Lecture 15: Logistic Regression

William Webber (william@williamwebber.com)

COMP90042, 2014, Semester 1, Lecture 15

What we'll learn in this lecture

- Model-based regression and classification
- Logistic regression as a probabilistic classifier

Model-based regression and classification

- NB instance of model-based probabilistic classification
- In more general form, expressible as:

$$P(c|\vec{x}) = f(\vec{x}, \vec{\beta}) \tag{1}$$

where:

$f()$ is some function
$\vec{x}$ vector of feature scores, $\{x_1, \ldots, x_n\}$
$\vec{\beta}$ vector of feature weights, $\{\beta_0, \beta_1, \ldots, \beta_n\}$
β_0 is for intercept

- More specifically:

$$P(c|\vec{x}) = f(\{\beta_0, \beta_1 x_1, \ldots, \beta_n x_n\}) \tag{2}$$

- Idea is then to learn "best" $\vec{\beta}$

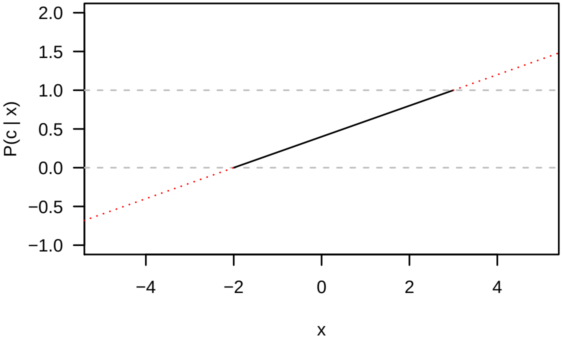
Linear model

$$P(c|\vec{x})) = f(\vec{x}, \vec{\beta}) = \beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n \quad (3)$$

- Might try simple linear model
- Fitted with ordinary least squares ($\approx$ straight line [hyperplane] of best fit)

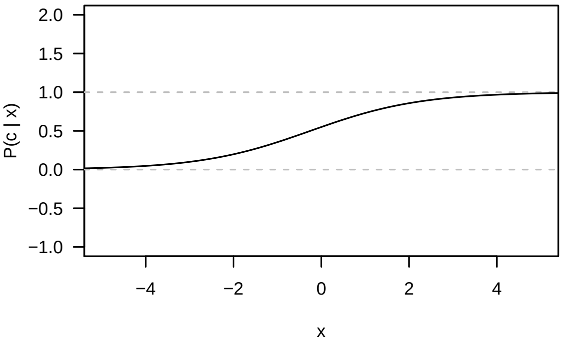
Linear model

$$P(c|\vec{x}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n \quad (4)$$



- But probabilities bound between 0 and 1
- Meaning of probabilities outside range unclear
- Artificial to bound $\vec{\beta}$ to this range

Sigmoid model



- What we want is response variable (y, $P(c|\vec{x})$) bounded between $[0, 1]$
- But predictor variable, x_i, unbounded (at least by model)
- General shape of such a function is a sigmoid or "S-shaped curve"

Log-linear models

$$P(c|\vec{x}) = \beta_0 \cdot \beta_1^{x_1} \cdot \ldots \cdot \beta_n^{x_n} \quad (5)$$

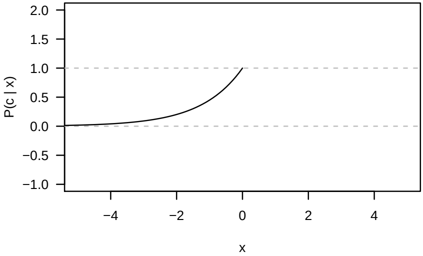
$$\log P(c|\vec{x}) = \log \beta_0 + x_1 \log \beta_1 + \ldots + x_n \log \beta_n \quad (6)$$

- Natural (see NB) to express total probability
- as (weighted) product of individual probabilities
- exponentiated by frequency of events
- Taking log of this gives log-linear model
- Directly fit $\log \beta_i$, so can write as:

$$\log P(c|\vec{x}) = \beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n \quad (7)$$

$\log(P) = \beta x$

$$\begin{aligned} \log(P) &= \beta x \\ P &= e^{\beta x} \end{aligned}$$



- But curve has unbalanced shape:
  - Fine granularity of response as $P \to 0$
  - Coarse response as $P \to 1$

Balanced in P

- Want behaviour that is same for high P and low P
- This is provided by *log odds* or *logit*:

$$\mathrm{logit}(P) = \log \frac{P}{1-P} \tag{8}$$

$$\mathrm{logit}(1-P) = -\mathrm{logit}(P) \tag{9}$$

Logistic regression

Putting this together, we get:

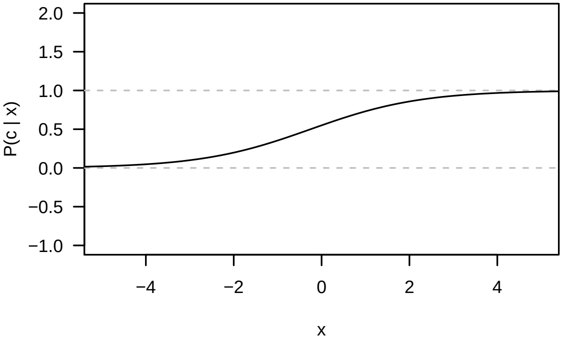
$$\text{logit}\, P(c|\vec{x}) = \log \frac{P(c|\vec{x})}{1 - P(c|\vec{x})} = \beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n \quad (10)$$

$$P(c|\vec{x}) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n)}} \quad (11)$$

- Expression on rhs of (11) known as logistic function
- So this is called logistic regression

Logistic function

$$y = \frac{1}{1 + e^{-(\beta_0 + \beta x)}} \tag{12}$$



- And, happily, the logistic function sigmoid
- (Indeed, is archetypal sigmoid function)

Fitting the model

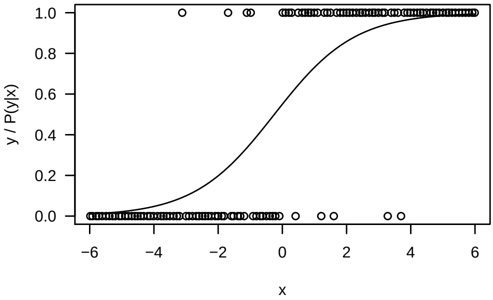
Doc	Terms ($\mathbf{X}_d$)						Class (y)
1	X_{11}	X_{12}	$\cdots$	X_{1t}	$\cdots$	X_{1n}	1
2	X_{21}	X_{22}	$\cdots$	X_{2t}	$\cdots$	X_{2n}	0
$\vdots$							$\vdots$
d	X_{d1}	X_{d2}	$\cdots$	X_{dt}	$\cdots$	X_{dn}	0
$\vdots$							$\vdots$
m	X_{m1}	X_{m2}	$\cdots$	X_{mt}	$\cdots$	X_{mn}	1

- Training data feature vectors **X** with labels $\vec{y}$
- Labels for binary classification: member, or non-member
- Have to determine vector $\vec{\beta}$ such that:

$$P(y_d|matX_d) = \left(1 + \exp(-(\beta_0 + \sum_i \beta_i x_i))\right)^{-1} \quad (13)$$

  "best fits" data
- Free to use any values for X_{dt}
  - Length-normalized TF*IDF one choice

Data and model



- The data being fitted are binary
- The fitting value is a probability, $P(y_d = c|\mathbf{X}_d)$
- We're fitting a curve of Bernoulli (one-event binomial) vars
- . . . that best fits the observed data

Maximum likelihood estimation

For weights $\vec{\beta}$, the *likelihood* of the data $\mathbf{X}$ and labels $\vec{y}$ given that model is:

$$L(\vec{\beta}) = \prod_{l:y_l=1} P(\mathbf{X}_l) \prod_{l:y_l=0} [1 - P(\mathbf{X}_l)] \quad (14)$$

For logistic model:

$$P(\mathbf{X}_l) = \frac{1}{1 + \mathrm{e}^{-(\beta_0 + \sum_i \beta_i X_{li})}} \quad (15)$$

- We have to find $\vec{\beta}$ that maximizes (14)
- This is done by a computer using iterative methods

Logistic regression in practice

	Collection		
Classifier	hotmail	trec-2005	trec-2006
NB	0.2479	0.8196	0.8017
NB-IR	0.5561	0.9207	0.9521
Log. Reg	0.4877	0.9461	0.9384
SVM	0.4830	0.9477	0.9754

Table : Normalized AUC on spam filtering; from Kotz and Yih, "Raising the Baseline for High-Precision Text Classifiers", KDD 2007. NB-IR is NB with IR features (length-normalized TF*IDF)

- Logistic regression for text classification generally "almost, but not quite" as good as SVM
- (Note, on this task, NB with LN-TF*IDF does well
- ...and see paper for variants that do even better)
- On our GCAT 1000/1000 data, with length-normalized TF*IDF features, LR got accuracy 93%, F1 88%

Interpreting logistic regression: weights

- β_i for term i gives importance of that term in model
  - (but interpretation subject to term dependencies)
- For topic GCAT (Govt/Social), highest-weight terms were:

Positive		Negative	
Term	Weight	Term	Weight
sunday	0.869	shar	-0.951
socc	0.643	newsroom	-0.926
minist	0.635	trad	-0.669
eu	0.629	stock	-0.593
saturday	0.599	compan	-0.580

Interpreting logistic regression: probabilites

- Logistic regression directly gives reasonable probabilities
- (given constraint of model)
- For GCAT 1000/1000

$P(c)$		
$\geq$	$<$	% positive
0.00	0.05	2.4%
0.05	0.10	14.8%
0.10	0.30	26.9%
0.30	0.50	48.9%
0.50	0.70	74.2%
0.70	0.90	89.7%
0.90	0.95	93.8%
0.95	1.00	99.2%

Looking back and forward

Back

- Model as $P(c|\vec{x}) = f(\beta_1 x_1, \cdots, \beta_n x_n)$ where
  - x_i is feature *score* (differs for each document)
  - β_i is feature *weight* (common across topics)
- Learn weights that best "fit" training data
- Free to use whatever values for x_1 (e.g. normalized TF*IDF)
- But probabilities bound between $[0, 1]$

Looking back and forward

Back

- Sigmoid function maps unbounded feature scores to bounded probabilities
- Log odds gives even treatment to high, low probabilities
- Logistic model ties these together
- Learn weights $\vec{\beta}$ using maximum likelihood
- Effectiveness "almost, but not quite" as good as SVM
- But gives us feature weights, reasonable probabilities

Looking back and forward

Forward

- Next lecture: advanced topics in classification
- e.g. active learning
- Later: topic modelling

Further reading

- Klienbaum and Klein, "Logistic Regression", 3rd edn (2010) (detailed, gradual introduction to logistic regression)
- Hastie, Tibshirani, and Friedman, "The ELements of Statistical Learning" (2001) (briefer, more technical description)