Logistic Regression

B. Ravindran

Problem Setting

- $\mathcal{X} \subset \mathcal{R}$ is the input space
  - $\mathbf{X} = (X_1, X_2, \ldots,$ is a random variable describing the input
- $\mathcal{Y} \subset$ is the output space with K number of classes.
  - $\mathbf{Y}$ is a random variable describing the output.
- $\Pr(\mathbf{X}, \mathbf{Y}) = \Pr(\mathbf{Y} \mid \mathbf{X})$ is the data distribution
  - $\Pr(\mathbf{Y} \mid \mathbf{X} =$ is the predicted output probabilities given an input $\bar{x} \in$

Problem Setting

- We are interested in the probability of a class given the data point:

$$\Pr(\mathbf{Y} = k \mid \mathbf{X} =$$

- If the above probability is known for all K classes, we can predict the label as:

$$\hat{y} = \arg\max_{k} \Pr(\mathbf{Y} = k \mid \mathbf{X}$$

Assumption

Logistic Regression (LR) assume that the log-odds are linear.

$$\log\left(\frac{\Pr(\mathbf{Y}=1\,|\,\mathbf{X}=\bar{x})}{\Pr(\mathbf{Y}=K\,|\,\mathbf{X}=\bar{x})}\right) = \beta_{10} + \bar{\beta}_1^T\bar{x}$$

$$\log\left(\frac{\Pr(\mathbf{Y}=2\,|\,\mathbf{X}=\bar{x})}{\Pr(\mathbf{Y}=K\,|\,\mathbf{X}=\bar{x})}\right) = \beta_{20} + \bar{\beta}_2^T\bar{x}$$

$$\vdots$$

Assumption

Logistic Regression (LR) assume that the log-odds are linear.

$$\log\left(\frac{\Pr(\mathbf{Y}=1\,|\,\mathbf{X}=\bar{x})}{\Pr(\mathbf{Y}=K\,|\,\mathbf{X}=\bar{x})}\right) = \beta_{10} + \bar{\beta}_1^T \bar{x}$$

$$\log\left(\frac{\Pr(\mathbf{Y}=2\,|\,\mathbf{X}=\bar{x})}{\Pr(\mathbf{Y}=K\,|\,\mathbf{X}=\bar{x})}\right) = \beta_{20} + \bar{\beta}_2^T \bar{x}$$

$$\vdots$$

$$\Pr(\mathbf{Y}=1\,|\,\mathbf{X}=\bar{x}) = \frac{\exp\left(\beta_{10} + \bar{\beta}_1^T \right.}{1 + \sum_{l=1}^{K-1} \exp\left(\beta_l \right.}$$

$$\Pr(\mathbf{Y}=2\,|\,\mathbf{X}=\bar{x}) = \frac{\exp\left(\beta_{20} + \bar{\beta}_2^T \right.}{1 + \sum_{l=1}^{K-1} \exp\left(\beta_l \right.}$$

1

Fitting LR models

- We maximize the likelihood of betas given the dataset.
- We assume that all the N data pairs are observed independently.
- Maximizing likelihood is equivalent tp maximizing log-likelihood which is defined as:

$$\mathcal{L}(\boldsymbol{\beta}) = \sum^{N} \log \Pr(\mathbf{Y} = y_i \mid$$

- We can optimize the above equation using any gradient descent based algorithm.

Fitting LR model for 2 class setting

- Assume there are 2 classes: Class 1 and Class 0.

- We want to maximize the log-likelihood which is defined as:

$$\mathcal{L}\big([\beta_0;\bar{\beta}]\big) \;=\; \sum^{N} y_i \,\log\left(P(\bar{x}_i)\right) \;+\; (1-y_i)\log$$

$$\text{where,}\quad P(\bar{x}_i) = \Pr(\mathbf{Y} = 1 \,|$$

Fitting LR model for 2 class setting

Rearranging log-likelihood

$$\mathcal{L}\big(\big[\beta_0;\bar{\beta}\big]\big) \;=\; \sum_{i=1}^{N} y_i \,\log\left(P(\bar{x}_i)\right) \,+\, (1-y_i)\log\left(1-P(\right.$$

$$=\sum_{i=1}^{N} \,\log\left(1-P(\bar{x}_i)\right) \,+\, y_i \log\left(\frac{P(\bar{x}_i)}{1-P(\bar{x}_i)}\right.$$

Fitting LR model for 2 class setting

- To maximize log-likelihood, we have to differentiate $\mathcal{L}$ w.r.t. **beta** and set it to zero.

$$\frac{\partial \mathcal{L}}{\partial \beta_i} = \sum^{N} \left(y_i - P\left(\bar{x}_i \,; \, [\beta_0; \bar{\beta}]\right)\right)$$

for notational simplicity we will refer $\bar{\beta} = [\beta_0;$

Fitting LR model for 2 class setting

- To maximize log-likelihood, we have to differentiate $\mathcal{L}$ w.r.t. **beta** and set it to zero.

$$\frac{\partial \mathcal{L}}{\partial \beta_i} = \sum^{N} \Big(y_i - P\big(\bar{x}_i \, ; \, \bar{\beta}\big)\Big)\bar{x}$$

- It is not easy to find **beta**, that satisfies the above equation. We can use an iterative algorithm to find **beta**. One such algorithm is called Newton-Raphson algorithm.

Fitting LR model for 2 class setting

Newton-Raphson method:

$$\bar{\beta}_{new} = \bar{\beta}_{old} - \left(\frac{\partial^2 \mathcal{L}}{\partial \bar{\beta}\, \partial \bar{\beta}^T} \right)$$

Rewriting the notations in vector/matrix form, say

$$\text{data point} \qquad \mathbf{X} : N \times (p+1)$$

$$\text{probability} \qquad \bar{P} : N \times 1 \text{, where } \bar{P}$$

Fitting LR model for 2 class setting

Newton-Raphson method:

$$\bar{\beta}_{new} = \bar{\beta}_{old} - \left(\frac{\partial^2 \mathcal{L}}{\partial \bar{\beta}\, \partial \bar{\beta}^T} \right.$$

$$\frac{\partial \mathcal{L}}{\partial \beta_j} = \sum^{N} \Big(y_i - P\big(\bar{x}_i \, ; \, [\beta_0;$$

$$\Longrightarrow$$

$$\frac{\partial \mathcal{L}}{\partial \bar{\beta}} = \mathbf{X}^T \left(\bar{y} - \bar{P} \right)$$

$$\frac{\partial^2 \mathcal{L}}{\quad} = - \, \mathbf{X}^T$$

Fitting LR model for 2 class setting

Therefore, Newton-Raphson method:

$$\bar{\beta}_{new} = \bar{\beta}_{old} - \left(\frac{\partial^2 \mathcal{L}}{\partial \bar{\beta}\, \partial \bar{\beta}^T} \right)^{-1} \frac{\partial \mathcal{L}}{\partial \bar{\beta}}$$

$$= \bar{\beta}_{old} + \left(\mathbf{X}^T \mathbf{W} \mathbf{X}\right)^{-1} \mathbf{X}^T \left(\bar{y} - \bar{P}\right)$$

Fitting LR model for 2 class setting

Therefore, Newton-Raphson method:

$$\bar{\beta}_{new} = \bar{\beta}_{old} - \left(\frac{\partial^2 \mathcal{L}}{\partial \bar{\beta}\, \partial \bar{\beta}^T} \right)^{-1} \frac{\partial \mathcal{L}}{\partial \bar{\beta}}$$

$$= \bar{\beta}_{old} + \left(\mathbf{X}^T \mathbf{W} \mathbf{X} \right)^{-1} \mathbf{X}^T \left(\bar{y} - \bar{P} \right)$$

The above solution can also be obtained from solving weighted least square:

$$\bar{\beta}_{new} = \arg\min_{\bar{\beta}} \left(\bar{z} - \mathbf{X}\bar{\beta} \right)^T \mathbf{W} \left(\bar{z} - \mathbf{X}\bar{\beta} \right), \quad \text{where } \bar{z} = \left(\mathbf{X}\bar{\beta}_{old} \right. -$$

LDA vs LR

- Both produce linear boundaries.

- LDA assumes that the observations are drawn from the normal distribution with common variance in each class, while logistic regression does not have this assumption.

- Logistic regression is unstable when the classes are well separated.

- In the case where N is small, and the distribution of predictors X is approximately normal, then LDA is more stable than Logistic Regression.

Summary

- Logistic Regression is a classification approach!
- Assumes that the class probabilities are given by a logit or sigmoid function.
- Directly models the separating surface as a linear function.
- Especially popular in binary classification.
- Can be combined with Lasso to yield a sparse classifier.