Multiple regression

Hao Nguyen

HMU

Seminar optima
Ngày 30 tháng 5 năm 2019

Table of Contents

1. Introduction
2. The model
3. Further Assumptions of the Least Squares Model
4. Fitting the Model
5. Goodness of Fit
6. The bootstrap
7. Regularization

Introduction

Source: Internet

Cây khế

Này em
Cây khế gãy rồi
Nỗi chua vẫn hỏi thăm
Người trồng cây.

Phùng Cung (1928–1977)

Boston housing

Size	Num of	Num of	⋮	Age of home	Price
($feet^2$)	bedrooms	floors		($years$)	($1000)
x_1	x_2	x_3		x_k	y
2081	5	1		45	420
1564	4	2		30	245
1356	3	2		26	332
x_{i1}	x_{i2}	x_{i3}		x_{ik}	y_i
$\dots$	$\dots$	$\dots$		$\dots$	$\dots$
x_{n1}	x_{n2}	x_{n3}		x_{nk}	y_n

Simple linear regression

$$y_i = \alpha + \beta x_i + \epsilon_k$$

- The plain truth is that when we only have two variables.

The model

- The multiple regression model assumes that:
  $$y_i = \alpha + \beta_1 x_{i1} + \ldots + \beta_k x_{ik} + \epsilon_i$$
  where
- y_i : dependent variable, continuous variable,
- $x_{i0} = 1$,x_{i1},. . . ,x_{ik} : independent variables,
- $\alpha(\beta_0)$,β_1,. . . ,β_{ik} : parameters, regression coefficients.
- Use actual data to calculate beta:
  $$\hat{y}_i = \alpha + \beta_1 x_{i1} + \ldots + \beta_k x_{ik}$$
- Residuals : $\epsilon_i = y_i$ - $\hat{y}_i$

Matrix, vector

$$y_i = \alpha + \beta_1 x_{i1} + \ldots + \beta_k x_{ik} + \epsilon_i$$
$$\hookrightarrow y = X\beta + \epsilon$$

where

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad X = \begin{bmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1k} \\ 1 & x_{21} & x_{22} & \ldots & x_{2k} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} & \ldots & x_{nk} \end{bmatrix} \quad \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} \quad \epsilon = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

- The model has $K + 1$ parameters.

Parameter estimates

- Least squares method – PP bình phương tối thiểu

Least squares function

$$\mathcal{L} = \sum_{i=1}^{N} \epsilon_i^2 = \sum_{i=1}^{N} (y_i - \sum_{j=0}^{k} \beta_j x_{ij})^2 = \|y - X\beta\|_2^2$$

Parameters of the model

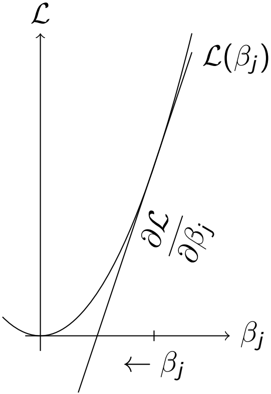
$$\frac{\partial \mathcal{L}}{\partial \beta_j} = -2 \sum_{i=1}^{N} (y_i - \sum_{j=0}^{k} \beta_j x_{ij}) x_{ij} = 0 \qquad j = 0, 1, \ldots, k$$

$$\hookrightarrow X^T (X\beta - y) = 0$$

- Solution:[1]
  $\beta = (X^T X)^\dagger X^T y$

[1] Least Squares, Pseudo-Inverses, PCA & SVD

Gradient Descent for Mutiple Variables



- Algorithm:

  Repeat {

  $$\beta_j := \beta_j - \alpha\frac{\partial\mathcal{L}}{\partial\beta_j}$$

  $$\hookrightarrow \beta_j - 2\alpha\sum_{i=1}^{n}(\sum_{j=0}^{k}\beta_j x_{ij} - y_i)x_{ij}$$

  (simultaneously update β_j for $j = 0, 1, \dots, k$)

  }

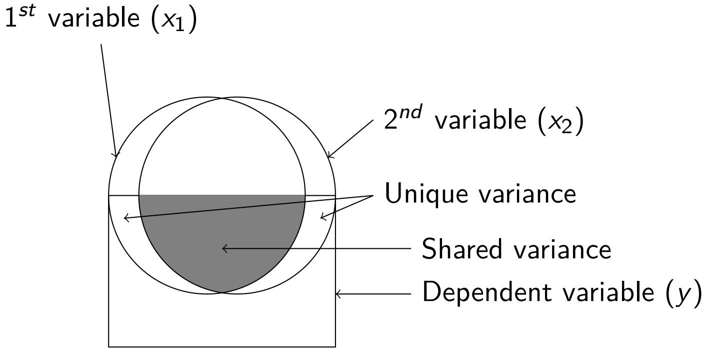
Goodness of Fit

All models are wrong, but some are useful.

– George Box

- Again we can look at the R-squared:
  $$R^2 = \frac{SS_{reg}}{SS_{total}} = 1 - \frac{SS_{error}}{SS_{total}}$$
- $SS_{total} = \sum_{i=1}^{n} (y_i - \bar{y})^2$ (Total of sum squares)
- $SS_{reg} = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2$ (SS due to the regression model)
- $SS_{error} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$ (SS due to random component)

Goodness of Fit



Hình: Shared variance in multiple regression.

- Multicollinearity : strong correlations among predictor variables can make it difficult to identify the unique relation between each predictor variable and the dependent variable. But doesn't affect the predicted value of the model.

Adjusted R-square

- The only drawback of R^2 is that if new predictors (X) are added to our model, R^2 only increases or remains constant but it never decreases. We can not judge that by increasing complexity of our model, are we making it more accurate?
- That is why, we use "Adjusted R-Square".
- The Adjusted R-Square is the modified form of R-Square that has been adjusted for the number of predictors in the model. It incorporates model's degree of freedom. The adjusted R-Square only increases if the new term improves the model accuracy.

$$R^2 \text{ adjusted} = 1 - \frac{(1 - R^2)(n-1)}{n-k-1}$$

Bootstrap method

- In many cases, there is no probabilistic theory to build a sample distribution (eg. Sample distribution for median).
- In that case we can bootstrap new datasets by choosing n data points with replacement from our data. And then we can compute the medians of those synthetic datasets.
- We can take the same approach to estimating the standard errors of our regression coefficients. We repeatedly take a bootstrap_sample of our data and estimate β based on that sample.

Ridge regression

Everything should be made as simple as possible, but not simpler.

– Albert Einstein

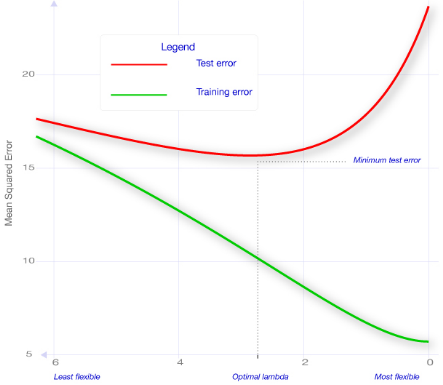
- Regularization is an approach in which we add to the error term a penalty that gets larger as β gets larger.
- Eg, in ridge regression, we add a penalty proportional to the sum of the squares of the β_j. (except that typically we don't penalize β_0 , the constant term).

$$\mathcal{L} = \sum_{i=1}^{N}(y_i - \sum_{j=0}^{k}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{k}\beta_j^2$$

- λ ($\lambda > 0$) is tuning parameters of model.
  λ is large, the β will approach zero.
  $\lambda \approx 0$, the results will be same as conventional regression.

Ridge regression

- A cross-validation approach is used to select the best value for λ



- Note : Usually you'd want to rescale your data before using this approach.

Gradient Descent for Ridge

- Repeat {

$$\beta_j := \beta_j - 2\alpha(\sum_{i=1}^{n}(y_i - \sum_{j=0}^{k}\beta_j x_{ij})x_{ij} + \lambda\beta_j)$$

}

When to use Ridge Regression

- Ridge regression may produce the best results when there are a large number of features. In cases where the number of features are greater than the number of observations, the matrix used in the normal equations may not be invertible. But using Ridge Regression enables this matrix to be inverted.

[1] Reference: Ridge Regression

Lasso Regression

- Another approach is *lasso regression*, which uses the penalty:
  $$\mathcal{L} = \sum_{i=1}^{N}(y_i - \sum_{j=0}^{k}\beta_j x_{ij})^2 + \lambda \sum_{j=1}^{k}|\beta_j|$$
- Whereas the ridge penalty shrank the coefficients overall, the lasso penalty tends to force coefficients to be 0, which makes it good for learning sparse models.

Practice

- Comparison between code in books and sklearn.
- An other data.

Questions?

References

- Joel Grus (2019)
  Data Science from Scratch, 2nd Edition
  O'Reilly Media
- Timothy C. Urdan (2016)
  Statistics in Plain English, 4th Edition, chapter 13
  Routledge
- Tuan V. Nguyen (2018)
  Phân tích dữ liệu với R
  Nhà Xuất Bản Tổng hợp TP.HCM
- Andrew Y. Ng (2011)
  Multivariate Linear Regression
  Coursera