ORIE 3120: Practical Tools for OR, DS, and ML

Model Selection and Logistic Regression

Professor Udell

Operations Research and Information Engineering
Cornell

April 20, 2020

Outline

Model selection

which features should appear in your model? two regimes

small data: (this class)

- use domain knowledge to decide features
- drop features with very small p values

big data: (ORIE 4741)

- use cross-validation to select best model
- use held-out test set to assess model performance

Model selection and p values

- if you fit **very few** models, and assumptions hold, then p values are reliable
- p values are **not** reliable if you fit many models or select from many features

Model selection and p values

- if you fit **very few** models, and assumptions hold, then p values are reliable
- p values are **not** reliable if you fit many models or select from many features

solution: use a held-out test set

- split dataset into training data and testing data before you begin
- use training dataset to select model
- use test dataset to assess quality of fit

Model selection demo

Demo:
`https://github.com/madeleineudell/orie3120-sp2020/blob/master/demos/model-selection.ipynb`

demo shows three methods for model selection:

- dropping big p-values up to threshold
- dropping big p-values to minimize AIC
- using the Lasso to select features

there are many more!

Aikake Information Criterion (AIC)

Continuous data:

$$\begin{aligned} \text{AIC} &= \text{RSS} + 2p \\ &= \sum_{i=1}^{n} \hat{\epsilon}_i^2 + 2p \end{aligned}$$

- decreases as model fit improves
- increases with more covariates p
- models with small AIC predict better
- AIC can also be defined for other models (*e.g.*, for binary data)

AIC example: electricity usage

Example: Electricity usage

Model	AIC
Linear	427.3
Quadratic	409.5
Cubic	411.4
Quartic	413.4

- a difference of 1 or 2 in AIC values is not important
- if several models have nearly the same AIC values, then generally one uses the simplest

Outline

Stepwise variable selection

start with some model

- the model is modified in steps
- in each step a variable is either added or dropped
- select the move that decreases AIC the most
- the algorithm stops when no move decreases AIC

Outline

Part 2: Logistic Regression For Binary outcomes

- Often the response is binary, e.g.,
  - “no” or “yes”
  - “defective” or “good”
  - “dead” or ”alive”
  - often coded “0” or “1”
- Alternatively, the response is the number of “yes” responses in a number of “trials”
- Binary regression:
  - model the conditional probability of “yes” given the predictors

Logistic Regression is Useful

Binary regression: data

For the ith case:

- $X_{i,1}, \ldots, X_{i,p}$ are the predictors
- n_i is the number of "trials"
- $p(X_{i,1}, \ldots, X_{i,p})$ is the conditional probability of a "yes" or, equivalently, that $Y_i = 1$
- $Y_i | X_{i,1}, \ldots, X_{i,p}$ is Binomial$\{p(X_{i,1}, \ldots, X_{i,p}), n_i\}$
- So

$$Pr(Y_i = y | X_{i,1}, \ldots, X_{i,p})$$

$$= \binom{n_i}{y} p(X_{i,1}, \ldots, X_{i,p})^y \{1 - p(X_{i,1}, \ldots, X_{i,p})\}^{n_i - y}$$

for $y = 0, \ldots, n_i$

Modeling $p(X_1, \ldots, X_p)$: first attempt

From previous slide:

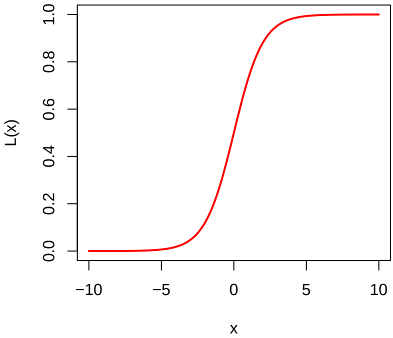
$p(X_1, \ldots, X_p)$ is the conditional probability of a "yes"

Linear model:

$$p(X_1, \ldots, X_p) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

What is wrong with this model?

Logistic function



$$L(x) = \frac{1}{1 + \exp(-x)}$$

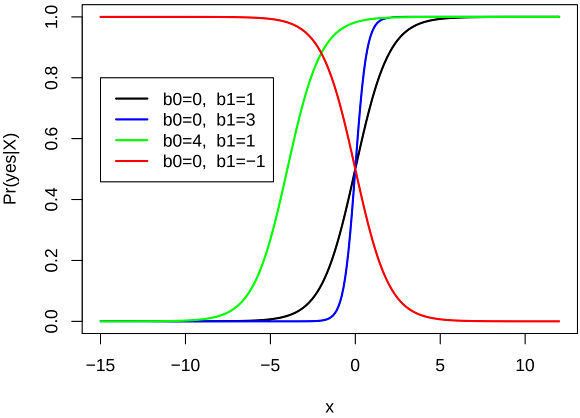
Logistic regression model

$$p(X_1, \ldots, X_n) = L(\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p)$$

Let's look at the simplest case, $p = 1$:

$$p(X) = L(\beta_0 + \beta_1 X)$$

Some logistic models with one X



Logit function

$$p(X_1, \ldots, X_n) = L(\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p)$$

implies that

$$L^{-1}\{p(X_1, \ldots, X_n)\} = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

L^{-1} is called the "logit" function and is

$$L^{-1}(p) = \log\left(\frac{p}{1-p}\right)$$

Also called "log-odds"

Link function

The “odds” for “yes” against “no” is

$$\frac{p}{1-p}$$

So the logistic model says that the log-odds equals

$\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$

The logit function is called the “link” function because it links

- $p(X_1, \ldots, X_n)$, and
- $\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$

Maximum Likelihood Estimation

Let y_i be the value of Y_i actually observed. Then the likelihood function evaluated at $\beta_0, \beta_1, \ldots, \beta_p$ is

$$\text{Likelihood}(\beta_0, \beta_1, \ldots, \beta_p) := Pr(Y_1 = y_1, \ldots, Y_n = y_n) =$$

$$= \prod_{i=1}^{n} \binom{n_i}{y_i} p(X_{i,1}, \ldots, X_{i,p})^{y_i} \{1 - p(X_{i,1}, \ldots, X_{i,p})\}^{n_i - y_i}$$

Maximum likelihood estimation

- The maximum likelihood estimates are the values of $\beta_0, \beta_1, \ldots, \beta_p$ that make $\mathrm{Likelihood}(\beta_0, \beta_1, \ldots, \beta_p)$ as large as possible.
- The MLE's are computed by an iterative algorithm.
- Fisher scoring (aka Newton's method) is one of the popular algorithms
- If you want details on computing the MLE, take Learning with Big Messy Data!

Maximum likelihood is a general estimation method

- As we have seen, MLE is used for logistic regression
  - but MLE is not a special-purpose tool used just for logistic regression
- MLE = least squares for linear regression with normally distributed noise
- MLE is used for a wide variety of other statistical models
- MLE is, by far, the most popular estimation method

Logistic regression demo

Demo:
`https://github.com/madeleineudell/orie3120-sp2020/blob/master/demos/logistic-regression.ipynb`

GLMs

Logistic regression is an example of a generalized linear model (GLM)

A GLM is similar to a LM, except

- the linear prediction equation

$$E(Y|X_1, \ldots, X_p) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

  is replaced by

$$E(Y|X_1, \ldots, X_p) = H(\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p)$$

  for a suitable function H

- $H =$ logistic function for logistic regression

GLMs, cont.

- The conditional normal distribution of Y given $X_1, \ldots, X_p$ is replaced by another family of distributions
  - binomial distributions for logistic regression
- Poisson regression is another example of a GLM
  - Y_i is Poisson
  - $H(x) = \exp(x)$ because the mean of a Poisson is positive

GLMs

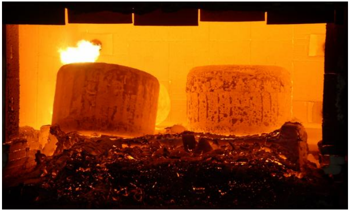
$$E(Y|X_1, \ldots, X_p) = H(\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p)$$

implies that

$$\frac{\partial}{\partial X_j} E(Y|X_1, \ldots, X_p) = H'(\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p)\beta_j$$

so the coefficients in a GLM can be interpreted in roughly the same way in a LM

Example: Heating steel ingots to be rolled is hard!



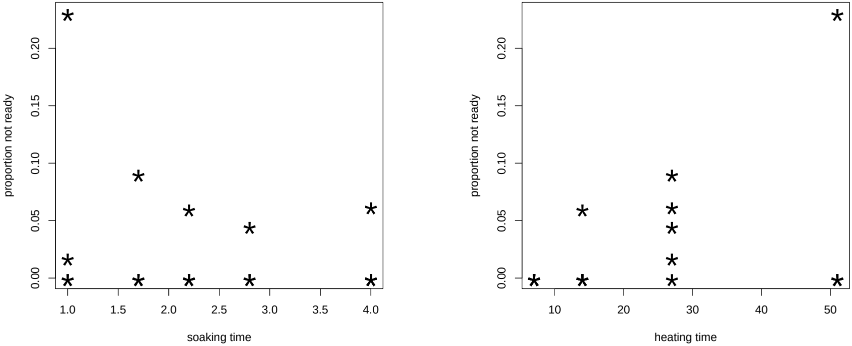
Example: ingots data

Soak Time	Heat Time	Not Ready	n_i
1	7	0	10
1	14	0	31
1	27	1	56
1	51	3	13
1.7	7	0	17
1.7	14	0	43
1.7	27	4	44
1.7	51	0	1
2.2	7	0	7
2.2	14	2	33
2.2	27	0	21
2.2	51	0	1
2.8	7	0	12
2.8	14	0	31
2.8	27	1	22
2.8	51	0	0
4	7	0	9
4	14	0	19
4	27	1	16
4	51	0	1

n_i = number of ingots prepared

proportion not ready = (Not Ready) $/n_i$

Let's look at the data



Outline

Need analog of sum of squares

- In linear regression, we found $\hat{\beta}_0, \ldots, \hat{\beta}_p$ by minimizing the sum of squared residuals,

$$\text{Sum of Squared Residuals} = \sum_{i=1}^{n} \Big\{ Y_i - (\beta_0 + \beta_1 X_{i,1} + \ldots + \beta_p X_{i,p}) \Big\}^2$$

$$= \text{positive constant} \times (-2 \times \text{log-likelihood}) + \text{another constant}$$

- The same $\hat{\beta}_0, \ldots, \hat{\beta}_p$ minimize

$$-2\times \text{ log-likelihood}$$

- We define the Deviance to be

$$\text{Deviance} = -2\times \text{ log-likelihood}$$

Deviance is the analog of sum of squares

Logistic regression:

Notation: $\hat{p}_i = L(\beta_0 + \beta_1 X_{i,1} + \ldots + \beta_p X_{i,p})$

For simplicity: Assume the binary, not binomial, case

The MLE maximizes

$$= \prod_{i=1}^{n} \hat{p}_i^{y_i} (1 - \hat{p}_i)^{1-y_i}$$

and minimizes

$$\text{Deviance} := -2 \sum_{i=1}^{n} \Big[y_i \log(\hat{p}_i) + (1 - y_i) \log(1 - \hat{p}_i) \Big]$$

Deviance residuals

$$\text{Deviance} := -2\sum_{i=1}^{n}\underbrace{\Big[y_i\log(\hat{p}_i) + (1-y_i)\log(1-\hat{p}_i)\Big]}_{\leq 0}$$

$$= \sum_{i=1}^{n}\Big\{(\text{Deviance residual})_i\Big\}^2$$

where

$$(\text{Deviance residual})_i = \pm\sqrt{-2\Big\{y_i\log(\hat{p}_i) + (1-y_i)\log(1-\hat{p}_i)\Big\}}$$

- $\pm$ is determined so that the deviance residual has the same sign as $\Big\{y_i - \hat{p}_i\Big\}$

Deviance residuals: when are they small?

$$\text{Deviance} = \sum_{i=1}^{n} \left\{ (\text{Deviance residual})_i \right\}^2$$

$$(\text{Deviance residual})_i = \pm\sqrt{-2\left\{ y_i \log(\hat{p}_i) + (1 - y_i) \log(1 - \hat{p}_i) \right\}}$$

$(\text{Deviance residual})_i = 0$ if and only if

- $y_i = 1$ and $\hat{p}_i = 1$

or

- $y_i = 0$ and $\hat{p}_i = 0$

Deviance and AIC

Binary data:

$\text{AIC} = \text{Deviance} + 2 \times (\# \text{ parameters})$

Binomial data:

$\text{AIC} = \text{Deviance} + 2 \times (\# \text{ parameters}) + \text{constant}$

The constant comes from the logs of the binomial coefficients

AIC for model comparison

$$\begin{aligned}\text{AIC} &= -2\log\Big(\text{maximized likelihood}\Big) \\ &+ 2\Big(\text{number of parameters}\Big) \\ &= \underbrace{\text{Deviance}}_{\text{poor fit penalty}} + \underbrace{2\Big(\text{number of parameters}\Big)}_{\text{complexity penalty}}\end{aligned}$$

- AIC can be used with any GLM
  - including any LM
- Smaller is better: Models with small AIC predict better