Lecture 11: Logistic Regression

Iain Styles

22 November 2018

Learning Outcomes

By the end of this lecture you should be able to:

- Understand the concept of *odds*
- Know how to map a continuous problem onto a semi-discrete problem
- Model a classification problems using regression-like methods
- Apply logistic regression to a dataset/

Logistic Regression

- The basis of LDA is to classify a point $\mathbf{x}$ by maximising the posterior distribution $P(\Pi_i|\mathbf{x})$

Logistic Regression

- The basis of LDA is to classify a point $\mathbf{x}$ by maximising the posterior distribution $P(\Pi_i|\mathbf{x})$
- LDA formulates this by explicit construction of the PDF
- In LR, we will not construct the PDF explicitly

Logistic Regression

- The basis of LDA is to classify a point $\mathbf{x}$ by maximising the posterior distribution $P(\Pi_i|\mathbf{x})$
- LDA formulates this by explicit construction of the PDF
- In LR, we will not construct the PDF explicitly
- Instead, we assume it exists, and then model functions of it using regression techniques.

Odds

- The key quantity that we will work with is *odds*

Odds

- The key quantity that we will work with is *odds*
- Familiar to anyone who is interested in horse racing etc

Odds

- The key quantity that we will work with is *odds*
- Familiar to anyone who is interested in horse racing etc
- Odds: ratio of probabilities of two outcomes

Odds

- The key quantity that we will work with is *odds*
- Familiar to anyone who is interested in horse racing etc
- Odds: ratio of probabilities of two outcomes
- Writing $p_i(\mathbf{x}) = P(\Pi_i|\mathbf{x})$ then for **two-class** classification (Π_0 and Π_1), the odds of being in Π_1 are

$$o_1 = p_1/p_0 = p_1/(1 - p_1) \tag{1}$$

Odds

- The key quantity that we will work with is *odds*
- Familiar to anyone who is interested in horse racing etc
- Odds: ratio of probabilities of two outcomes
- Writing $p_i(\mathbf{x}) = P(\Pi_i|\mathbf{x})$ then for **two-class** classification (Π_0 and Π_1), the odds of being in Π_1 are

$$o_1 = p_1/p_0 = p_1/(1 - p_1) \tag{1}$$

- Odds are in range $[0, \infty]$.

Odds

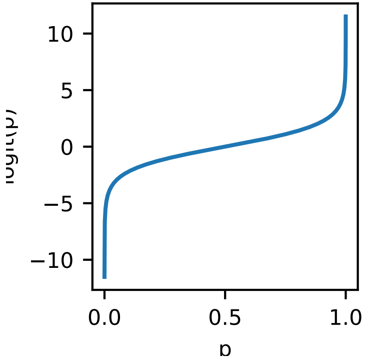
- The key quantity that we will work with is *odds*
- Familiar to anyone who is interested in horse racing etc
- Odds: ratio of probabilities of two outcomes
- Writing $p_i(\mathbf{x}) = P(\Pi_i|\mathbf{x})$ then for **two-class** classification (Π_0 and Π_1), the odds of being in Π_1 are

$$o_1 = p_1/p_0 = p_1/(1 - p_1) \tag{1}$$

- Odds are in range $[0, \infty]$.
- Want to apply a regression model – how?

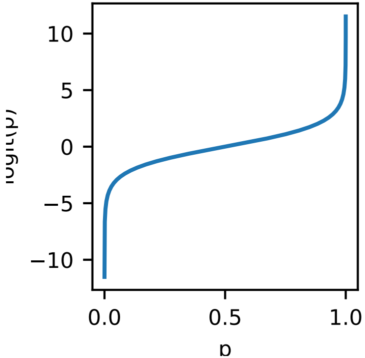
Logit

- Take the log of the odds – log-odds – the *logit*



Logit

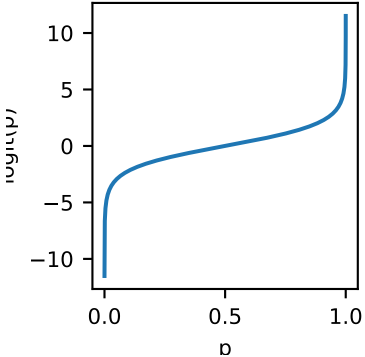
- Take the log of the odds – log-odds – the *logit*



- $\text{logit}(p_1) = \ln \frac{p_1}{1-p_1}$ maps $[0, 1] \mapsto [-infty, \infty]$

Logit

- Take the log of the odds – log-odds – the *logit*



- $\mathrm{logit}(p_1) = \ln \frac{p_1}{1-p_1}$ maps $[0,1] \mapsto [-infty, \infty]$
- Can use regression methods on logit to compute p_1?

Logistic Regression

- Goal: model $p_1(\mathbf{x}) = P(\Pi_1|\mathbf{x})$ using regression

Logistic Regression

- Goal: model $p_1(\mathbf{x}) = P(\Pi_1|\mathbf{x})$ using regression
- logit maps p_1 to $\mathbb{R}$ so can use regression

Logistic Regression

- Goal: model $p_1(\mathbf{x}) = P(\Pi_1|\mathbf{x})$ using regression
- logit maps p_1 to $\mathbb{R}$ so can use regression
- Model $\mathrm{logit}(p_1) = w_0 + w_1 x_1 + \cdots + w_n x_n = \mathbf{w}^{\mathrm{T}}\mathbf{x}$

Logistic Regression

- Goal: model $p_1(\mathbf{x}) = P(\Pi_1|\mathbf{x})$ using regression
- logit maps p_1 to $\mathbb{R}$ so can use regression
- Model $\mathrm{logit}(p_1) = w_0 + w_1 x_1 + \cdots + w_n x_n = \mathbf{w}^{\mathrm{T}}\mathbf{x}$
- Can now re-express $p_1(\mathbf{x})$ as

$$p_1(\mathbf{x}, \mathbf{w}) = \frac{\exp(\mathbf{w}^{\mathrm{T}}\mathbf{x})}{1 + \exp(\mathbf{w}^{\mathrm{T}}\mathbf{x})} \tag{2}$$

Logistic Regression

- Goal: model $p_1(\mathbf{x}) = P(\Pi_1|\mathbf{x})$ using regression
- logit maps p_1 to $\mathbb{R}$ so can use regression
- Model $\mathrm{logit}(p_1) = w_0 + w_1 x_1 + \cdots + w_n x_n = \mathbf{w}^{\mathrm{T}}\mathbf{x}$

logit(p)

- Can now re-express $p_1(\mathbf{x})$ as

$\frac{exp}{1+exp}$ $\frac{e^x}{1+e^x}$ =

$$p_1(\mathbf{x}, \mathbf{w}) = \frac{\exp(\mathbf{w}^{\mathrm{T}}\mathbf{x})}{1 + \exp(\mathbf{w}^{\mathrm{T}}\mathbf{x})} \tag{2}$$

log

- So logit allows us to model the probability directly using linear regression

$(1-p)\,\mathrm{logit}(p) = p$

$\mathrm{logit}(p) = \frac{p}{1-p}$ $\quad p = \frac{\mathrm{logit}(p)}{1+\mathrm{logit}(p)}$

Maxmising the likelihood

- Let us imagine now that we have a set of observations $\{\mathbf{x}_i, y_i\}_{i=1}^{N}$
- Assume independence of *binary* observations $y_i = \{0, 1\}$

Maxmising the likelihood

$logit(p) = \ln \frac{p}{1-p}$

- Let us imagine now that we have a set of observations $\{\mathbf{x}_i, y_i\}_{i=1}^N$
- Assume independence of *binary* observations $y_i = \{0, 1\}$
- The overall likelihood is then

$$\mathcal{L}(\mathbf{w}) = \prod_{i=1}^{N} p_1(\mathbf{x}_i, \mathbf{w})^{y_i} p_0(\mathbf{x}_i, \mathbf{w})^{1-y_i} \tag{3}$$

$$= \prod_{i=1}^{N} p_1(\mathbf{x}, \mathbf{w})^{y_i} (1 - p_1(\mathbf{x}, \mathbf{w}))^{1-y_i} \tag{4}$$

Maximising the likelihood

- The log-likelihood is then

$$\ln \mathcal{L}(\mathbf{w}) = \sum_{i=1}^{N} y_i \ln(p_1(\mathbf{x}, \mathbf{w})) + (1 - y_i) \ln(1 - p_1(\mathbf{x}, \mathbf{w})) \tag{5}$$

$$= \sum_{i=1}^{N} y_i \left[\ln(p_1(\mathbf{x}, \mathbf{w})) - \ln(1 - p_1(\mathbf{x}, \mathbf{w}))\right] + \ln(1 - p_1(\mathbf{x}, \mathbf{w})) \tag{6}$$

logit(p)

$$= \sum_{i=1}^{N} y_i \ln \frac{p_1(\mathbf{x}, \mathbf{w}))}{1 - p_1(\mathbf{x}, \mathbf{w})} + \ln(1 - p_1(\mathbf{x}, \mathbf{w})) \tag{7}$$

$$= \sum_{i=1}^{N} y_i \mathbf{w}^{\mathrm{T}} \mathbf{x} - \ln(1 + \exp(\mathbf{w}^{\mathrm{T}} \mathbf{x})). \tag{8}$$

Maximising the likelihood

- The optimal model weights are then

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \left[\sum_{i=1}^{N} y_i \mathbf{w}^{\mathrm{T}} \mathbf{x} - \ln(1 + \exp(\mathbf{w}^{\mathrm{T}} \mathbf{x})) \right] \qquad (9)$$

Maximising the likelihood

- The optimal model weights are then

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \left[\sum_{i=1}^{N} y_i \mathbf{w}^{\mathrm{T}} \mathbf{x} - \ln(1 + \exp(\mathbf{w}^{\mathrm{T}} \mathbf{x})) \right] \tag{9}$$

- Cannot be solved analytically

Maximising the likelihood

- The optimal model weights are then

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \left[\sum_{i=1}^{N} y_i \mathbf{w}^{\mathrm{T}} \mathbf{x} - \ln(1 + \exp(\mathbf{w}^{\mathrm{T}} \mathbf{x})) \right] \qquad (9)$$

- Cannot be solved analytically
- Need to use optimisation techniques such as IRLS
- A data point $\mathbf{x}$ should be assigned to class 1 if $p_1 > 1 - p_1$, i.e when $\mathrm{logit}(p_1) > 0$.
- The decision rule is therefore

$$\mathbf{w}^{*\mathrm{T}} \mathbf{x} > 0 \quad \rightarrow \quad \mathbf{x} \in \Pi_1$$

Maximising the likelihood

- The optimal model weights are then

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \left[\sum_{i=1}^{N} y_i \mathbf{w}^{\mathrm{T}} \mathbf{x} - \ln(1 + \exp(\mathbf{w}^{\mathrm{T}} \mathbf{x})) \right] \tag{9}$$

- Cannot be solved analytically
- Need to use optimisation techniques such as IRLS
- A data point $\mathbf{x}$ should be assigned to class 1 if $p_1 > 1 - p_1$, i.e when $\mathrm{logit}(p_1) > 0$.

logit $(p_1) > 0$

- The decision rule is therefore

$$\mathbf{w}^{*\mathrm{T}} \mathbf{x} > 0 \quad \rightarrow \quad \mathbf{x} \in \Pi_1$$

- or equivalently assign to the most probable class based on

$$p_1 = \frac{\exp(\mathbf{w}^{*\mathrm{T}} \mathbf{x})}{1 + \exp(\mathbf{w}^{*\mathrm{T}} \mathbf{x})} \quad \text{and} \quad p_0 = 1 - p_1 = \frac{1}{1 + \exp(\mathbf{w}^{*\mathrm{T}} \mathbf{x})} \tag{10}$$

Extension to multiple classes

- Multinomial LR

Extension to multiple classes

- Multinomial LR
- Core idea: "pivot" all classes against a single reference class

one vs all

Extension to multiple classes

- Multinomial LR
- Core idea: "pivot" all classes against a single reference class
- Given M classes, compute $\ln(\frac{p_i}{p_M})$ for all $i \neq M$.

$$\ln \frac{p_1}{p_M} = \mathbf{w}_1^{*\mathrm{T}}\mathbf{x} \tag{11}$$

$$\ln \frac{p_2}{p_M} = \mathbf{w}_2^{*\mathrm{T}}\mathbf{x} \tag{12}$$

$$\ldots \tag{13}$$

$$\ln \frac{p_{M-1}}{p_M} = \mathbf{w}_{M-1}^{*\mathrm{T}}\mathbf{x} \tag{14}$$

$$\tag{15}$$

Extension to multiple classes

- $\ln \frac{p_i}{p_M} = \mathbf{w}_i^{*\mathrm{T}} \mathbf{x}$

Extension to multiple classes

- $\ln \frac{p_i}{p_M} = \mathbf{w}_i^{*\mathrm{T}} \mathbf{x}$
- Exponentiate:

$$p_i = p_M \exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x}) \quad \text{for i=}\{1,2,\ldots,\text{M-1}\} \tag{16}$$

Extension to multiple classes

- $\ln \frac{p_i}{p_M} = \mathbf{w}_i^{*\mathrm{T}} \mathbf{x}$
- Exponentiate:

$$p_i = p_M \exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x}) \quad \text{for i=}\{1,2,\ldots,\text{M-1}\} \tag{16}$$

- p_i must sum to 1

$$\sum_{i=1}^{M} p_i = 1 \quad \rightarrow \quad p_M = 1 - \sum_{i=1}^{M-1} p_M \exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x}) \tag{17}$$

Extension to multiple classes

- $\ln \frac{p_i}{p_M} = \mathbf{w}_i^{*\mathrm{T}} \mathbf{x}$
- Exponentiate:

$$p_i = p_M \exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x}) \quad \text{for i=}\{1,2,\ldots,\text{M-1}\} \tag{16}$$

- p_i must sum to 1

$$\sum_{i=1}^{M} p_i = 1 \quad \rightarrow \quad p_M = 1 - \sum_{i=1}^{M-1} p_M \exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x}) \tag{17}$$

- and therefore

$$p_M = \frac{1}{1 + \sum_{i=1}^{M-1} \exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x})}. \tag{18}$$

Extension to multiple classes

- $\ln \frac{p_i}{p_M} = \mathbf{w}_i^{*\mathrm{T}} \mathbf{x}$
- Exponentiate:

$$p_i = p_M \exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x}) \quad \text{for i=}\{1,2,\ldots,\text{M-1}\} \tag{16}$$

- p_i must sum to 1

$$\sum_{i=1}^{M} p_i = 1 \quad \rightarrow \quad p_M = 1 - \sum_{i=1}^{M-1} p_M \exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x}) \tag{17}$$

- and therefore

$$p_M = \frac{1}{1 + \sum_{i=1}^{M-1} \exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x})}. \tag{18}$$

- Finally, we substitute to obtain

$$p_i = p_M \exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x}) = \frac{\exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x})}{1 + \sum_{i=1}^{M-1} \exp(\mathbf{w}_i^{*\mathrm{T}} \mathbf{x})} \tag{19}$$

Extension to multiple classes

- Multiple binary LRs of each class against the pivot class Π_M

Extension to multiple classes

- Multiple binary LRs of each class against the pivot class Π_M
- Find the parameters $\mathbf{w}_i^*$ for each class i

Extension to multiple classes

- Multiple binary LRs of each class against the pivot class Π_M
- Find the parameters $\mathbf{w}_i^*$ for each class i
- Assign $\mathbf{x}$ to the class with the highest probability p_i.

Extension to multiple classes

- Multiple binary LRs of each class against the pivot class Π_M
- Find the parameters $\mathbf{w}_i^*$ for each class i
- Assign $\mathbf{x}$ to the class with the highest probability p_i.
- Let's try it – but this time with a library
- `https://colab.research.google.com/drive/1-vpNgx3PtdyRGv1pC0PYrR-X-vdf8jJT`

Summary

class . sklearn solver : , regularisation/penalty : ,

- LR applies regression methods to classification
- The statistical assumptions are much more relaxed in LR as compared to LDA.
- No assumption that the likelihoods are are multivariate Gaussian.
- There is no assumption that the class distributions have the same covariance.
- LR more robust to non-normality than LDA.
- LR is much less efficient than LDA for large sample sizes.
- LR can require larger dataset sizes to work well.

Summary

- LR applies regression methods to classification
- The statistical assumptions are much more relaxed in LR as compared to LDA.
- No assumption that the likelihoods are are multivariate Gaussian.
- There is no assumption that the class distributions have the same covariance.
- LR more robust to non-normality than LDA.
- LR is much less efficient than LDA for large sample sizes.
- LR can require larger dataset sizes to work well.