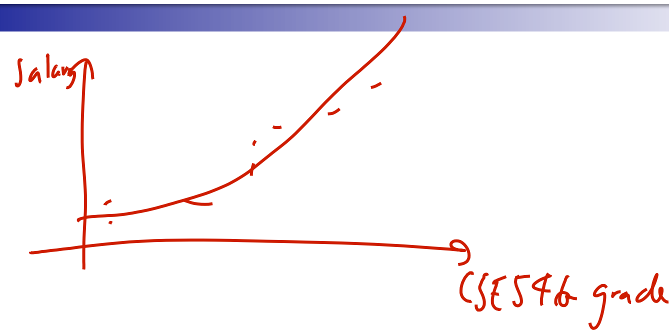
Classification
Logistic Regression

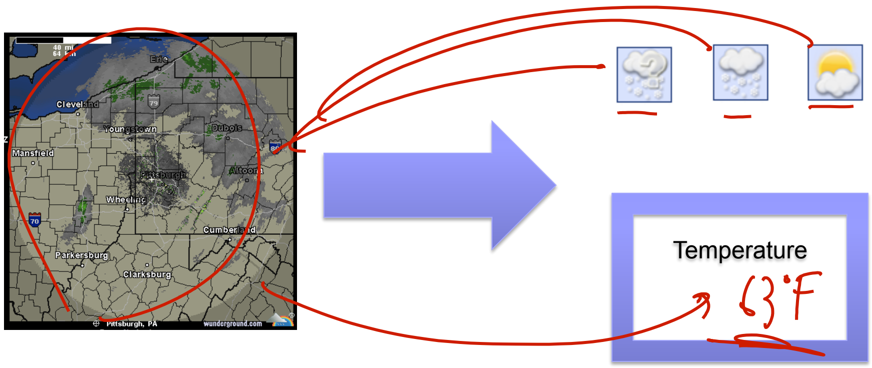
Machine Learning – CSE546
Carlos Guestrin
University of Washington

October 9, 2014



THUS FAR, REGRESSION: PREDICT A CONTINUOUS VALUE GIVEN SOME INPUTS

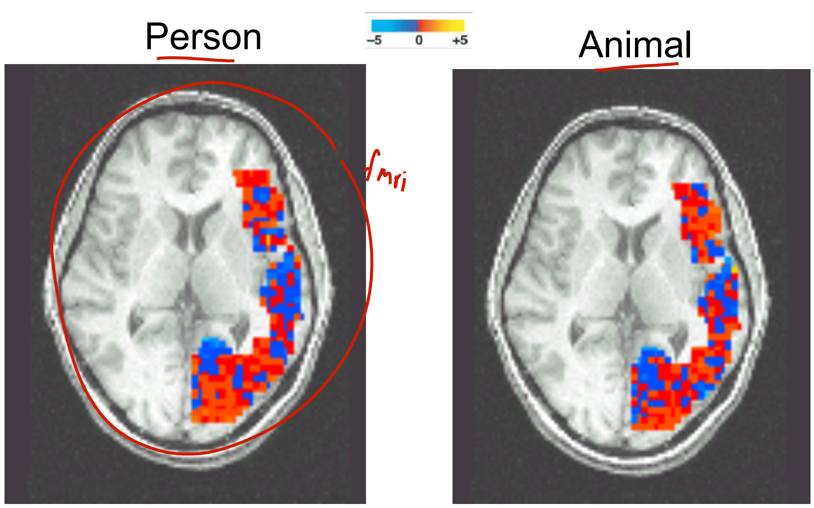
Weather prediction revisted



Reading Your Brain, Simple Example

[Mitchell et al.]

Pairwise classification accuracy: 85%



Classification

$X = \{GPA, ML\ Grade, \ldots\}$

In reg: $Y = \text{Salary}$

in classification: $Y = \{\text{hired}, \text{not hired}\}$

- **Learn**: h:**X** $\mapsto$ Y
  - **X** – features
  - Y – target classes
- Conditional probability: P(Y|**X**)

  $P(Y = \text{hired} \mid X = (GPA: 3.6,\ ML\ Grade: 3.9))$
- Suppose you know P(Y|**X**) exactly, how should you classify?
  - Bayes optimal classifier:

    $\hat{y} = \arg\max_y P(Y = y \mid X = x)$

  $P(\text{hired} \mid 3.6, 3.9) = 0.8$

  $P(y = \text{not hired} \mid 3.6, 3.9) = 0.2$

  $\Rightarrow \hat{y} = \text{hired}$!!
- **How do we estimate P(Y|X)?**

Link Functions

$Y \to [0,1]$

$X = (X_1, \ldots X_j \ldots)$

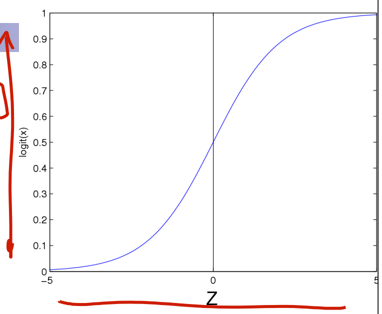
- Estimating P(Y|**X**): Why not use standard linear regression?

  $\underset{\in [0,1]}{P(Y|X)} = \underbrace{w_0 + \sum_i w_i x_i}_{\mathbb{R} \in (-\infty, +\infty)}$
- Combing regression and probability?
  - Need a mapping from real $(-\infty, +\infty)$ values to [0,1]
  - A link function! $g: \mathbb{R} \to [0,1]$

  Many options for g, but today a simple one

Logistic Regression

Logistic function (or Sigmoid): $\frac{1}{1+exp(-z)}$



- Learn P(Y|**X**) directly
  - Assume a particular functional form for link function
  - Sigmoid applied to a linear function of the input features: [0,1]

$$P(Y=0|X,W) = \frac{1}{1+exp(\underbrace{w_0+\sum_i w_i X_i}_{\mathbb{R}})}$$

[0,1]

$\mathbb{R}$

$$P(Y=1|X,W) = 1 - P(Y=0|X,W) = \frac{e^{w_0+\sum_i w_i X_i}}{1+e^{w_0+\sum_i w_i X_i}} \leftarrow h_i(x)$$

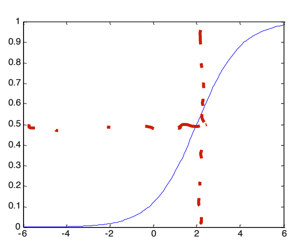
X_i indicators: $X_i = \mathbb{1}(\text{country} = \text{USA})$

Features can be discrete or continuous!

Understanding the sigmoid

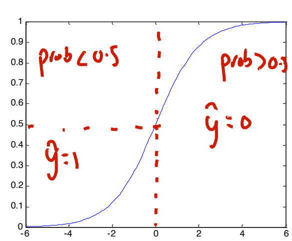
$$g(w_0 + \sum_i w_i x_i) = \frac{1}{1+e^{w_0+\sum_i w_i x_i}}$$

w_0=-2, w_1=-1



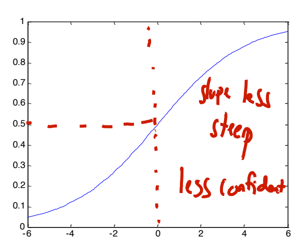
$w_0+\sum_i w_i x_i = 2$

w_0=0, w_1=-1



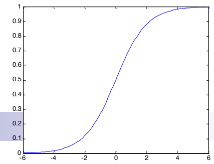
$w_0 + \sum_i w_i x_i$

w_0=0, w_1=-0.5



Logistic Regression – a Linear classifier

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



$$g(w_0 + \sum_i w_i x_i) = \frac{1}{1+e^{w_0 + \sum_i w_i x_i}}$$

$w_0 + \sum_i w_i x_i = 0$

$w_0 + \sum_i w_i x_i > 0$
$\Rightarrow g(w_0 + \sum_i w_i x_i) < 0.5$
$\Rightarrow P(Y=0 \mid X, w) < 0.5$
$\Rightarrow \hat{y} = 1$

$w_0 + \sum_i w_i x_i < 0$
$\Rightarrow g(w_0 + \sum_i w_i x_i) > 0.5$
$\Rightarrow P(Y=0 \mid X, w) > 0.5$
$\Rightarrow \hat{y} = 0$

+ + + − − − −

Very convenient!

$f > 1 \Rightarrow \ln f > 0$

$$P(Y=0 \mid X = <X_1, ... X_n>) = \frac{1}{1+exp(w_0 + \sum_i w_i X_i)}$$

implies

$$P(Y=1 \mid X = <X_1, ... X_n>) = \frac{exp(w_0 + \sum_i w_i X_i)}{1+exp(w_0 + \sum_i w_i X_i)}$$

implies

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

$\Rightarrow \hat{y} = 1$

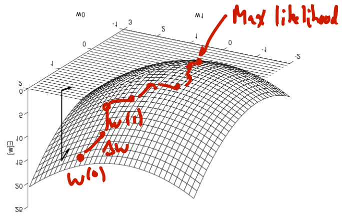
linear classification rule!

implies log odds ratio

$$\hat{y}=1 \Leftarrow 0 < \ln \frac{P(Y=1 \mid X)}{P(Y=0 \mid X)} = w_0 + \sum_i w_i X_i > 0$$

Optimizing concave function – Gradient ascent

- Conditional likelihood for Logistic Regression is concave. Find optimum with gradient ascent



Gradient: $\nabla_{\mathbf{w}} l(\mathbf{w}) = [\frac{\partial l(\mathbf{w})}{\partial w_0}, \ldots, \frac{\partial l(\mathbf{w})}{\partial w_n}]'$

Step size, η>0

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

$$w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \frac{\partial l(\mathbf{w})}{\partial w_i}$$

choice of η?
from theory
$\eta = \frac{\alpha}{\sqrt{t}}$
or
$\eta = \frac{\alpha}{t}$
or
for HW → $\eta = \alpha$

- Gradient ascent is simplest of optimization approaches
  - e.g., Conjugate gradient ascent can be much better
    Newton, LBFGS, ...

$\arg\max f = \arg\max \ln f$ $\ln \prod = \sum \ln$

Loss function: Conditional Likelihood

- Have a bunch of iid data of the form: $(x^j, y^j)_{1:N} = D = (D_X, D_Y)$

  X → ML Grade Y → hired?

- Discriminative (logistic regression) loss function:
  Conditional Data Likelihood $P(Y|X)$

$$\arg\max_w P(D_Y | D_X, w) \overset{iid}{=} \arg\max_w \prod_{j=1}^{N} P(y^j | x^j, w)$$

$$= \arg\max_w \ln \prod_j P(y^j | x^j, w)$$

$$\ln P(\mathcal{D}_Y \mid \mathcal{D}_{\mathbf{X}}, \mathbf{w}) = \sum_{j=1}^{N} \ln P(y^j \mid \mathbf{x}^j, \mathbf{w})$$

Expressing Conditional Log Likelihood

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

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

$$l(\mathbf{w}) \equiv \sum_{j=1}^{N} \ln P(y^j|\mathbf{x}^j,\mathbf{w})$$

$$= \sum_j \begin{cases} P(Y=1|x^j,w), & \text{when } y^j=1 \\ P(Y=0|x^j,w), & \text{when } y^j=0 \end{cases}$$

$$\ell(\mathbf{w}) = \sum_{j=1}^{N} y^j \ln P(Y=1|\mathbf{x}^j,\mathbf{w}) + (1-y^j)\ln P(Y=0|\mathbf{x}^j,\mathbf{w})$$

$$= \sum_j y^j \ln \frac{e^{w_0+\sum_i w_i x_i^j}}{1+e^{w_0+\sum_i w_i x_i^j}} + (1-y^j)\ln \frac{1}{1+e^{w_0+\sum_i w_i x_i^j}}$$

$$= \sum_{j=1}^{N} y^j\left(w_0+\sum_i w_i x_i^j\right) - \ln\left(1+e^{w_0+\sum_i w_i x_i^j}\right)$$

the beast we want to maximize WRT w

Maximizing Conditional Log Likelihood

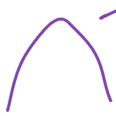
$P(Y=0|X,W) = \frac{1}{1+exp(w_0+\sum_i w_i X_i)}$

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

want $\max_w$

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

$$= \sum_j y^j(w_0+\sum_i^n w_i x_i^j) - \ln(1+exp(w_0+\sum_i^n w_i x_i^j))$$



Good news: $l(\mathbf{w})$ is concave function of **w**, no local optima problems

Bad news: no closed-form solution to maximize $l(\mathbf{w})$

Good news: concave functions easy to optimize

gradient ascent

Maximize Conditional Log Likelihood: Gradient ascent

$\frac{\partial}{\partial w} \ln f(w) = \frac{f'(w)}{f(w)}$ $\qquad \frac{\partial}{\partial w} e^f = f' e^f$

$$l(\mathbf{w}) = \sum_{j=1}^{N} y^j (w_0 + \sum_{i=1}^{k} w_i x_i^j) - \ln(1 + exp(w_0 + \sum_{i=1}^{k} w_i x_i^j))$$

$\nabla_w \ell(w)$: $\ell > 0$

$$\frac{\partial \ell}{\partial w_\ell} = \sum_{j=1}^{N} \left[y^j x_\ell^j - \underbrace{\frac{x_\ell^j \cdot e^{w_0 + \sum_i w_i x_i^j}}{1 + e^{w_0 + \sum_i w_i x_i^j}}}_{P(Y=1 \mid x^j, w)} \right]$$

$$\frac{\partial \ell}{\partial w_\ell} = \sum_{j=1}^{N} x_\ell^j \underbrace{\left(y^j - P(Y=1 \mid x^j, w) \right)}_{\text{diff truth \& prediction}}$$

weighted by how much feature ℓ plays a role in point j

Gradient Ascent for LR

initialize $w^{(0)} = 0$ or something else → all converge to same place, concavity

Gradient ascent algorithm: iterate until change < ε

$$w_0^{(t+1)} \leftarrow w_0^{(t)} + \eta \sum_{j=1}^{N} [y^j - \hat{P}(Y^j = 1 \mid \mathbf{x}^j, \mathbf{w}^{(t)})]$$

small step

gradient direction

For i=1,…,k,

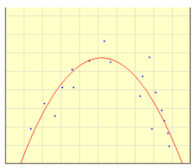
$$w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \sum_{j=1}^{N} x_i^j [y^j - \hat{P}(Y^j = 1 \mid \mathbf{x}^j, \mathbf{w}^{(t)})]$$

repeat

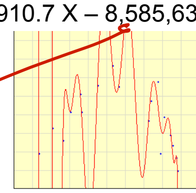
Regularization in linear regression

- Overfitting usually leads to very large parameter choices, e.g.:

-2.2 + 3.1 X – 0.30 X^2



-1.1 + 4,700,910.7 X – 8,585,638.4 X^2 + …



- Regularized least-squares (a.k.a. ridge regression), for $\lambda>0$:

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \sum_j \left(t(\mathbf{x}_j) - \sum_i w_i h_i(\mathbf{x}_j) \right)^2 + \lambda \sum_{i=1}^{k} w_i^2$$

$\|w\|_2^2$

$\|w\|_1$

Linear Separability

$\exists w \ \forall x^j$ $\quad w_0+\sum_i w_i x_i^j>0 \quad y^j=1$

$w_0+\sum_i w_i x_i^j<0 \quad y^j=0$

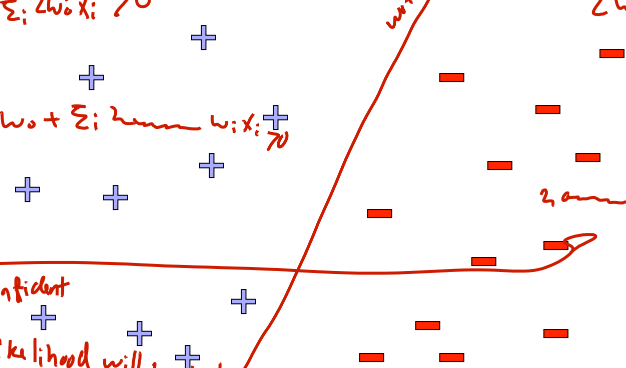
$w_0+\sum_i w_i x_i>0$

$2w_0+\sum_i 2w_i x_i>0$

$w_0+\sum_i w_i x_i=0$

$w_0+\sum_i w_i x_i<0$

$2w_0+\sum_i 2w_i x_i<0$



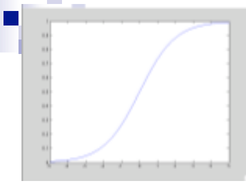
2~~~ $w_0+\sum_i$ 2~~~ $w_i x_i > 0$

2~~~ $w_0+\sum_i$ 2~~~ $w_i x_i < 0$

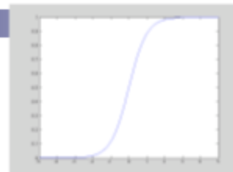
more confident

log likelihood will be higher

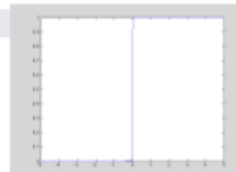
Large parameters → Overfitting



$\frac{1}{1+e^{-x}}$



$\frac{1}{1+e^{-2x}}$



$\frac{1}{1+e^{-100x}}$

- If data is linearly separable, weights go to infinity

$y^j=0$

$P(y=0 \mid w, x^j) \rightarrow 1$ $\qquad ||w|| \rightarrow \infty$

  - In general, leads to overfitting:
- Penalizing high weights can prevent overfitting…

Regularized Conditional Log Likelihood

- Add regularization penalty, e.g., L_2:

$$\ell(\mathbf{w}) = \ln \prod_{j=1}^{N} P(y^j|\mathbf{x}^j, \mathbf{w}) - \frac{\lambda}{2}\underbrace{||\mathbf{w}||_2^2}_{\sum_{i=1}^{k} w_i^2}$$

- Practical note about w_0:

don't regularize

- Gradient of regularized likelihood:

$$\frac{\partial \ell}{\partial w_\ell} = \text{same as before} + \frac{\partial}{\partial w_\ell} = -\lambda w_\ell$$

Standard v. Regularized Updates

- Maximum conditional likelihood estimate

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \quad \ln \prod_{j=1}^{N} P(y^j|\mathbf{x}^j, \mathbf{w})$$

Standard update $\forall w_i, i>0$

$$\boxed{w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \sum_j x_i^j [y^j - \hat{P}(Y^j = 1 \mid \mathbf{x}^j, \mathbf{w}^{(t)})]}$$

- Regularized maximum conditional likelihood estimate

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \quad \ln \prod_{j=1}^{N} P(y^j|\mathbf{x}^j, \mathbf{w}) - \frac{\lambda}{2}\sum_{i=1}^{k} w_i^2$$

regularization contribution to derivative

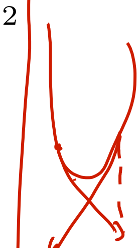
$$\boxed{w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \left\{ -\lambda w_i^{(t)} + \sum_j x_i^j [y^j - \hat{P}(Y^j = 1 \mid \mathbf{x}^j, \mathbf{w}^{(t)})] \right\}}$$

push w_i towards 0

$\ell(w^{(t+1)}) - \ell(w^{(t)})$

Please Stop!! Stopping criterion

$$\ell(\mathbf{w}) = \ln \prod_j P(y^j|\mathbf{x}^j, \mathbf{w})) - \lambda||\mathbf{w}||_2^2$$



- When do we stop doing gradient descent? $\varepsilon > 0$

OPT

$\ell(w^*) - \ell(w^{(t)}) < \varepsilon$

implies

- Because $l(\mathbf{w})$ is strongly concave:
  - i.e., because of some technical condition

$$\underbrace{\ell(\mathbf{w}^*) - \ell(\mathbf{w})}_{\text{don't know this}} \leq \frac{1}{2\lambda}||\nabla \ell(\mathbf{w})||_2^2 < \varepsilon$$

- Thus, stop when: $||\nabla \ell(w)||_2^2 < 2\lambda\varepsilon$

$\eta = \frac{\alpha}{\sqrt{t}}$

Sometimes OK $\eta = \alpha$

Digression: Logistic regression for more than 2 classes

- Logistic regression in more general case (C classes), where *Y in* {0,…,C-1}

For C class, need $(C-1)(K+1)$ params

$\forall$ classes $c \in \{1, \ldots, C-1\}$

$$P(Y=c|X,w) = \frac{e^{w_{c0} + \sum_i w_{ci} x_i}}{1 + \text{something}}$$

For $C=0$

$$P(C=0|X,w) = 1 - \sum_{c=1}^{C-1} P(Y=c|X,w)$$

$C=2$, K features

$\Rightarrow$ $K+1$ params to learn $w_0, \ldots, w_K$

$$P(Y=1|X,w) = \frac{e^{w_0 + \sum_i w_i x_i}}{1 + e^{w_0 + \sum_i w_i x_i}}$$

$$P(Y=0|X,w) = 1 - P(Y=1|X,w)$$

Digression: Logistic regression more generally

One against all is an alternative

- Logistic regression in more general case, where *Y in* {0,…,C-1}

for *c>0*

$$P(Y=c|\mathbf{x},\mathbf{w}) = \frac{\exp(w_{c0} + \sum_{i=1}^k w_{ci} x_i)}{1 + \sum_{c'=1}^{C-1} \exp(w_{c'0} + \sum_{i=1}^k w_{c'i} x_i)}$$

for *c=0* (normalization, so no weights for this class)

$$P(Y=0|\mathbf{x},\mathbf{w}) = \frac{1}{1 + \sum_{c'=1}^{C-1} \exp(w_{c'0} + \sum_{i=1}^k w_{c'i} x_i)}$$

Learning procedure is basically the same as what we derived! derivative is a little more complicated