Fundamentals of Data Science

Model selection using model scores

Ramesh Johari

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.

Here we describe a few strategies for model selection using model scores, then compare them in the context of a real dataset.

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 subset 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!

Example: Fuel economy dataset

Fuel economy dataset

Data on fuel economy of 2016 vehicles.

From: `https://www.fueleconomy.gov/feg/download.shtml`
(via DASL from Data Description, Inc.)
Contains data on fuel economy of 1211 U.S. vehicles in 2016.

Forward stepwise regression

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

Forward stepwise regression: Step 1

```
Start:  AIC=4158.86
CombinedMPG ~ 1

               Df Sum of Sq   RSS    AIC
+ CityMPG       1     36336  1152  -56.3
+ HighwayMPG    1     34459  3029 1114.3
+ CityCO2       1     33259  4229 1518.4
+ CombCO2       1     33248  4240 1521.7
+ Car.line    771     35808  1680 1940.5
+ HwyCO2        1     30335  7153 2154.9
+ Displacement  1     22487 15001 3051.7
+ Cylinders     1     20754 16735 3184.1
+ Transmission 23     12267 25222 3724.9
+ Division     45     11870 25618 3787.8
+ Class        10      8758 28731 3856.7
+ Mfr          24      7970 29518 3917.4
+ Gears         1      5195 32294 3980.2
<none>                      37488 4158.9
+ Sample       1        40 37448 4159.6
```

Forward stepwise regression: Step 2

```
Step:  AIC=-56.34
CombinedMPG ~ CityMPG

                Df Sum of Sq     RSS      AIC
+ HighwayMPG     1    983.00  169.13 -2377.90
+ Car.line     771   1049.33  102.81 -1440.75
+ HwyCO2         1    724.64  427.50 -1254.96
+ CombCO2        1    535.50  616.63  -811.34
+ CityCO2        1    315.62  836.51  -442.02
+ Class         10    211.45  940.68  -281.89
+ Division      45    240.43  911.71  -249.78
+ Transmission  23    206.21  945.93  -249.16
+ Displacement   1    165.09  987.04  -241.63
+ Mfr           24    160.98  991.16  -190.60
+ Cylinders      1    117.02 1035.11  -184.05
+ Gears          1     65.20 1086.93  -124.89
<none>                       1152.13   -56.34
+ Sample         1      0.01 1152.12   -54.36
```

Forward stepwise regression: Step 3

```
Step:  AIC=-2377.9
CombinedMPG ~ CityMPG + HighwayMPG

               Df Sum of Sq     RSS     AIC
+ CityCO2       1    12.083 157.047 -2465.7
+ CombCO2       1     9.524 159.605 -2446.1
+ Displacement  1     8.221 160.908 -2436.2
+ Cylinders     1     5.697 163.433 -2417.4
+ Class        10     5.935 163.194 -2401.2
+ HwyCO2        1     1.320 167.809 -2385.4
+ Division     45    12.223 156.906 -2378.7
<none>                      169.129 -2377.9
+ Sample        1     0.065 169.064 -2376.4
+ Gears         1     0.032 169.097 -2376.1
+ Transmission 23     5.403 163.727 -2371.2
+ Mfr          24     5.508 163.622 -2370.0
+ Car.line    771   120.153  48.976 -2336.7
```

Forward stepwise regression: Final output

```
Call:
lm(formula = CombinedMPG ~ CityMPG + HighwayMPG + CityCO2 +
    HwyCO2 + CombCO2 + Cylinders + Displacement,
    data = fuel_economy.df)

Coefficients:
 (Intercept)       CityMPG    HighwayMPG       CityCO2
     1.10126       0.59694       0.37391       0.01265
HwyCO2       CombCO2     Cylinders
0.01914      -0.03182       0.04376
Displacement
    -0.05049
```

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