Logistic Regression

Aarti Singh

Machine Learning 10-315
Sept 21, 2020

Discriminative Classifiers

Bayes Classifier:

$$f^*(x) = \arg\max_{Y=y} P(Y = y|X = x)$$
$$= \arg\max_{Y=y} P(X = x|Y = y)P(Y = y)$$

Why not learn P(Y|X) directly? Or better yet, why not learn the decision boundary directly?

- Assume some functional form for P(Y|X) or for the decision boundary
- Estimate parameters of functional form directly from training data

Today we will see one such classifier – **Logistic Regression**

Logistic Regression

Not really regression

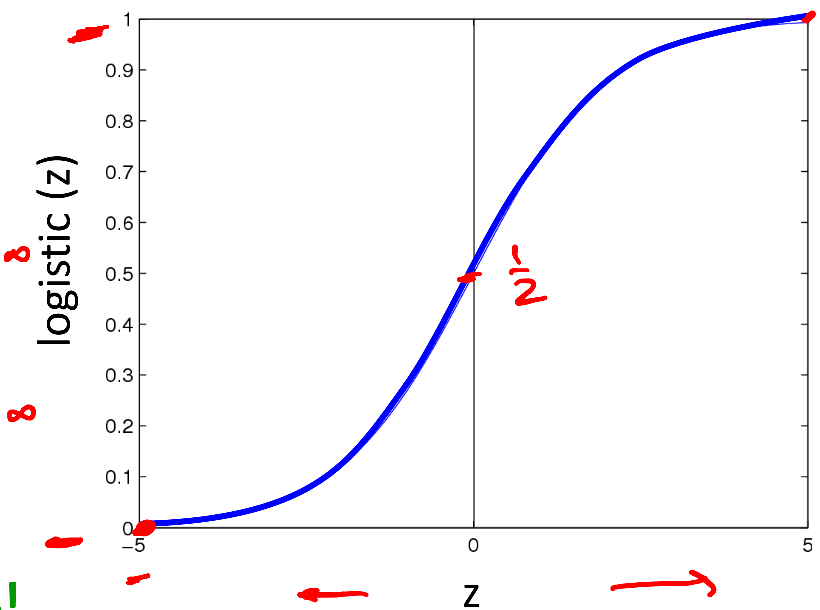
Binary classification

Assumes the following functional form for P(Y|X):

$$P(Y=0|X) = \frac{1}{1+\exp(w_0 + \sum_i w_i X_i)}$$

bias ← w_0; features ← X_i; weights of features ← w_i

Logistic function applied to a linear function of the data



Logistic function (or Sigmoid):

$$\frac{1}{1+exp(-z)} = \begin{cases} 0 & z \to -\infty \\ \frac{1}{2} & z = 0 \\ 1 & z \to \infty \end{cases}$$

Features can be discrete or continuous!

Logistic Regression is a Linear Classifier!

Assumes the following functional form for P(Y|X):

$$P(Y=0|X) = \frac{1}{1+\exp(w_0+\sum_i w_i X_i)} =$$

$X_0 = 1$

$$P(Y=1|X) = 1 - P(Y=0|X) = 1 - \frac{1}{1+\exp(\sum_i w_i X_i)}$$

$$= \frac{\exp(\sum_i w_i X_i)}{1+\exp(\sum_i w_i X_i)} =$$

$$= \frac{1}{1+\exp(-\sum_i w_i X_i)} =$$

Logistic Regression is a Linear Classifier!

Assumes the following functional form for P(Y|X):

$$P(Y=0|X) = \frac{1}{1+\exp(w_0 + \sum_i w_i X_i)}$$

$$w_0 + \sum_i w_i X_i = 0$$

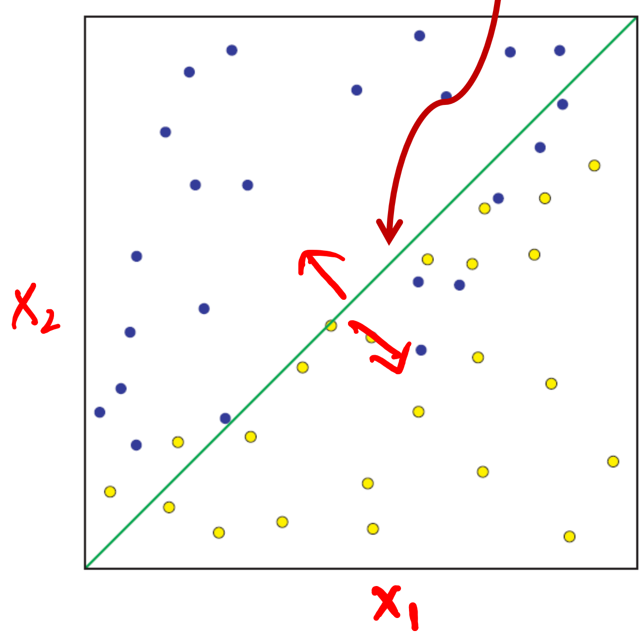
Decision boundary: Note - Labels are 0,1

$$P(Y=0|X) \underset{1}{\overset{0}{\gtrless}} P(Y=1|X)$$

X :

$$w_0 + \sum_i w_i X_i \underset{0}{\overset{1}{\gtrless}} 0$$

(Linear Decision Boundary)



Logistic Regression is a Linear Classifier!

Assumes the following functional form for P(Y|X):

$$P(Y=0|X) = \frac{1}{1+\exp(w_0 + \sum_i w_i X_i)}$$

– logistic

$$\Rightarrow P(Y=1|X) = \frac{\exp(w_0 + \sum_i w_i X_i)}{1+\exp(w_0 + \sum_i w_i X_i)}$$

$$\Rightarrow \frac{P(Y=1|X)}{P(Y=0|X)} = \exp(w_0 + \sum_i w_i X_i) \underset{0}{\overset{1}{\gtrless}} 1$$

$$\Rightarrow w_0 + \sum_i w_i X_i \underset{0}{\overset{1}{\gtrless}} 0$$

– linear

Training Logistic Regression

How to learn the parameters w_0, w_1, … w_d? (d features)

Training Data $\{(X^{(j)}, Y^{(j)})\}_{j=1}^{n}$ $\qquad X^{(j)} = (X_1^{(j)}, \ldots, X_d^{(j)})$

Maximum Likelihood Estimates

iid

$$\widehat{\mathbf{w}}_{MLE} = \arg\max_{\mathbf{w}} \prod_{j=1}^{n} P(X^{(j)}, Y^{(j)} \mid \mathbf{w})$$

$\equiv P(D|\theta)$

But there is a problem …

p(y|x) only

Don’t have a model for P(X) or P(X|Y) – only for P(Y|X)

Training Logistic Regression

How to learn the parameters w_0, w_1, … w_d? (d features)

Training Data $\{(X^{(j)}, Y^{(j)})\}_{j=1}^{n}$ $X^{(j)} = (X_1^{(j)}, \ldots, X_d^{(j)})$

Maximum (Conditional) Likelihood Estimates

$$\widehat{\mathbf{w}}_{MCLE} = \arg\max_{\mathbf{w}} \prod_{j=1}^{n} P(Y^{(j)} \mid X^{(j)}, \mathbf{w})$$

Discriminative philosophy – Don't waste effort learning P(X), focus on P(Y|X) – that's all that matters for classification!

Expressing Conditional log Likelihood

$P(Y=y|X,w) = \frac{exp(y \sum_i w_i X_i)}{1+exp(\sum_i w_i X_i)}$ $X_0 = 1$

$$P(Y=0|\mathbf{X},\mathbf{w}) = \frac{1}{1+exp(w_0+\sum_i w_i X_i)}$$

$$P(Y=1|\mathbf{X},\mathbf{w}) = \frac{exp(w_0+\sum_i w_i X_i)}{1+exp(w_0+\sum_i w_i X_i)} \leftarrow$$

log likelihood

$$l(\mathbf{w}) \equiv \ln \prod_j P(y^j|\mathbf{x}^j,\mathbf{w})$$

$\log_e = \ln$ $\log(ab) = \log a + \log b$

$\log \frac{a}{b} = \log a - \log b$

$$= \sum_j \ln P(y^j|x^j,w) = \sum_j \ln\left(\frac{exp(y^j \sum_i w_i x_i^j)}{1+exp(\sum_i w_i x_i^j)}\right)$$

$$= \sum_j \left[(y^j \sum_i w_i x_i^j) - \ln(1+exp(\sum_i w_i x_i^j))\right]$$

Expressing Conditional log Likelihood

$$P(Y=0|\mathbf{X},\mathbf{w}) = \frac{1}{1+exp(w_0+\sum_i w_i X_i)}$$

$$P(Y=1|\mathbf{X},\mathbf{w}) = \frac{exp(w_0+\sum_i w_i X_i)}{1+exp(w_0+\sum_i w_i X_i)}$$

$$\begin{aligned} l(\mathbf{w}) &\equiv \ln \prod_j P(y^j|\mathbf{x}^j,\mathbf{w}) \\ &= \sum_j \left[y^j (w_0 + \sum_i^d w_i x_i^j) - \ln(1 + exp(w_0 + \sum_i^d w_i x_i^j)) \right] \end{aligned}$$

Good news: *l*(**w**) is concave function of **w** !

Bad news: no closed-form solution to maximize *l*(**w**)

Good news: can use iterative optimization methods (gradient ascent)

That’s M(C)LE. How about M(C)AP?

$$p(\mathbf{w} \mid Y, \mathbf{X}) \;\propto\; P(Y \mid \mathbf{X}, \mathbf{w})p(\mathbf{w})$$

- Define priors on **w**
  - Common assumption: Normal distribution, zero mean, identity covariance
  - “Pushes” parameters towards zero

$$p(\mathbf{w}) = \prod_i \frac{1}{\kappa\sqrt{2\pi}} \; e^{\frac{-w_i^2}{2\kappa^2}}$$

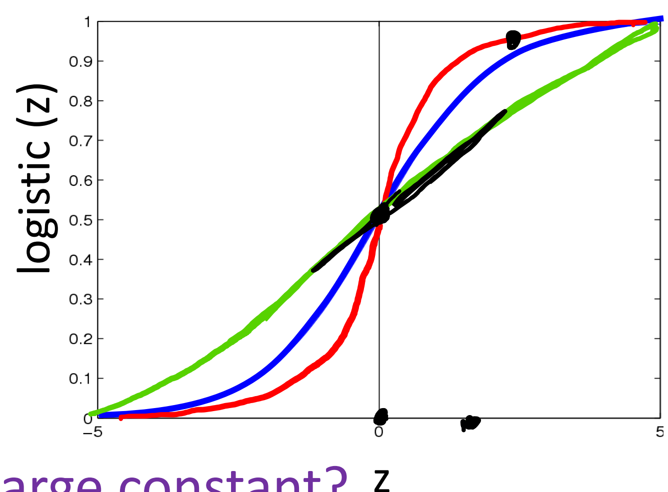
Zero-mean Gaussian prior

$$\frac{1}{1+exp(-0.01z)}$$

Logistic function (or Sigmoid):

$$\frac{1}{1+exp(-z)}$$

$$\frac{1}{1+exp(-100z)}$$



➢ What happens if we scale z by a large constant?

That's M(C)LE. How about M(C)AP?

$$p(\mathbf{w} \mid Y, \mathbf{X}) \propto P(Y \mid \mathbf{X}, \mathbf{w}) p(\mathbf{w})$$

$$p(\mathbf{w}) = \prod_{i} \frac{1}{\kappa\sqrt{2\pi}} \, e^{\frac{-w_i^2}{2\kappa^2}}$$

Zero-mean Gaussian prior

- M(C)AP estimate

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \ln \left[p(\mathbf{w}) \prod_{j=1}^{n} P(y^j \mid \mathbf{x}^j, \mathbf{w}) \right]$$

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \sum_{j=1}^{n} \ln P(y^j \mid \mathbf{x}^j, \mathbf{w}) - \sum_{i=1}^{d} \frac{w_i^2}{2\kappa^2}$$

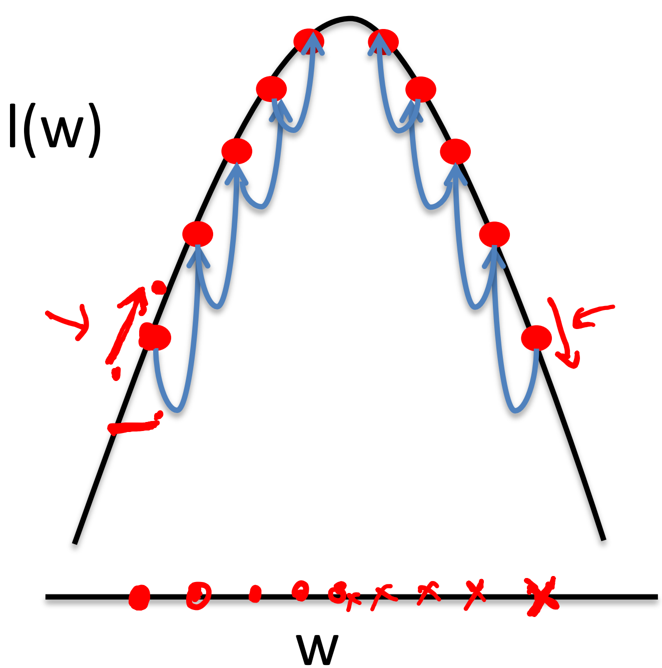
Penalizes large weights

Still concave objective!

Iteratively optimizing concave function

- Conditional likelihood for Logistic Regression is concave
- Maximum of a concave function can be reached by

Gradient Ascent Algorithm



Initialize: Pick **w** at random

Gradient:

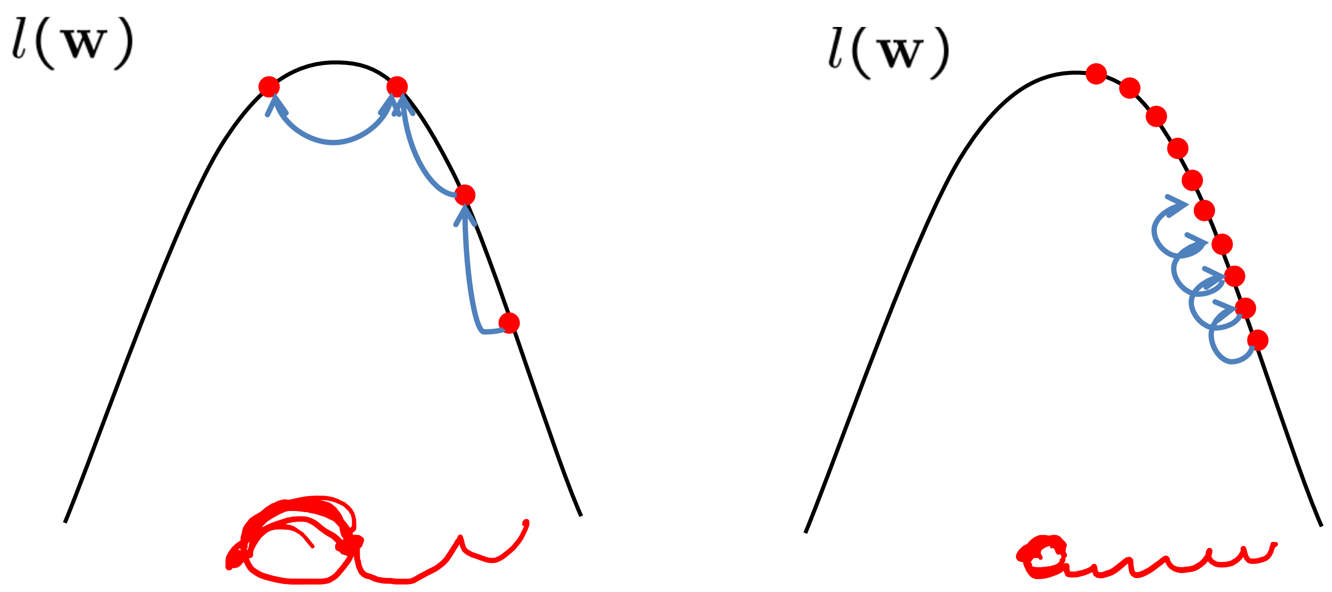
$$\nabla_{\mathbf{w}} l(\mathbf{w}) = \left[\frac{\partial l(\mathbf{w})}{\partial w_0}, \ldots, \frac{\partial l(\mathbf{w})}{\partial w_d}\right]'$$

Update rule: Learning rate, η>0

$$\Delta \mathbf{w} = \eta \nabla_{\mathbf{w}} l(\mathbf{w})$$

$$w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \left.\frac{\partial l(\mathbf{w})}{\partial w_i}\right|_t$$

Effect of step-size η



Large η => Fast convergence but larger residual error
Also possible oscillations

Small η => Slow convergence but small residual error