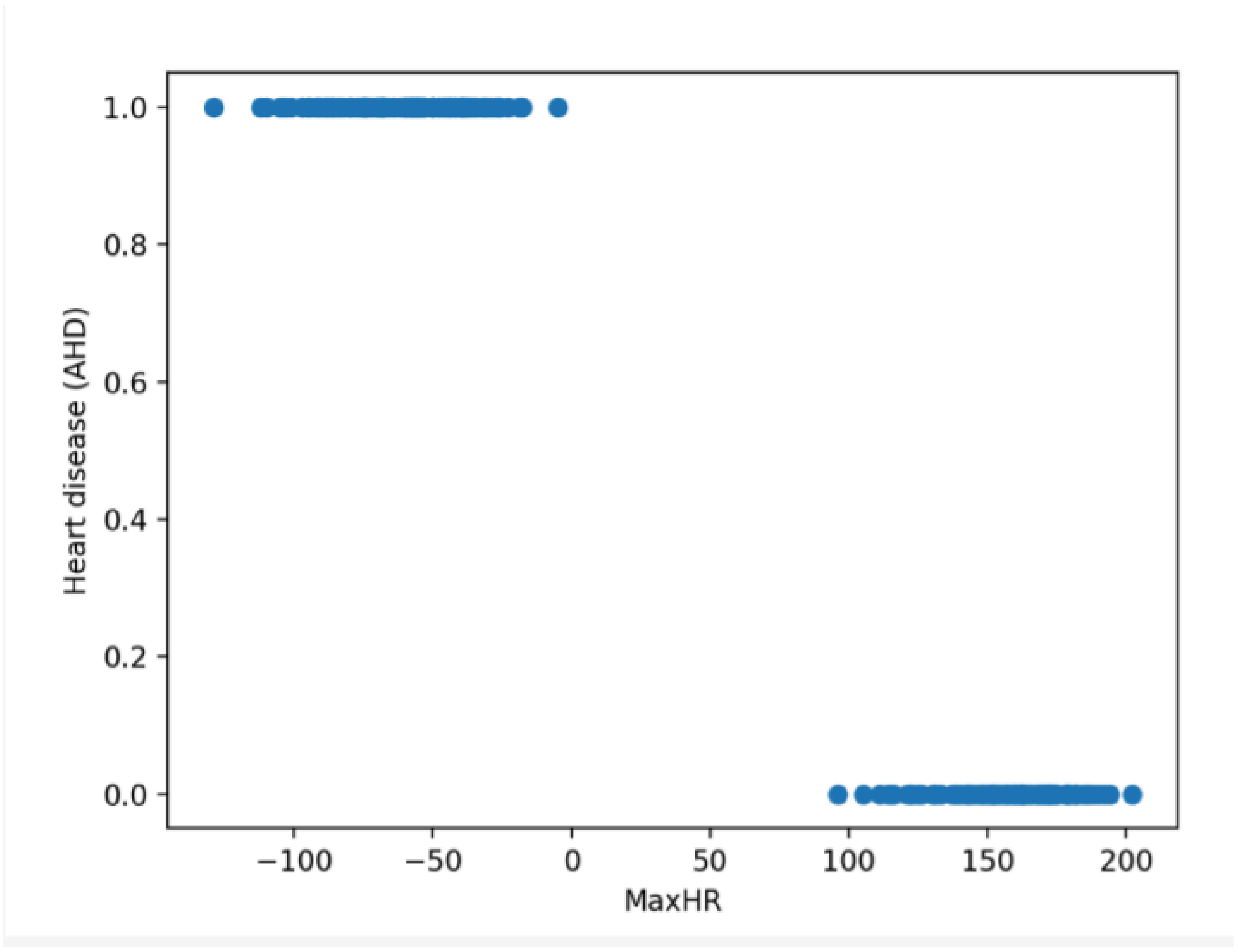
Regularization in Logistic Regression

Bookmark this page

Motivation

The problem of separation arises when the binary responses can be separated into all 0s and all 1s as a linear function of the predictors, as in the image below.



The Maximum Likelihood Estimation (MLE) does not exist when the data is perfectly separable. The likelihood always increases with the magnitude of the estimated coefficients. To solve this issue, we need to use regularization.

Regularization in Logistic Regression

Based on the Likelihood framework, a loss function can be determined based on the log-likelihood function. We saw in linear regression that maximizing the log-likelihood is equivalent to minimizing the sum of squares.

$$\arg\min \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \arg\min \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_{1i} + \ldots + \beta_p x_{pi}))^2$$

And a regularization approach was to add a penalty factor to this equation, which for Ridge Regression becomes:

$$\arg\min \left[\sum_{i=1}^{n} \left(y_i - (\beta_0 + \sum_{j=1}^{n} (\beta_j x_{ji})\right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right]$$

Note: this penalty shrinks the estimates towards zero, and has the analogue of using a Normal prior centered at zero in the Bayesian paradigm.

A similar approach can be used in logistic regression. Here, maximizing the log-likelihood is equivalent to minimizing the following loss function:

$$\arg\min \left[-\sum_{i=1}^{n} (y_i \ln(p_i) + (1 - y_i) \ln(1 - p_i)) \right]$$

where

$$p_i = \left(1 - e^{-\beta_0 + \beta_1 x_{1i} + \ldots + \beta_p x_{pi}}\right)^{-1}$$

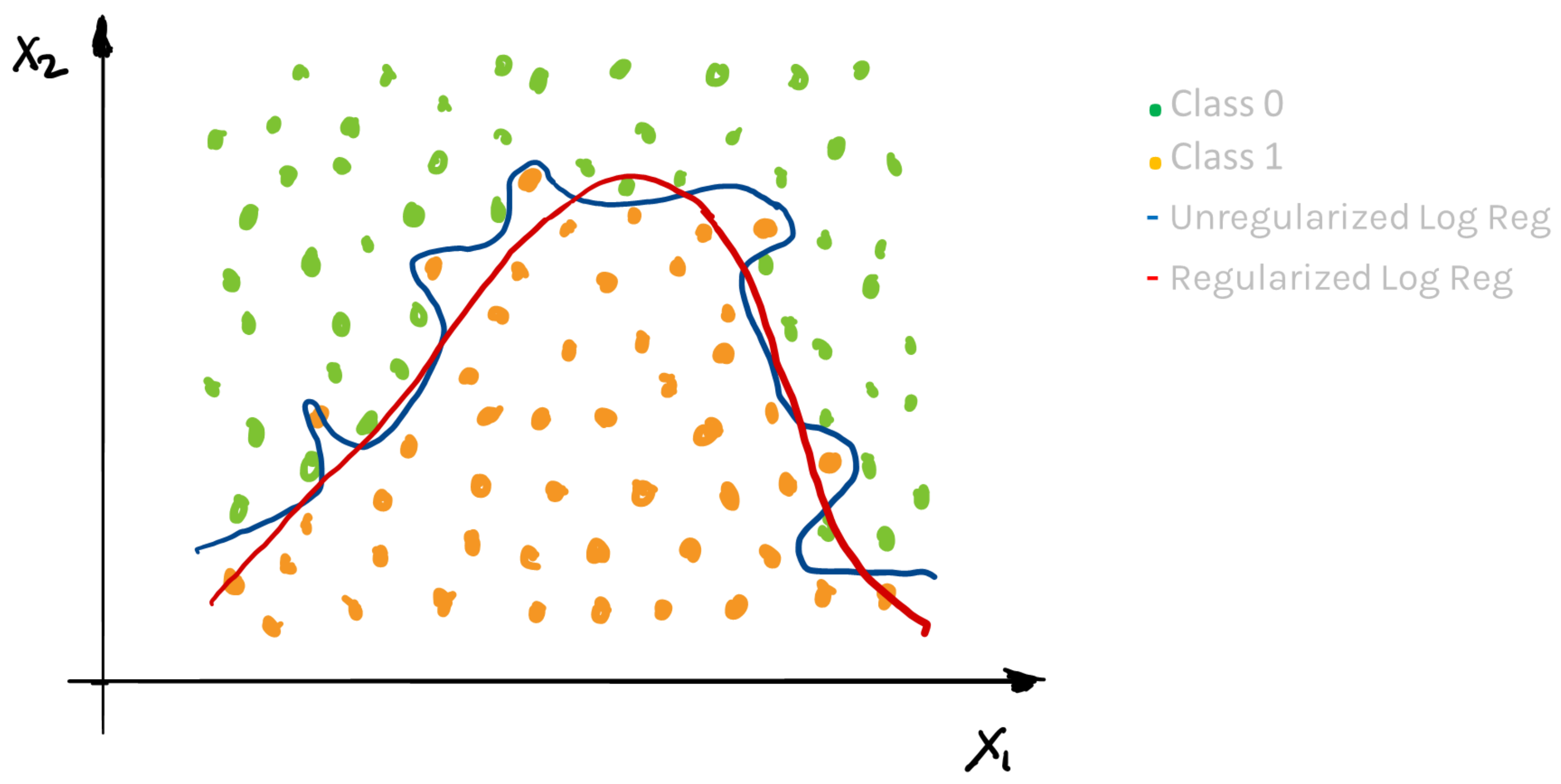
A penalty factor can then be added to this loss function and results in a new loss function that penalizes large values of the parameters:

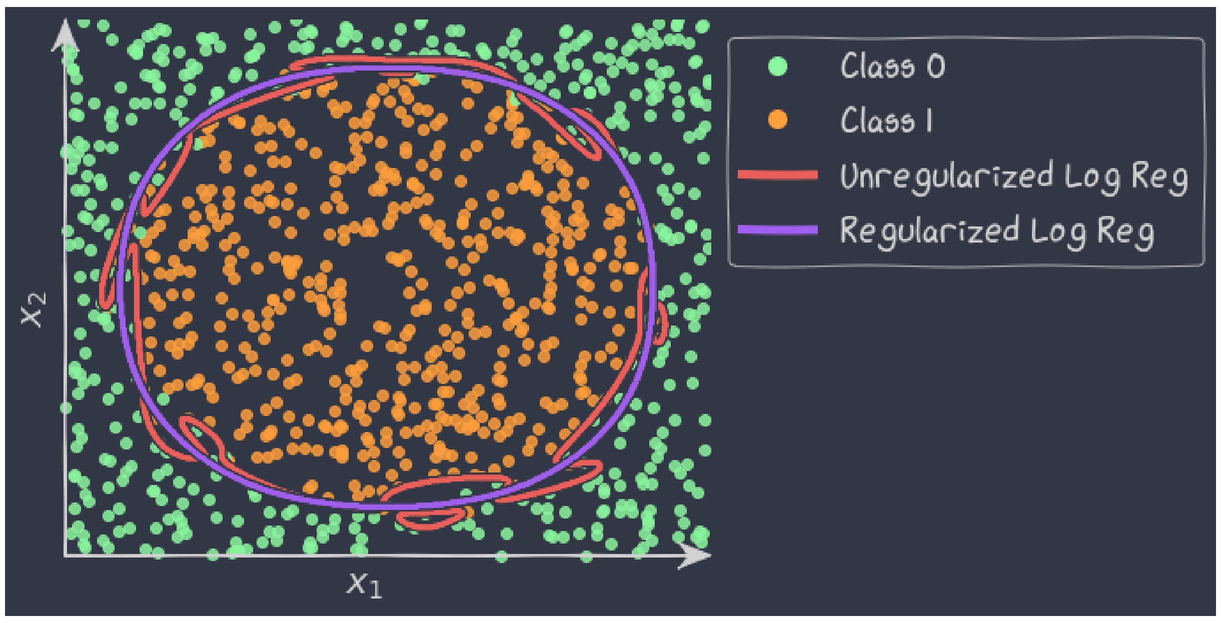
Calculator

$$\arg\min\left[-\sum_{i=1}^{n}\left(y_i\ln\left(p_i\right)+\left(1-y_i\right)\ln\left(1-p_i\right)\right)+\lambda\sum_{j=1}^{p}\beta_j^2\right]$$

The result is just like in linear regression: it shrinks the parameter estimates towards zero. In practice, the intercept is usually not part of the penalty factor. Note: the sklearn package uses a different tuning parameter: instead of λ they use a constant that is essentially $C = 1/\lambda$.

Just like in linear regression, the shrinkage factor must be chosen. Through building multiple training and test sets (cross-validation), we can select the best shrinkage factor to mimic out-of-sample prediction.





Discussion Board (External resource)

Haga clic en Aceptar para que su nombre de usuario y dirección de correo electrónico se envíen a una aplicación de terceros.

Aceptar

< Previous   Next >

[edX for Business]
[Open edX]
[Careers]
[News]

Legal

[Terms of Service & Honor Code]
[Privacy Policy]
[Accessibility Policy]
[Trademark Policy]
[Sitemap]
[Cookie Policy]
Your Privacy Choices

Connect

[Idea Hub]
[Contact Us]
[Help Center]
[Security]
[Media Kit]

Calculator