MS&E 226: "Small" Data

Lecture 7: Model selection (v3)

Ramesh Johari
ramesh.johari@stanford.edu

October 17, 2016

Model selection

Overview

Model selection refers to the process of comparing a variety of models (using, e.g., model complexity scores, cross validation, or validation set error.

In this lecture we describe a few strategies for model selection, then compare them in the context of a couple of real datasets.

Throughout, *our goal is prediction*. Therefore we compare models through estimates of their generalization error ("model scores"): e.g., training error (sum of squared residuals), R^2, C_p, AIC, BIC, cross validation, validation set error, etc.

Model selection: Goals

There are two types of qualitative goals in model selection:

- *Minimize prediction error*. This is our primary goal in this lecture.
- *Interpretability*. We will have more to say about this in the next unit of the class.

Both goals often lead to a desire for "parsimony": roughly, a desire for smaller models over more complex models.

Subset selection

Suppose we have p covariates available, and we want to find which p to include in a linear regression fit by OLS.

One approach is:

- For each subset $S \subset \{1, \ldots, p\}$, compute the OLS solution with just the subset of covariates in S.
- Select the subset that minimizes the chosen model score.

Implemented in R via the `leaps` package (with C_p or R^2 as model score).

Problem: Computational complexity scales exponentially with number of covariates.

Forward stepwise selection

Another approach:

1. Start with $S = \emptyset$.
2. Add the single covariate to S that leads to greatest reduction in model score.
3. Repeat steps 1-2.

Implemented in R via the `step` function (with AIC or related model scores).

The computational complexity of this is only quadratic in the number of covariates (and often much less).

Backward stepwise selection

Another approach:

1. Start with $S = \{1, \ldots, p\}$.
2. Delete the single covariate from S that leads to greatest reduction in model score.
3. Repeat steps 1-2.

Also implemented via `step` in R.

Also quadratic computational complexity, though it can be worse than forward stepwise selection when there are many covariates. (In fact, backward stepwise selection can't be used when $n \leq p$ — why?)

Stepwise selection: A warning

When applying stepwise regression, you are vulnerable to the same issues discussed earlier:

- The same data is being used repeatedly to make selection decisions.
- In general, this will lead to downward biased estimates of your prediction error.

The train-validate-test methodology can mitigate this somewhat, by providing an objective comparison.

To reiterate: Practitioners often fail to properly isolate test data during the model building phase!

Regularization

Lasso minimizes:

$$\text{SSE} + \lambda \sum_{j=1}^{p} |\hat{\beta}_j|$$

where $\lambda > 0$.

Ridge regression minimizes:

$$\text{SSE} + \lambda \sum_{j=1}^{p} |\hat{\beta}_j|^2.$$

where $\lambda > 0$.

Regularization

Both lasso and ridge regression are "shrinkage" methods for model selection:

- Relative to OLS, both lasso and ridge regression will yield coefficients $\hat{\beta}$ that have "shrunk" towards zero.
- The most explanatory covariates are the ones that will be retained.
- Lasso typically yields a much smaller subset of nonzero coefficients than ridge regression or OLS (i.e., fewer nonzero entries in $\hat{\beta}$).

To use these for model selection, tune λ to minimize the desired model score.

Intuition for lasso

Why does lasso tend to "truncate" more coefficients at zero than ridge?

Intuition for lasso

Example: Crime dataset

Crime dataset

From:
`http://lib.stat.cmu.edu/DASL/Datafiles/USCrime.html`

Contains data on crime rates in 47 US states in 1960.
Synthesized from government statistics.

Crime dataset

Variable name	Description
R	Crime rate: # of offenses reported to police per million population
Age	The number of males of age 14-24 per 1000 population
S	Indicator variable for Southern states (0 = No, 1 = Yes)
Ed	Mean # of years of schooling x 10 for persons of age 25 or older
Ex0	1960 per capita expenditure on police by state and local government
Ex1	1959 per capita expenditure on police by state and local government
LF	Labor force participation rate per 1000 civilian urban males age 14-24

Crime dataset

Variable name	Description
M	The number of males per 1000 females
N	State population size in hundred thousands
NW	The number of non-whites per 1000 population
U1	Unemployment rate of urban males per 1000 of age 14-24
U2	Unemployment rate of urban males per 1000 of age 35-39
W	Median value of transferable goods and assets or family income in tens of $
X	The number of families per 1000 earning below 1/2 the median income

Forward stepwise regression

```
> fm.lower = lm(data = crime.df, R ~ 1)
> fm.upper = lm(data = crime.df, R ~ .)
> step(fm.lower,
     scope = list(lower = fm.lower,
        upper = fm.upper),
     direction = "forward")
```

Forward stepwise regression: Step 1

```
Start:  AIC=344.58
R ~ 1

       Df Sum of Sq   RSS    AIC
+ Ex0   1     32533 36276 316.49
+ Ex1   1     30586 38223 318.95
+ W     1     13402 55408 336.40
+ N     1      7837 60973 340.90
+ Ed    1      7171 61638 341.41
+ M     1      3149 65661 344.38
<none>              68809 344.58
+ LF    1      2454 66355 344.87
+ X     1      2205 66604 345.05
+ U2    1      2164 66646 345.08
+ S     1       565 68244 346.19
+ Age   1       551 68258 346.20
+ U1    1       175 68634 346.46
+ NW    1        73 68736 346.53
```

Forward stepwise regression: Step 2

```
Step:  AIC=316.49
R ~ Ex0

       Df Sum of Sq   RSS    AIC
+ X     1    7398.2 28878 307.77
+ Age   1    6167.4 30109 309.73
+ M     1    2505.2 33771 315.13
+ NW    1    2324.3 33952 315.38
+ S     1    2191.0 34085 315.56
+ W     1    1808.7 34468 316.09
<none>               36276 316.49
+ Ex1   1    1461.7 34815 316.56
+ LF    1     774.8 35501 317.48
+ U2    1     178.5 36098 318.26
+ N     1      56.7 36220 318.42
+ U1    1      28.8 36247 318.45
+ Ed    1       7.7 36269 318.48
```

Forward stepwise regression: Final output

```
Call:
lm(formula = R ~ Ex0 + X +
    Ed + Age + U2 + W, data = crime.df)

Coefficients:
(Intercept)      Ex0        X       Ed      Age       U2
  -618.5028   1.0507   0.8236   1.8179   1.1252   0.8282
          W
     0.1596
```

Backward stepwise regression yields the same result. Is this an interpretable model?

Example: Baseball hitters

Baseball hitters

Data taken from *An Introduction to Statistical Learning*.

Consists of statistics and salaries for 263 Major League Baseball players.

We use this dataset to:

- Develop the train-test method
- Apply lasso and ridge regression
- Compare and interpret the results

We'll use the `glmnet` package for this example.

Loading the data

glmnet uses matrices rather than data frames for model building:

```
> library(ISLR)
> library(glmnet)

> data(Hitters)
> hitters.df = subset(na.omit(Hitters))

> X = model.matrix(Salary ~ 0 ., hitters.df)
> Y = hitters.df$Salary
```

Training vs. test set

Here is a simple way to construct training and test sets from the single dataset:

```
train.ind = sample(nrow(X), round(nrow(X)/2))
X.train = X[train.ind,]
X.test = X[-train.ind,]
Y.train = Y[train.ind]
Y.test = Y[-train.ind]
```

Ridge and lasso

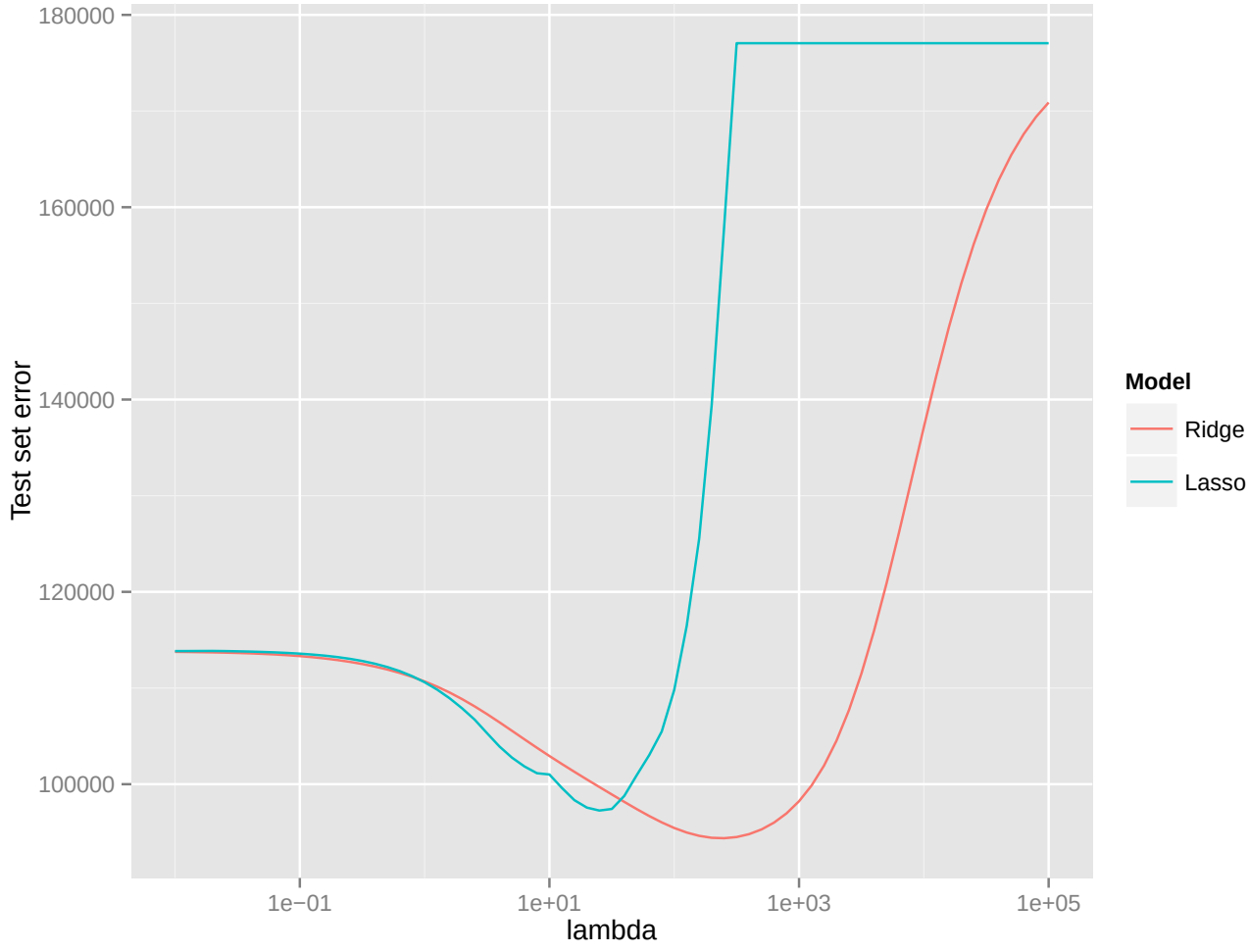
Building a lasso model:

```
> lambdas = 10^seq(-2,3.4,0.1)
> fm.lasso = glmnet(X.train,
      Y.train, alpha = 1,
      lambda = lambdas, thresh = 1e-12)
```

Setting `alpha = 0` gives ridge regression.
Make predictions as follows at $\lambda =$ `lam`:

```
> mean( (Y.test -
    predict(fm.lasso, s = lam, newx = X.test))^2 )
```

Results



What is happening to lasso?

Lasso coefficients

Using `plot(fm.lasso, xvar="lambda")`: