STA 314: Statistical Methods for Machine Learning I

Lecture - Logistic Regression in Binary Classification

Xin Bing (Fred for today :o)

Department of Statistical Sciences
University of Toronto

Review

- In classification, $X \in \mathcal{X}$ and $Y \in C = \{0, 1, \ldots, K-1\}$.

$\in \mathbb{R}^p$ $\downarrow \; - \; - \; . \; K-1$

- The Bayes rule — → argmax$_k$ f: find $k \in C$ s.t. f is maximized

$$f^*(\mathbf{x}) = \arg\max_{k \in C} \mathbb{P}\{Y = k \mid X = \mathbf{x}\}, \qquad \forall \mathbf{x} \in \mathcal{X}$$

has the smallest expected error rate.

$C = \{0, 1\}$

- For binary classification, our goal is to estimate

$$p(\mathbf{x}) := \mathbb{P}(Y = 1 \mid X = \mathbf{x}), \qquad \forall \mathbf{x} \in \mathcal{X}$$

↑ simplicity

$1 - p(x) = P(Y = 0 \mid X = x)$

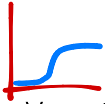
Logistic Regression

Logistic Regression is a parametric approach that postulates parametric structure on the function $p : \mathcal{X} \mapsto [0, 1]$.

$\mapsto p = P(Y=1|x)$

- It is assumed that

$$p(\mathbf{x}) := p(\mathbf{x}; \boldsymbol{\beta}) = \frac{e^{\beta_0+\beta_1 x_1+\cdots+\beta_p x_p}}{1 + e^{\beta_0+\beta_1 x_1+\cdots+\beta_p x_p}}, \qquad \forall \mathbf{x} \in \mathcal{X}.$$



The function $f(t) = e^t/(1 + e^t)$ is called the logistic function. $\beta_0, \ldots, \beta_p$ are the parameters.

- We always have $0 \leq p(\mathbf{x}) \leq 1$.
- Note that $p(\mathbf{x}; \boldsymbol{\beta})$ is **NOT** a linear function either in $\mathbf{x}$ or in $\boldsymbol{\beta}$.

Logistic Regression

$p(x) = P(Y=1 \mid x)$ $\quad 1-p(x) = P(Y=0 \mid x)$ $\quad p(x) = \frac{e^{\beta_0 + \vec{x}\vec{\beta}}}{1+e^{\beta_0+\vec{x}\vec{\beta}}}$

- A bit of rearrangement gives

$\frac{p(x)}{1-p(x)}$

$$\underbrace{\frac{p(\mathbf{x})}{1-p(\mathbf{x})}}_{\text{odds}} = e^{\beta_0+\beta_1 x_1+\cdots+\beta_p x_p},$$

$1 - p(x) = \frac{1}{1+e^{\beta_0+\vec{x}\vec{\beta}}}$

$$\underbrace{\log\left[\frac{p(\mathbf{x})}{1-p(\mathbf{x})}\right]}_{\text{log-odds (a.k.a. logit)}} = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p.$$

if we see this in Linear Reg

$\beta \to$ change of effect for Y on x

odds $\in [0, \infty)$ and log-odds $\in (-\infty, \infty)$.

- Similar interpretation as linear models[1]

[1] Each β_j represents the change of log-odds for one unit increase in X_j (with other features held fixed).

Logistic regression

Our interests:

- **Prediction**: for any $\mathbf{x}_0 \in \mathcal{X}$, classify its corresponding label y_0.
- **Estimation**: how to estimate the vector of β by using our training data?

$\beta_0, \ldots \beta_1, \ldots \beta_p$

Prediction at different levels under logistic regression

Let $\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \ldots, \hat{\beta}_p)$ be any estimates of $\boldsymbol{\beta}$. → log odds

$\hat{f}: \mathbb{R}^p \to C$

- Prediction of **the logit** at $\mathbf{x} \in \mathcal{X}$:

$$\log\left(\frac{p(x)}{1-p(x)}\right) = \widehat{\text{logit}}(\mathbf{x}) = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p.$$

- Prediction of **the conditional probability** $p(\mathbf{x}) = \mathbb{P}(Y = 1 | X = \mathbf{x})$:

$$\hat{p}(\mathbf{x}) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p}} = \hat{P}(Y=1 | x)$$

- Classify **the label** Y at $X = \mathbf{x}$:

$$\hat{y} = \begin{cases} 1, & \text{if } \hat{p}(\mathbf{x}) \geq 0.5; \\ 0, & \text{otherwise.} \end{cases}$$

Maximum Likelihood Estimator (MLE)

Given $\mathcal{D}^{train} = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\}$ with $y_i \in \{0, 1\}$, we estimate the parameters by **maximizing the likelihood** of $\mathcal{D}^{train}$.

The maximum likelihood principle

We seek the estimates of parameters such that the fitted probability are the closest to the individual's observed outcome.

$L(\theta, x_i, y_i) \rightarrow l(\theta) = \log L(\theta)$

$\rightarrow \frac{\partial l}{\partial \beta} \rightarrow 0$

Computation of the MLE under Logistic Regression

General steps of computing the MLE:

$L(\theta)$ $L(\beta)$

- Write down the likelihood, as always!

- Solve the optimization problem.

Likelihood under Logistic Regression

$$\text{odds} = \frac{p(x)}{1-p(x)} = \text{odd-ratio}$$

For simplicity, let us set $\beta_0 = 0$ such that

$$p(\mathbf{x};\boldsymbol{\beta}) = \frac{e^{\mathbf{x}^\top\boldsymbol{\beta}}}{1+e^{\mathbf{x}^\top\boldsymbol{\beta}}}, \qquad 1 - p(\mathbf{x};\boldsymbol{\beta}) = \frac{1}{1+e^{\mathbf{x}^\top\boldsymbol{\beta}}}.$$

The data consists of $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ with

$$y_i \sim \text{Bernoulli}(p(\mathbf{x}_i;\boldsymbol{\beta})), \qquad 1 \le i \le n.$$

$$\Leftrightarrow \quad y_i = \begin{cases} 1 & \text{with } p(x_i;\beta) \\ 0 & \text{with } 1 - p(x_i;\beta) \end{cases}$$

- What is the likelihood of $(\mathbf{x}_i, y_i)$?

$$p(x_i;\beta)^{y_i} \cdot (1-p(x_i;\beta))^{1-y_i}$$

Likelihood under Logistic Regression

The likelihood of each data point $(\mathbf{x}_i, y_i)$ at any $\boldsymbol{\beta}$ is

$$L(\boldsymbol{\beta}; \mathbf{x}_i, y_i) \propto [p(\mathbf{x}_i; \boldsymbol{\beta})]^{y_i} [1 - p(\mathbf{x}_i; \boldsymbol{\beta})]^{1-y_i}$$

with

$$p(\mathbf{x}_i; \boldsymbol{\beta}) = \frac{e^{\mathbf{x}_i^\top \boldsymbol{\beta}}}{1 + e^{\mathbf{x}_i^\top \boldsymbol{\beta}}}.$$

The sign $\propto$ means "proportional to, up to some multiplicative term that does not involve the parameter $\boldsymbol{\beta}$.

The joint likelihood of $\boxed{\text{all data points is}}$ $(x_1, y_1) \cdots (x_n, y_n)$.

$$\underline{L(\boldsymbol{\beta})} = \prod_{i=1}^{n} [p(\mathbf{x}_i; \boldsymbol{\beta})]^{y_i} [1 - p(\mathbf{x}_i; \boldsymbol{\beta})]^{1-y_i}.$$

$\hookrightarrow \log L(\beta) = \ell(\beta)$.

Log-likelihood under Logistic Regression

The log-likelihood at any $\boldsymbol{\beta}$ is

$$
\begin{aligned}
\ell(\boldsymbol{\beta}) &= \log\left\{\prod_{i=1}^{n}[p(\mathbf{x}_i;\boldsymbol{\beta})]^{y_i}[1-p(\mathbf{x}_i;\boldsymbol{\beta})]^{1-y_i}\right\} \\
&= \sum_{i=1}^{n}[y_i\log(p(\mathbf{x}_i;\boldsymbol{\beta}))+(1-y_i)\log(1-p(\mathbf{x}_i;\boldsymbol{\beta}))] \\
&= \sum_{i=1}^{n}\left[y_i\log\left(\frac{p(\mathbf{x}_i;\boldsymbol{\beta})}{1-p(\mathbf{x}_i;\boldsymbol{\beta})}\right)+\log(1-p(\mathbf{x}_i;\boldsymbol{\beta}))\right] \\
&= \sum_{i=1}^{n}\left[y_i\mathbf{x}_i^\top\boldsymbol{\beta}-\log\left(1+e^{\mathbf{x}_i^\top\boldsymbol{\beta}}\right)\right].
\end{aligned}
$$

(verify)

maximize $\ell(\beta) \Leftrightarrow$ minimize $-\ell(\beta)$.

How to compute the MLE?

How do we maximize the log-likelihood

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left[y_i \mathbf{x}_i^\top \boldsymbol{\beta} - \log\left(1 + e^{\mathbf{x}_i^\top \boldsymbol{\beta}}\right) \right]$$

for logistic regression?

- It is equivalent to minimize $-\ell(\boldsymbol{\beta})$ over $\boldsymbol{\beta}$. $\rightarrow \frac{\partial \ell}{\partial \beta} = 0$
- No direct solution: taking derivatives of $\ell(\boldsymbol{\beta})$ w.r.t. $\boldsymbol{\beta}$ and setting them to 0 doesn't have an explicit solution.
- Need to use iterative procedure.

Gradient descent for solving the MLE under logistic regression

Recall we would like to solve

(if anything confusing, GD lecture note 1.)

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^p} -\ell(\boldsymbol{\beta})$$

where

$$-\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left[-y_i \mathbf{x}_i^\top \boldsymbol{\beta} + \log\left(1 + e^{\mathbf{x}_i^\top \boldsymbol{\beta}}\right) \right].$$

The gradient at any $\boldsymbol{\beta}$ is that, for any $j \in \{1, \ldots, p\}$,

$$-\frac{\partial \ell(\boldsymbol{\beta})}{\partial \beta_j} = \sum_{i=1}^{n} \left[-y_i + \frac{e^{\mathbf{x}_i^\top \boldsymbol{\beta}}}{1 + e^{\mathbf{x}_i^\top \boldsymbol{\beta}}} \right] x_{ij} \qquad \text{(verify this!)}$$

Updates and stopping criteria

Therefore, at the $(k+1)$th iteration, with the learning rate α,

$$\hat{\boldsymbol{\beta}}^{(k+1)} = \hat{\boldsymbol{\beta}}^{(k)} - \alpha \sum_{i=1}^{n} \left[-y_i + \frac{e^{\mathbf{x}_i^\top \hat{\boldsymbol{\beta}}^{(k)}}}{1 + e^{\mathbf{x}_i^\top \hat{\boldsymbol{\beta}}^{(k)}}} \right] \mathbf{x}_i .$$

any state $\beta^{(k)}$.

Initialization $\boldsymbol{\beta}^{(0)} = 0$.

↳ non-informative.

- The objective value stops changing: $|\ell(\hat{\boldsymbol{\beta}}^{(k+1)}) - \ell(\hat{\boldsymbol{\beta}}^{(k)})|$ is small, say, $\leq 10^{-6}$.
- The parameter stops changing: $||\hat{\boldsymbol{\beta}}^{(k+1)} - \hat{\boldsymbol{\beta}}^{(k)}||_2$ is small or $||\hat{\boldsymbol{\beta}}^{(k+1)} - \hat{\boldsymbol{\beta}}^{(k)}||_2 / ||\hat{\boldsymbol{\beta}}^{(k)}||_2$ is small.
- Stop after M iterations for some specified M, e.g. $M = 1000$.

Gradient descent for solving the MLE under logistic regression

part : $L(\beta) \leftarrow y_i \sim$ Bernoulli.

- The negative log-likelihood

part : $\rightarrow$ GD.

$$-\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left[-y_i \mathbf{x}_i^\top \boldsymbol{\beta} + \log\left(1 + e^{\mathbf{x}_i^\top \boldsymbol{\beta}}\right)\right]$$

is convex in $\boldsymbol{\beta}$ (check this).

- So we can use gradient descent to find the MLE.

$\hat{\beta}^{(M)} \approx \hat{\beta}_{MLE}$

M : # of iteration

Why MLE?

The MLE, whenever can be computed, has many nice properties!

- Asymp. consistent $\rightarrow n \to \infty$

$$\hat{\beta} - \beta \to 0, \quad \text{in probability as } n \to \infty.$$

- Asymp. normal

covariance Σ

$$\sqrt{n}\left(\hat{\beta} - \beta\right) \to N(0, \Sigma) \quad \text{in distribution as } n \to \infty.$$

$\vec{0}$

- Asymp. efficient:

Σ is the "smallest" among all asymptotic unbiased estimators.

Any downsides? computation, model misspecification ...

Inference under logistic regression

$Z = \frac{\hat{\mu} - \mu}{se(\hat{\mu})}$

Let $\hat{\boldsymbol{\beta}}$ be the MLE of $\boldsymbol{\beta}$.

- Z-statistic is similar to t-statistic in regression, and is defined as

$$\frac{\hat{\beta}_j - \beta_j}{SE(\hat{\beta}_j)}, \quad \forall j \in \{0, 1, \dots, p\}$$

$H_0: \mu = 0$

  where $SE(\hat{\beta}_j)$ is the asymp. variance of $\hat{\beta}_j$ (equal to $\hat{\Sigma}_{jj}/n$ in the previous slide).

- It produces p-value for testing the null hypothesis

$$H_0 : \beta_j = 0 \quad \text{v.s.} \quad H_1 : \beta_j \neq 0.$$

  A large (absolute) value of the z-statistic or small p-value indicates evidence against H_0.

Example: Default data

{student , non-student}

Suppose that we are interested in predicting

the probability of default for a given customer

by using **student status** as the only feature.

By encoding $x_i = 1\{\text{the } i\text{th customer is student}\}$ and, $y_i = 1$ if default happens and 0 otherwise. Fit the logistic regression model

$$y_i \sim \text{Bernoulli}(p(x_i)), \qquad p(x_i) = \frac{e^{\beta_0+\beta_1 x_i}}{1+e^{\beta_0+\beta_1 x_i}}.$$

$\downarrow$

$P(Y=1|x) = P($customer default | student status

Prediction of $p(x)$

The fitted maximum likelihood estimates of β_0 and β_1 satisfy:

	Coefficient	Std.Error	Z-statistic	P-value
β_0 Intercept	-3.5	0.071	-49.55	<0.0001
β_1 student[Yes]	0.405	0.115	3.52	0.0004

$P(Y=1 \mid x=1)$

$$\hat{p}(x=1) = \hat{\mathbb{P}}(\text{default} \mid \text{student}) = \frac{e^{-3.5+0.405\times 1}}{1+e^{-3.5+0.405\times 1}} \approx 0.043$$

$$\hat{p}(x=0) = \hat{\mathbb{P}}(\text{default} \mid \text{non-student}) = \frac{e^{-3.5+0.405\times 0}}{1+e^{-3.5+0.405\times 0}} \approx 0.029$$

$P(Y=1 \mid x=0)$.

$0.405 \rightarrow$ the effect of student status (non-stud v.s student) that it has on log odds of customer default

$$\text{log odd} = \log\left(\frac{P(x)}{1-P(x)}\right) =$$

$$\text{odds ratio} \quad \frac{P(x)}{1-P(x)} = e^{0.405} \approx$$

$\rightarrow$ how much more likely customer is defaulting v.s not defaulting

Example: Default data

Consider using more predictors: **balance**(X_1), **income**(X_2), and **student status**(X_3).

$$\log\left(\frac{p(\mathbf{x}_i)}{1-p(\mathbf{x}_i)}\right) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3}$$

The maximum likelihood estimates yield:

	Coefficient	Std.Error	Z-statistic	P-value
Intercept	-10.87	0.492	-22.08	<0.0001
balance	0.006	0.0002	24.74	<0.0001
income	0.003	0.0082	0.37	0.712
student[Yes]	-0.647	0.2362	-2.74	0.0062

change of dir

Question: how does the coefficient of student status changes?

Metrics used for evaluating classifiers

In classification, we have several metrics that can be used to evaluate a given classifier.

- The most commonly used metric is the overall classification accuracy.

- For **binary** classification, there are a few more out there.....

Cont'd example: the Default Data

- Classify whether or not an individual will default on the basis of credit card balance and student status.

- **The confusion matrix** of fitted logistic regression

		True default status		
		No	Yes	Total
Predicted default status	No	9,644	252	9,896
	Yes	23	81	104
	Total	9,667	333	10,000

- The training error rate is $(23 + 252)/10000 = 2.75\%$.

Type of Errors for binary classification

		True default status		
		No	Yes	Total
Predicted	No	9,644	$\boxed{252}$	9,896
default status	Yes	23	81	104
	Total	9,667	333	10,000

No

1. **False positive rate (FPR)**: The fraction of negative examples that are classified as positive: 23/9667 = 0.2% in default data.

(Yes)

(Yes)

2. **False negative rate (FNR)**: The fraction of positive examples that are classified as negative: 252/333 = 75.7% in default data.[2]

(No). 252/(252+81) = 75.7%

[2]For a credit card company that is trying to identify high-risk individuals, the error rate 75.7% among individuals who default is unacceptable.

Control the false negative rate

Q: How to modify the logistic classifier to lower the false negative rate?

the fraction of **positive** examples as **negative**

the fraction of **default** examples classified as **non-default**

$$p(x) = \frac{e^{\vec{x}\vec{\beta}}}{1+e^{\vec{x}\vec{\beta}}}.$$

- The current classifier is based on the rule

$$\begin{aligned}
\hat{y}_i &= 1 \quad \text{(default)}, &&\text{if} \quad \hat{\mathbb{P}}(\text{default} = \textit{yes} \mid X = \mathbf{x}_i) \geq 0.5 \\
\hat{y}_i &= 0 \quad \text{(non-default)}, &&\text{otherwise.}
\end{aligned}$$

Control the false negative rate

- To lower FNR, we reduce the number of negative predictions. Classify $X = \mathbf{x}$ to *yes* if

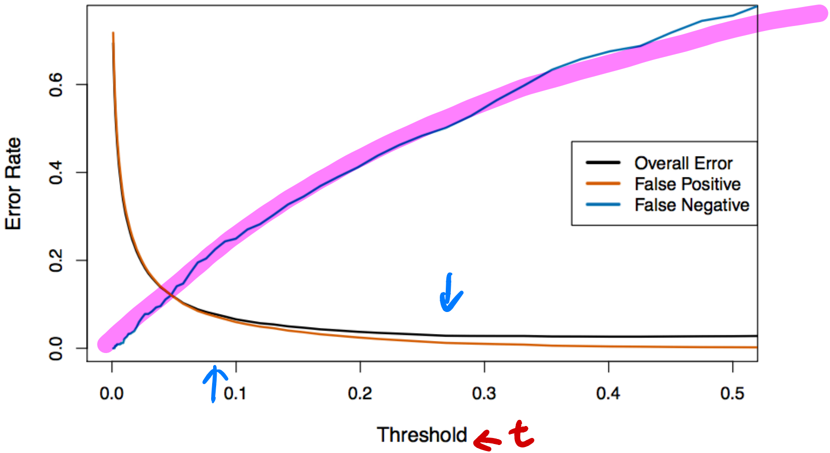
$$\hat{\mathbb{P}}(Y = yes \mid X = \mathbf{x}) \geq t.$$

for some $0 \leq t < 0.5$. to further ← 0.5

  - Why starts with $t = 0.5$? ←
  - What happens for $t = 0$? → alway predict yes
  - What happens for $t = 1$? → always predict no

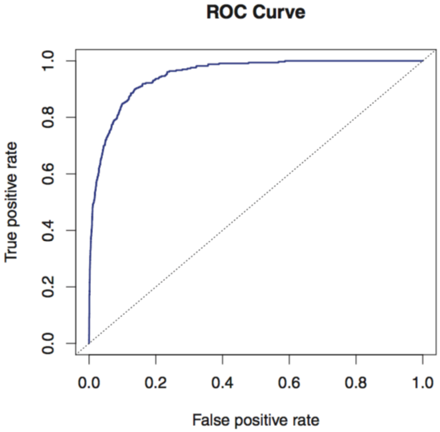
Trade-off between FPR and FNR

We can achieve better balance between FPR and FNR by varying the threshold t:



ROC Curve

The **ROC curve** is a popular graphic for simultaneously displaying FPR and TPR = 1 - FNR for all possible thresholds.



The overall performance of a classifier, summarized over all thresholds, is given by the area under the curve (**AUC**). High AUC is good.

More metrics in the binary classification

		Predicted class		
		− or Null	+ or Non-null	Total
True class	− or Null	True Neg. (TN)	False Pos. (FP)	N
	+ or Non-null	False Neg. (FN)	True Pos. (TP)	P
	Total	N^*	P^*	

Name	Definition	Synonyms
False Pos. rate	FP/N	Type I error, 1−Specificity
True Pos. rate	TP/P	1−Type II error, power, sensitivity, recall
Pos. Pred. value	TP/P^*	Precision, 1−false discovery proportion
Neg. Pred. value	TN/N^*	

The above also defines **sensitivity** and **specificity**.