regression $\longrightarrow$ continuous outcomes

classification $\longrightarrow$ binary outcomes

$\underline{\underline{Y_i = 1}}$ or $\underline{\underline{Y_i = 0}}$

design matrix $X \longrightarrow$ set of predictors or covariates

$P(Y_i = 1 \mid X_i)$

talk about hard classification

$\hookrightarrow \hat{Y}_i = 1$ or $\hat{Y}_i = 0$

$Y_i = 1$ is 10%

$\Longrightarrow$ evaluate predictions probabilistically using Brier score

$\hat{p}_i = P(Y_i = 1 \mid X_i)$

size n

Brier score $\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{p}_i)^2$

compare this to the Brier score of naive probabilistic

$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - \bar{Y})^2$$

probabilistic R^2

$$1 - \frac{\text{Brier Score}}{\text{Naive Brier Score}}$$

analogy: regular R^2

$$1 - \frac{\frac{1}{n}\sum(Y_i - \hat{Y}_i)^2}{\frac{1}{n}\sum(Y_i - \bar{Y})^2}$$

<u>hard predictions</u>

$\hat{Y}_i = 1$ or 0 . $\hat{Y}_i = 1$ if $\hat{p}_i > k$.

		True Values T	True Values F
prediction	T	True positives	False positive
	F	False negative	True negatives

upsell:

1. True positive: +9.99
2. False negative: 0
3. <u>False positive</u>: -10
4. True negative: 0

<u>Logistic Regression</u>

$$X = \begin{bmatrix} \vec{x}_1 \\ \vdots \\ \vec{x}_n \end{bmatrix} \qquad Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad y_i = 0 \text{ or } 1$$

$$p_i = P(y_i = 1 \mid \vec{x}_i)$$

observed y for person

predictors for unit i

probabilities have to be in $[0,1]$

and lines will always

diverge unless slopes

are all 0

$$\log\left(\frac{p_i}{1-p_i}\right) = \vec{X}_i^T \beta$$

$$= \sum_{j=1}^{d} X_{ij} \cdot \beta_j$$

$$\frac{p_i}{1-p_i} = \exp(\vec{X}_i^T \beta)$$

$$p_i = (1-p_i)\exp(\vec{X}_i^T \beta)$$

$$\therefore \quad p_i = \frac{\exp(X_i^T \beta)}{1+\exp(X_i^T \beta)} \qquad 1-p_i = \frac{1}{1+\exp(X_i^T\beta)}$$

$$L(\beta \mid Y, X) \doteq \prod_{i=1}^{n} p_i(\beta)^{Y_i} (1-p_i(\beta))^{1-Y_i}$$

$$\ell(Y \mid X) = \sum \left(Y_i \log(p_i) + (1-Y_i)\log(1-p_i) \right)$$

$$= \sum_{i=1}^{n} \left(Y_i \left(X_i^T\beta - \log(1+\exp(X_i^T\beta)) \right) \right.$$

$$\left. - (1-Y_i)\log(1+\exp(X_i^T\beta)) \right)$$

$$= \sum_{i=1}^{n} Y_i X_i^T \beta - \log(1 + \exp(X_i^T \beta))$$

C

$\Rightarrow$ no closed-form solution to this

but I can the derivative and

use a Newton-Raphson solver

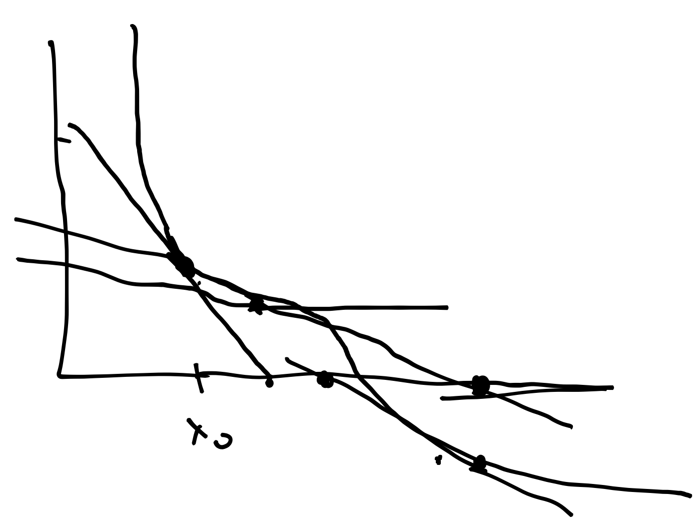
to find β's that gives zeros

to $\ell(\beta | X, Y)$

Newton-Raphson algorithm (1-d)

$f(x) \rightarrow$ want to find a roots of

this function



$$f(x) \approx f(x_0) + f'(x_0)(x - x_0)$$

$$0 = f(x_0) + f'(x_0)(x_1 - x_0)$$

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

$$\vdots$$

$$x_n = x_{n-1} - \frac{f(x_{n-1})}{f'(x_{n-1})}$$

$\Rightarrow$ terminate this when $|f(x_n)| < \varepsilon$

$\varepsilon = 1 \times 10^{-6}$

or $n > 1000$

minimize f (convex function)

maximize f (concave)

$$x_{n+1} = x_n - \frac{f'(x_n)}{f''(x_n)} \quad ;$$

$$\vec{x}_{n+1} = \vec{x}_n - \left(\nabla^2 f(x_n)\right)^{-1} \cdot \nabla f(x_n)$$

$\hookrightarrow$ this gives a recursion that will converge to min/max convex/concave

For machine learning purposes

we have equivalent versions of

LASSO and ridge regression

$$\max_{\beta} \ l(\beta \mid X, Y) - \lambda \sum_{i=1}^{d} \beta_i^2$$

$\hookrightarrow$ ridge equivalent

lasso equival

$$\lambda \sum_{i=1}^{d} |\beta_i|$$

$p_i \qquad i = 1, \ldots, n$

$\min p_i = .01 \qquad$ 10 bins

$\max p_i = .9$

lengths $\dfrac{.9 - .01}{10} = .089$

.03

.04

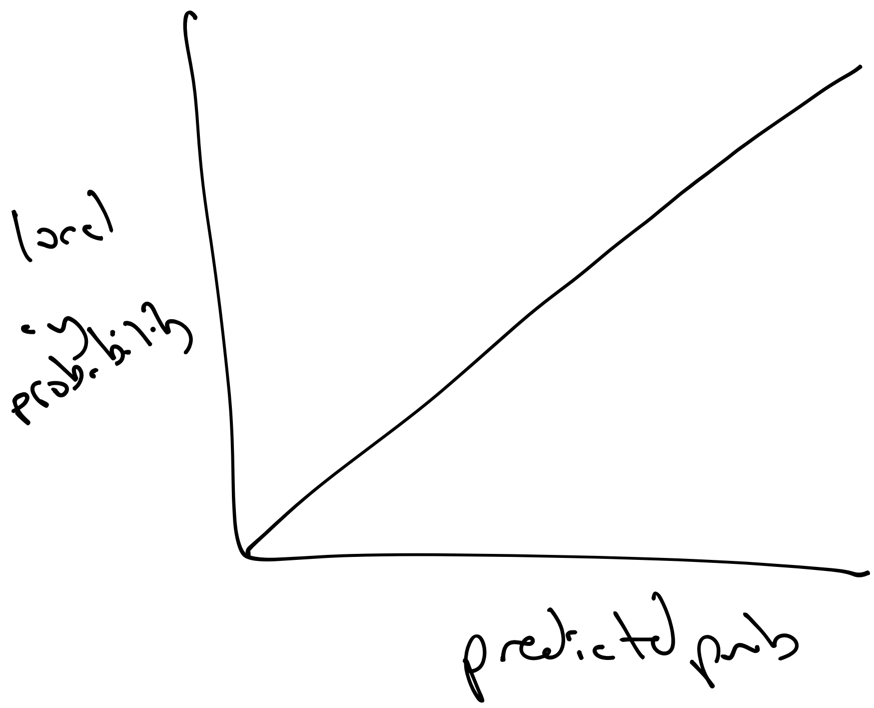
bin 1: $[.01, .09)$

bin 2: $[.09, .179)$ $\rightarrow$ calculate the avg of $\hat{p}_i$'s

calculate Y_i's

$\vdots$

bin 10: $[\quad .9]$


predicted prob