Logistic Regression

Midterm Review Friday

- Come with questions

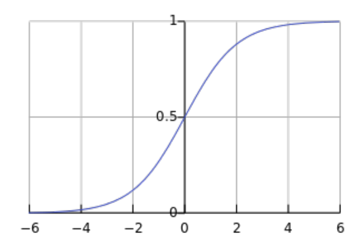
Logistic Regression

Idea:

- Naïve Bayes allows computing P(Y|X) by learning P(Y) and P(X|Y)
- Why not learn P(Y|X) directly?

Idea

- Similar to perceptron, but sigmoid function applied to linearity.



$$S(z) = \frac{1}{1+e^{-z}}$$

- Error function is based on Max Likelihood →
  as a result we get genuine probabilities as output of prediction (not just 1,0 discrete values).

Models so far, and the new one

$$z = \sum_{i=0}^{m} w_i x_i$$

Perceptron	Linear regression	Logistic regression
$h(\mathbf{x},\mathbf{w}) = \text{sign}(z)$	$h(\mathbf{x},\mathbf{w}) = z$	$h(\mathbf{x},\mathbf{w}) = \text{S}(z)$

Output $h(\mathbf{x},\mathbf{w})$ will be interpreted as probability.

Why? Because of the range of S, and particular cost function we'll optimize.

Probability Interpretation

- Assume tuples ($\mathbf{x}$, y), where y is binary, are generated from some noisy data-source according to some distribution f.

$$p(y \mid \mathbf{x}) = \begin{cases} f(\mathbf{x}) & \text{for } y = 0 \\ 1 - f(\mathbf{x}) & \text{for } y = 1 \end{cases}$$

For mathematical convenience we will use [0,1] as our label set instead of [-1,1]

- We will learn $f(\mathbf{x})$ by approximating it with the sigmoid S(z), i.e. S($\mathbf{w.x}$)

$$\mathbf{w} = [w_0 = b, w_1, .., w_m] \qquad \mathbf{x} = [x_0 = 1, x_1, .., x_m]$$

- Makes sense as S(z) is a function from 0 to 1.

Approximation

- What is the probability of y=0? $p(y=0\,|\,\mathbf{x}) = \frac{1}{1+e^{\mathbf{w}'\mathbf{x}}}$
  (according to our approximation)

- What is the probability of y=1? $p(y=1\,|\,\mathbf{x}) = 1 - \frac{1}{1+e^{\mathbf{w}'\mathbf{x}}}$
  (according to our approximation)

$$= \frac{1+e^{\mathbf{w}'\mathbf{x}}-1}{1+e^{\mathbf{w}'\mathbf{x}}}$$

$$= \frac{e^{\mathbf{w}'\mathbf{x}}}{1+e^{\mathbf{w}'\mathbf{x}}}$$

Training Logistic Regression: Maximum Conditional Likelihood Estimation (MCLE)

- we have L training examples: $\{<X^1,Y^1>, \ldots <X^N,Y^N>\}$

- maximum likelihood estimate for parameters W

$$W_{MLE} = \arg\max_w \; P(<X^1,Y^1>, \ldots <X^N,Y^N> |W)$$

$$= \arg\max_w \prod_{i=1}^{N} P(<X^i,Y^i> |W)$$

- maximum conditional likelihood estimate

Training Logistic Regression: MCLE

- Choose parameters W=<w_0, ... w_n> to maximize conditional likelihood of training data

where
$$p(y=0\,|\,\mathbf{x}) = \frac{1}{1+e^{\mathbf{w}'\mathbf{x}}}$$
$$p(y=1\,|\,\mathbf{x}) = \frac{e^{\mathbf{w}'\mathbf{x}}}{1+e^{\mathbf{w}'\mathbf{x}}}$$

- Training data D = $\{<X^1,Y^1>, \ldots <X^N,Y^N>\}$
- Data likelihood = $\prod_{i=1}^{N} P(<X^i,Y^i>|W)$
- Data conditional likelihood = $\prod_{i=1}^{N} P(Y^i|X^i,W)$

$$W_{MCLE} = \arg\max_w \prod_{i=1}^{N} P(Y^i|X^i,W)$$

Expressing Conditional Log Likelihood

$$\mathcal{L}(W) \equiv ln \left(\prod_i P(Y^i|X^i,W) \right)$$

$$= \sum_i ln\; P(Y^i|X^i,W)$$

$$p(y=0\,|\,\mathbf{x}) = \frac{1}{1+e^{\mathbf{w}'\mathbf{x}}}$$
$$p(y=1\,|\,\mathbf{x}) = \frac{e^{\mathbf{w}'\mathbf{x}}}{1+e^{\mathbf{w}'\mathbf{x}}}$$

$$\mathcal{L}(W) = \sum_i \underline{Y^i\; ln\; P(Y^i=1|X^i,W)} + \sum_i \underline{(1-Y^i)\; ln\; P(Y^i=0|X^i,W)}$$

For the samples with Y^i=1

For the samples with Y^i=0

Expressing Conditional Log Likelihood

$$\mathcal{L}(W) = \sum_i Y^i\; ln\; P(Y^i=1|X^i,W) + \sum_i (1-Y^i)\; ln\; P(Y^i=0|X^i,W)$$

$$= \sum_i Y^i\; ln\; \frac{P(Y^i=1|X^i,W)}{P(Y^i=0|X^i,W)} + \sum_i ln\; P(Y^i=0|X^i,W)$$

Expressing Conditional Log Likelihood

$$\mathcal{L}(W) = \sum_i Y^i\; ln\; P(Y^i=1|X^i,W) + \sum_i (1-Y^i)\; ln\; P(Y^i=0|X^i,W)$$

$$= \sum_i Y^i\; ln\; \frac{P(Y^i=1|X^i,W)}{P(Y^i=0|X^i,W)} + \sum_i ln\; P(Y^i=0|X^i,W)$$

$$= \sum_i Y^i\;(w'x^i) - \sum_i ln(1+e^{w'x^i})$$

$$p(y=0\,|\,\mathbf{x}) = \frac{1}{1+e^{\mathbf{w}'\mathbf{x}}}$$
$$p(y=1\,|\,\mathbf{x}) = \frac{e^{\mathbf{w}'\mathbf{x}}}{1+e^{\mathbf{w}'\mathbf{x}}}$$

Maximizing Conditional Log Likelihood

$$\mathcal{L}(W) = \sum_i Y^i\ (w'x^i) - \sum_i ln(1 + e^{w'x^i})$$

$$\frac{d}{dw}\mathcal{L}(W) = \sum_i Y^i x^i - \sum_i \frac{x^i e^{w'x^i}}{1 + e^{w'x^i}}$$

$$= \sum_i x^i (Y^i - P(Y = 1|W, x^i))$$

- **Good news: *L(W)* is concave function of *W***
- **Bad news: no closed-form solution to maximize *L(W) [i.e., no canonical equation]***

Iterative Method

- Start at $\mathbf{w}_0$=**1**; take a step along **steepest slope**
- Fixed step size:

$$\mathbf{w}_1 = \mathbf{w}_0 + \eta \mathbf{v}$$

- **v** is a **vector** in the direction of the **steepest slope**.
  - What's the steepest slope?

Gradient Descent Algorithm

Initialize $\mathbf{w}_0$=**0**

For *t*=0,1,2,…do

Compute the gradient and update the weights

$$\frac{d}{dw}\mathcal{L}(W) = \sum_i x^i (Y^i - P(Y = 1|W, x^i)$$

$$w \leftarrow w + \kappa \left(\sum_i x^i (Y^i - P(Y = 1|W, x^i) \right)$$

Iterate with the next step until **w** doesn't change too much (or for a fixed number of iterations)

Return final **w**.

Making predictions

- A new tuple comes: (**x**,?)

$$p(y=0 \mid \mathbf{x}) = \frac{1}{1+e^{\mathbf{w}'\mathbf{x}}}$$

$$p(y=1 \mid \mathbf{x}) = \frac{e^{\mathbf{w}'\mathbf{x}}}{1+e^{\mathbf{w}'\mathbf{x}}}$$

- Fix a threshold in [0,1] to make predictions.

$p(y=1 \mid \mathbf{x}) >$ threshold	Predict *y*=1
$p(y=1 \mid \mathbf{x}) \le$ threshold	Predict *y*=0

Example

GPA, GRE, and success.
100, 800, 1
90, 800, 1
90, 700, 1
70, 600, 0
60, 700, 0
60, 700, 1
50, 600, 0
50, 650, 01
50, 800, 1
50, 700, 0
50, 700, 1

Dummy	GPA	GRE	y
1	1.0	1.0	1
1	0.9	1.0	1
1	0.9	0.875	1
1	0.7	0.75	0
1	0.6	0.875	0
1	0.6	0.875	1
1	0.5	0.75	0
1	0.5	0.8125	0
1	0.5	1.0	1
1	0.5	0.875	0
1	0.5	0.875	1

Scaled so that max GPA/GRE is 1

Example

Logistic regression:
-7.44 + 4.4056*GPA + 5.57 * GRE

$$p(y=0 \mid gpa, gre) = \frac{1}{1+e^{(-7.44 + 4.4056*\mathrm{GPA} + 5.57 * \mathrm{GRE})}}$$

Fix a threshold in [0,1] to make predictions.

$p(y=1 \mid gpa, gre) >$ threshold	Predict *y*=1
$p(y=1 \mid gpa, gre) \le$ threshold	Predict *y*=0

Odds

Definition: $$odds(0 \text{ vs. } 1 \text{ given } \mathbf{x}) = \frac{p(y=0 \mid \mathbf{x})}{p(y=1 \mid \mathbf{x})}$$

Formula: $$odds(0 \text{ vs. } 1 \text{ given } \mathbf{x}) = \frac{\frac{1}{1+e^{\mathbf{w}'\mathbf{x}}}}{\frac{e^{\mathbf{w}'\mathbf{x}}}{1+e^{\mathbf{w}'\mathbf{x}}}}$$

$$= \frac{1}{e^{\mathbf{w}'\mathbf{x}}}$$

$$= e^{-\mathbf{w}'\mathbf{x}}$$

We predict ? if this number is less than ?

Interpretation

$$odds(\text{successful vs. unsuccessful given } gpa \text{ and } gre) = e^{-7.44 + 4.4056 \cdot \text{GPA} + 5.57 \cdot \text{GRE}}$$

If GPA increases by .1 then the odds of success will increase by 55%

$$e^{0.441} \approx 1.55$$

If GRE increases by .1 then the odds of success will increase by 75%

$$e^{0.557} \approx 1.75$$

Logistic Regression in Matlab

```
kappa = 0.1;
[n,p] = size(X);
num_its = 500;

L = zeros(1,num_its);
for t=1:num_its,
  L(t) = sum(y.*(X*w) - log(1+exp(X*w)));
  w = w + kappa * ...
       sum( X.*(repmat(y - exp(X*w)./ (1+exp(X*w)),1,p)))';
end;
plot(L)
ylabel('Likelihood')
xlabel('Iteration')
fprintf('After optimizing, accuracy is %.2f\n',1-sum(abs((exp(-1*X*w)<1)-y))/length(y))
```

```
X=[
1 1.0   1.0;
1 0.9   1.0;
1 0.9   0.875;
1 0.7   0.75;
1 0.6   0.875;
1 0.6   0.875;
1 0.5   0.75;
1 0.5   0.8125;
1 0.5   1.0;
1 0.5   0.875;
1 0.5   0.875];
```

```
y=[1 1 1 0 0 1 0 0 1 0 1]';

w=[1; 1; 1];
```

```
After optimizing, accuracy is
0.82
```