$

UCLAAnderson
SCHOOL of MANAGEMENT

f. Bayes Regression – Prior on Beta

$$\beta \sim N\left(0, \left[.01 I_n\right]^{-1}\right)$$

Here the prior variance is 100 for each beta element or a prior standard deviation of 10. What we are saying is that we believe with probability .95 that each beta lies in the interval, [-20,20].

This is a very diffuse (variable) prior which appears to impart little information (note: all of the X variables are assumed to be on the same scale!).

Another suggestion for A:

- Estimate the regression betas on an earlier sample, obtain their standard errors, and use the these to form a *data-driven prior* as we did earlier when shrinking the mean return estimate

f. Bayes Regression – Bayes Estimator

The posterior corresponding to this prior and the normal regression likelihood has a non-standard form (it is not normal). But it is possible to write down a formula for the posterior mean.

$$\tilde{\beta} = \left(X'X + A\right)^{-1}\left(X'y + A\bar{\beta}\right)$$

Note that this estimator works even if there are linear dependencies in the X'X matrix (i.e. no variation in some of the X variables or perfect co-linearity). A more suggestive version is

$$\tilde{\beta} = \left(X'X + A\right)^{-1}\left(X'X\hat{\beta} + A\bar{\beta}\right)$$

The Bayes estimator is a matrix-weighted average of Least Squares and the prior mean. That means that it "shrinks" the Least Square estimator toward the prior mean.

UCLAAnderson
SCHOOL of MANAGEMENT

f. Bayes Regression, Ridge Regression

$$\tilde{\beta} = \left(X'X + A\right)^{-1}\left(X'X\hat{\beta} + A\bar{\beta}\right)$$

The amount shrinkage depends on the relative size of the X'X and A matrices.

If X'X is large (large N and/or a lot of variation in our data), there will be very little shrinkage. This applies variable by variable as well. If one of the variables has only a tiny bit of variation, then its coefficient will get shrunk a lot while others will not be shrunk much.

Even if the elements of A are very small, the Bayes estimator will "keep least squares out of trouble."

In fact, this is the Ridge regression, where the magnitude of prior uncertainty, A, scales with the Lagrangian "lambda"

UCLAAnderson
SCHOOL of MANAGEMENT

f. Bayes Regression, the LASSO

If we instead of a Normal prior, specify a Laplace distributed prior, we can view the LASSO estimator as a Bayesian regression:

$$\boldsymbol{y}|\beta,\lambda,\sigma\sim N(\boldsymbol{X}\beta,\sigma^2\boldsymbol{I}_{N\times N})$$

$$\beta|\lambda,\sigma\sim\prod_{j=1}^{p}\frac{\lambda}{2\sigma}e^{-\frac{\lambda}{\sigma}|\beta_j|}$$

The negative log posterior density (minus an unimportant constant term) is then of the same form as the LASSO objective function and so the solutions are the same (try to work this out yourself!)

UCLAAnderson
SCHOOL of MANAGEMENT

f. Bayes Regression, the LASSO

The Bayesian interpretation of the Ridge and the LASSO allows for computation/simulation of the ***posterior distribution*** of the regression coefficients, which allows for ***statistical inference*** and tests

This is a major benefit, as the LASSO and Ridge themselves, with just ad hoc constraints, cannot be used in conjunction with the standard OLS standard error apparatus (including White, Rogers, Hansen-Hodrick, and Newey-West standard errors).

- With the posterior distribution of a parameter, it is straightforward to ask, e.g., "What is the probability that this parameter is less than or equal to zero?"

UCLAAnderson
SCHOOL of MANAGEMENT

g. "Machine Learning"-based trading

In problem set 4, you will code up a fully automated low-frequency trading algorithm. Basically, we are coding up an algorithm for a long-short equity fund that trades low-frequency strategies

- The "low frequency"-part comes simply from the fact that we will be using the annual stock return and accounting characteristics dataset StockRetAcct_insample.dta (see Topic 1).
- A related paper by Nagel et al ("Shrinking the Cross-Section") is posted on the CCLE web-page
- The goal is to find a mean-variance efficient portfolio
- The math and setup for the problem set follows on the next slides

UCLAAnderson
SCHOOL of MANAGEMENT

g. "Machine Learning"-based trading

Since the MVE portfolio is traded, we can write:

$$R^e_{MVE,t+1} = \sum_{i=1}^{N_t} w_{i,t} R^e_{i,t+1}$$

Assume the portfolio weights, *w*, are a linear function of *K* stock characteristics, $X_{i,t}$--a Kx1 vector for each *i* and *t*:

$$w_{i,t} = b' X_{i,t}$$

Note that *b* is a *Kx1* constant vector (identifying assumption).

- However, we will let *b* be updated sequentially through the sample
- Assume (we can simply impose this) that all characteristics have cross-sectional mean equal to zero and cross-sectional variance equal to one

UCLAAnderson
SCHOOL of MANAGEMENT

g. "Machine Learning"-based trading

With these assumptions, we can write:

$$R^e_{MVE,t+1} = \sum_{i=1}^{N_t} b' X_{i,t} R^e_{i,t+1}$$
$$= b' \sum_{i=1}^{N_t} X_{i,t} R^e_{i,t+1} = b' F_{t+1}$$

Note that F_{t+1} is a Kx1 vector of factor returns, based on long-short portfolios formed using the K characteristics in $X_{i,t}$.

Our questions are:

1. What are the *K* characteristics in $X_{i,t}$?
2. What are the coefficients in *b*?

With this information, we can form the mean-variance portfolio based on our estimated portfolio weights and evaluate out of sample performance

UCLAAnderson
SCHOOL of MANAGEMENT

g. "Machine Learning"-based trading

From mean-variance math we have that the optimal combination of these *K* "mutual funds" with returns F_{t+1} (that makes the mean-variance efficient portfolio) is:

$$b = \Sigma^{-1} E[F_{t+1}]$$

Where Σ is the covariance matrix of the factor returns. Thus, we have the implication:

$$\Sigma b = E[F_{t+1}]$$

The sample counterpart is:

$$\overline{F_j} = \sum_{k=1}^{K} b_k cov_T\left(F_{k,t}, F_{j,t}\right) + \varepsilon_j$$

Where b_k is the *k*'th element of b, $\overline{F_j}$ is the sample mean of the *j*'th factor, and $cov_T\left(F_{k,t}, F_{j,t}\right)$ is the sample covariance between factor *k* and factor *j*

UCLAAnderson
SCHOOL of MANAGEMENT

g. "Machine Learning"-based trading

Repeating from previous slide:

$$\overline{F_j} = \sum_{k=1}^{K} b_k cov_T\left(F_{k,t}, F_{j,t}\right) + \varepsilon_j$$

This is a cross-sectional regression!

But, note that the solution is the in-sample mean-variance efficient combination of the factors. If there are many factors (as we will have), these are very badly measured.

Also, note that with *K* factors, we have *K* coefficients in *b*. Thus, the R2 in the in-sample cross-sectional regression will mechanically equal 1…

We need to add regularization to our procedure!! Let's use the elastic net, which includes Ridge and LASSO as special cases, penalizing the *b*-coefficients from being different from zero.

UCLAAnderson
SCHOOL of MANAGEMENT

g. “Machine Learning”-based trading

Problem set 4 asks you do run this procedure sequentially through the sample using a large set of characteristics.

In particular, you are asked to use all the characteristics in StockRetAcct_insample, their squares and all interaction terms.

This is a “machine learning”-exercise as you are asking the computer to sort out the functional form of the relation between characteristics and expected returns, as well as the MVE portfolios weights, in a fully automated fashion.

UCLAAnderson
SCHOOL of MANAGEMENT

Shrinkage Postscript

a. Ridge and Bayesian Regression Revisited

b. Standard errors vs. posterior distribution

c. Numerical posterior distributions

UCLAAnderson
SCHOOL of MANAGEMENT

a. Ridge regression objective function

Recall Ridge Regression:

$$\min_{\beta} \frac{1}{2} \sum_{i=1}^{N} (y_i - \beta x_i)^2 \quad s.t. \;\; \beta'\beta \leq B$$

Here, *B* is the scalar "coefficient budget" the regression is given. Let's express it in matrix form with an "L2" constraint:

$$\min_{\beta} \frac{1}{2} \{(Y - X\beta)'(Y - X\beta) + \lambda\beta'\beta\}$$

The *K* first-order conditions are:

$$-X'(Y - X\beta) + \lambda\beta = 0$$

Solving, we get:

$$\beta = (X'X + \lambda I_K)^{-1} X'Y$$

UCLAAnderson
SCHOOL of MANAGEMENT

a. Bayesian Regression

Standard OLS, where we assume the residual variance (σ^2) is known, says that estimated "betas" are normally distributed around the "true" value

$$\hat{\beta} \sim N(\beta_0, (X'X)^{-1}\sigma^2)$$

Assume you have a Normal prior for the true Kx1 beta-vector

$$\beta_0 \sim N\left(\bar{\beta}, \sigma^2 I_K / A\right)$$

Where A is a scalar and I_K is the $K \times K$ identity matrix.

Recall our shrinkage-math from the beginning of Topic 4, where we discussed how to get the posterior mean using both a prior about the mean and the sample mean estimate. (see next slide)

a. Shrinkage Math

Recall shrinkage-math from beginning of Topic 4

$$\mu_i^{shrunk} = w_i \frac{1}{T}\sum_{t=1}^{T} R_{i,t}^e + (1 - w_i)\mu_{prior}$$

$$where\ \ w_i = \frac{T\sigma_i^{-2}}{T\sigma_i^{-2} + \sigma_{Prior}^{-2}}$$

The multiple regression analogue is:

$$\beta^{PosteriorMean} = w\hat{\beta} + (I_K - w)\bar{\beta}$$

$$where\ \ w = \left((X'X)\sigma^{-2} + A\sigma^{-2}I_K\right)^{-1}(X'X)\sigma^{-2}$$
$$= (X'X + AI_K)^{-1}(X'X)$$

a. Bayesian Regression Posterior

The regression analogue is:

$$\beta^{PosteriorMean} = w\hat{\beta} + (I_K - w)\bar{\beta}$$

$$= (X'X + AI_K)^{-1}(X'X)\hat{\beta} + \left(I_K - (X'X + AI_K)^{-1}(X'X)\right)\bar{\beta}$$

$$= (X'X + AI_K)^{-1}\left(X'X\hat{\beta} + AI_K\bar{\beta}\right)$$

This can be written:

$$\beta^{PosteriorMean} = (X'X + AI_K)^{-1}\left(X'Y + AI_K\bar{\beta}\right)$$

a. Bayesian Regression Posterior

Since $X'X\hat{\beta} = X'Y$, we can write:

$$\beta^{PosteriorMean} = (X'X + AI_K)^{-1}\left(X'Y + AI_K\bar{\beta}\right)$$

With a prior centered around zero for all betas ($\bar{\beta} = 0$), we have:

$$\beta^{PosteriorMean} = (X'X + AI_K)^{-1}X'Y$$

which is the same as the Ridge Regression, with A = λ

Ok, but what about confidence bounds for $\beta^{Posterior}$?

b. Standard errors vs. posterior distribution

In standard, "frequentist" statistics, the data gives an estimate that is a random variable centered around truth (truth = null hypothesis). E.g.:

$$\hat{\beta} \sim N(\beta_0, (X'X)^{-1}\sigma^2)$$

In Bayesian statistics, the parameters themselves (not the estimate) are random variables with a distribution decoded in the prior, e.g.

$$\beta_0 \sim N\left(\bar{\beta}, \sigma^2 I_K / A\right)$$

and, after observing data, the posterior. In the case where we know the residual variance, σ^2, the posterior in our case is:

$$\beta^{Posterior} \sim N\left(\beta^{PosteriorMean}, (X'X + AI_K)^{-1}\sigma^2\right)$$

b. Standard errors vs. posterior distribution

The posterior in our case is:

$$\beta^{Posterior} \sim N\left(\beta^{PosteriorMean}, (X'X + AI_K)^{-1}\sigma^2\right)$$

Note that we have the variance of the posterior, which is Normal. Thus, it is now straightforward to make probabilistic statements, such as:

- "The 95% confidence bound for the true parameter is from 'a' to 'b' "

But, (a) where did this math come from, and (b) what do we do in the real-world case where we don't know σ^2?

- Regarding (a), all Bayesian linear regression math is standard and from the Topic 4 slides, you can derive everything yourself.
- Regarding (b), we will get there soon.
- That said, I also posted the Wikipedia-page(!) on it to CCLE Week 4.

UCLAAnderson
SCHOOL of MANAGEMENT

c. Numerical posterior distributions

Assume the prior and the likelihood functions both are known analytically.

- In our example, that is indeed the case. The data (regression residuals) are normally distributed given β and σ^2.
- We assumed σ^2 is known (for now) and that the prior on β is Normal.

Easy to get an approximate posterior distribution for β:

1. Choose a set of grid values for β (e.g., K = 1,000 equal spaced points between -5 and +5 standard deviations of prior). Denote the *k*'th grid value $\beta^{(k)}$.
2. Get the associated *K* values of the Normal pdf of the prior, p_k^{prior}
3. Get the associated *K* values of the Normal pdf for the data (values of the likelihood function for each beta-value on the grid), l_k^{data}
4. For each of the *K* beta-values on the grid, multiply these two numbers. Denote the *k*'th such value $p_k^{posterior} = l_k^{data} \times p_k^{prior}$
5. Define the numerical (discretized) pdf as

$$p^{posterior}\left(\beta = \beta^{(k)}\right) = p_k^{posterior} / \sum_{j=1}^{K} p_j^{posterior}$$

c. Numerical posterior distributions

If σ^2 is unknown, we need a prior over both β and σ^2.

It turns out a *hierarchical prior* is useful here. In particular, define prior over β conditional on σ^2 similar to before:

$$p(\beta|\sigma^2) = N\left(\bar{\beta}, A\sigma^2\right)$$

Next, assume the prior on σ^2 is *Inverse Gamma* distributed

$$p(\sigma^2) = IG(a, b)$$

Then, the joint prior distribution is:

$$p(\beta, \sigma^2) = p(\beta|\sigma^2)p(\sigma^2)$$

UCLAAnderson
SCHOOL of MANAGEMENT

c. Numerical posterior distributions

Note that, the marginal prior distribution of β, $p(\beta)$ is not known analytically, though it is easy to simulate. (we will do this in a second)

In this case, the posterior distribution is known to be Normal-Inverse-Gamma in the same conditional way as the prior (posterior distribution of beta conditional on σ^2 is Normal, posterior of σ^2 is Inverse Gamma)

- See Wikipedia-handout for background math

How do we get confidence bound on beta? Compute marginal distribution numerically (through simulation this time)!

1. Draw a value from the posterior over $\sigma^2, p(\sigma^2)$
2. Using this value for σ^2, draw a value of β from the posterior $p(\beta|\sigma^2)$
3. Repeat lots of times
4. Compute histogram for β based on simulated data
5. Use this histogram as discretized marginal probability density function for β