Regression Analysis

Logistic Regression

Nicoleta Serban, Ph.D.
Professor
Stewart School of Industrial and Systems Engineering

Classification

Georgia Tech

About This Lesson

Georgia Tech

Classification Objective

Data: $\{(x_{1,1}, x_{1,2}, \cdots, x_{1,p}), \boldsymbol{Y}_1\}, \cdots, \{(x_{n,1}, x_{n,2}, \cdots, x_{n,p}), \boldsymbol{Y}_n\}$,
where $\boldsymbol{Y}_1, \cdots, \boldsymbol{Y}_n$ are *binary* responses

Model: Probability of success given predictor(s)

$$\mathrm{p} = (x_1, \cdots, x_p) = \Pr(Y = 1 \mid x_1, \cdots, x_p)$$

Objective: Classify (predict) a new binary response $\overset{*}{Y}$ based on observed predicting variables $x^*{}_1,..,x^*{}_p$

- Predicted probability:

$$\hat{\mathrm{p}}(x^*{}_1, \cdots, x^*{}_p) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 x^*{}_1 + \cdots + \hat{\beta}_p x^*{}_p}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 x^*{}_1 + \cdots + \hat{\beta}_p x^*{}_p}}$$

- If the predicted probability is large, then classify $\overset{*}{Y}$ as a success

Georgia Tech

Classification Objective

Data: $\{(x_{1,1}, x_{1,2}, \cdots, x_{1,p}), \boldsymbol{Y}_1\}, \cdots, \{(x_{n,1}, x_{n,2}, \cdots, x_{n,p}), \boldsymbol{Y}_n\}$,
where $\boldsymbol{Y}_1, \cdots, \boldsymbol{Y}_n$ are *binary* responses

Model: Probability of success given predictor(s)

$$\mathrm{p} = (x_1, \cdots, x_p) = \Pr(Y = 1 \mid x_1, \cdots, x_p)$$

Objective: Classify (predict) a new binary response $\overset{*}{Y}$ based on observed predicting variables $x^*{}_1,..,x^*{}_p$

- Predicted probability:

$$\hat{\mathrm{p}}(x^*{}_1, \cdots, x^*{}_p) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 x^*{}_1 + \cdots + \hat{\beta}_p x^*{}_p}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 x^*{}_1 + \cdots + \hat{\beta}_p x^*{}_p}}$$

- If the predicted probability is large, then classify $\overset{*}{Y}$ as a success

How good is the classification or prediction?

- Goodness of fit doesn't guarantee good prediction;
- If we have many models for classification, how do we choose among them?

Georgia Tech

Classification Error Rate

- **Predicted probability** given $x_1, \cdots, x_p$:

$$\hat{\mathrm{p}}(x_1, \cdots, x_p) = \frac{e^{\widehat{\beta}_0+\widehat{\beta}_1 x_1+\cdots+\widehat{\beta}_p x_p}}{1 + e^{\widehat{\beta}_0+\widehat{\beta}_1 x_1+\cdots+\widehat{\beta}_p x_p}}$$

- **Classifier:** $\mathrm{h}(x_1, \cdots, x_p) = \begin{cases} 1 \text{ if } \hat{\mathrm{p}}(x_1, \cdots, x_p) > \mathrm{r} \\ 0 \text{ otherwise} \end{cases}$,
  where r is a classification threshold between 0 and 1 (e.g., $\mathrm{r} = 1/2$)
- **Classification error rate:** $\mathrm{L}(\mathrm{h}) = 1 - \mathrm{Pr}\left(Y = \mathrm{h}(x_1, \cdots, x_p)\right)$
  - Training error
    - Use data to fit model, take proportion of responses misclassified
    - Biased downward as estimate of true classification error rate

Georgia Tech

Cross-Validation

Split the data $\{(x_{1,1}, x_{1,2}, \cdots, x_{1,p}), \boldsymbol{Y}_1\}, \cdots, \{(x_{n,1}, x_{n,2}, \cdots, x_{n,p}), \boldsymbol{Y}_n\}$, into:

- **Training Set**: Used to fit the model, i.e., to estimate $\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_p$
- **Testing/Validation Set**: Used to estimate the classification error rate

$$\hat{\mathrm{L}}(\mathrm{h}) = \frac{1}{\mathrm{m}} \mathrm{count}\left(\left(1 - \mathrm{h}(x_{i,1}, x_{i,2}, \cdots, x_{i,p})\right) = \boldsymbol{Y}_i\right), i \in \text{Validation Set},$$

where m is the size of the validation set

How to split the data?
- Random subsampling
- *k*-fold cross-validation (KCV)
  - Leave-one-out cross-validation (LOOCV)

Georgia Tech

Cross-Validation: How to Split Data?

Random Subsampling

- Randomly split the data into two portions (training and validation sets)
- Train on training set and test on validation set
- Randomly split multiple times
- Average the classification error rate across all random splits

***k*-fold cross-validation (KCV)**

- Randomly divide the data into k chunks (folds) of approximately equal size
- For i = 1 to k:
  - The training data consist of data without the i^{th} fold of data
  - The testing data consist of the i^{th} fold
  - Compute classification error rate $\hat{L}_i$ for the i^{th} fold testing data
  - Compute overall classification error: $\hat{L}(h) = \frac{1}{k}\sum_{i=1}^{k}\hat{L}_i$

Georgia Tech

Cross-Validation: How to Split Data?

Random Subsampling

- Randomly split the data into two portions (training and validation sets)
- Train on training set and test on validation set
- Randomly split multiple times
- Average the classification error rate across all random splits

***k*-fold cross-validation (KCV)**

- Randomly divide the data into k chunks (folds) of approximately equal size
- For i = 1 to k:
  - The training data consist of data without the i^{th} fold of data
  - The testing data consist of the i^{th} fold
  - Compute classification error rate $\hat{L}_i$ for the i^{th} fold testing data
  - Compute overall classification error: $\hat{L}(h) = \frac{1}{k}\sum_{i=1}^{k}\hat{L}_i$

Random CV or *k*-fold CV?

- Random subsampling is computationally more expensive than k-fold CV

How to choose *k*?

- Leave-one-out CV is KCV with $k = n$
  - Less computationally efficient than KCV
- The larger the k, the less bias but the more variance

Georgia Tech

Summary