ISTA 421/521 Introduction to Machine Learning

Lecture 16: Estimation Methods: Gradient Descent, Newton-Raphson

Clay Morrison

clayton@sista.arizona.edu

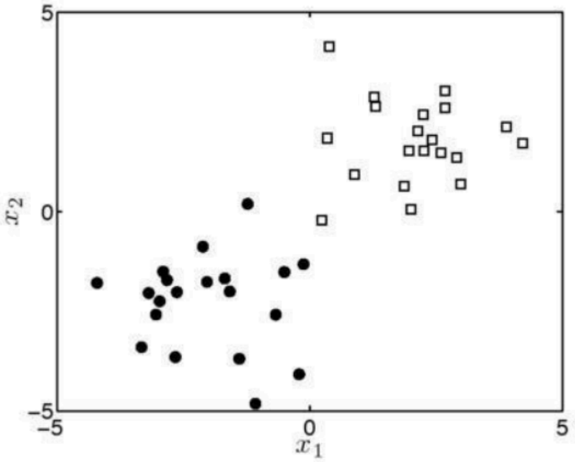
Gould-Simpson 819

Phone 621-6609

21 October 2014

Binary Classification!

- A very common type of problem
- *Many* different approaches; we'll start with a probabilistic method: logistic regression



two attributes (x_1 and x_2)

binary target, $t = \{0, 1\}$

t = 0 are dark circles
t = 1 are white squares

The Likelihood

- Assume the elements of **t** are independent, conditioned on **w**

$$p(\mathbf{t}|\mathbf{X},\mathbf{w}) = \prod_{n=1}^{N} p(t_n|\mathbf{x}_n,\mathbf{w})$$

- Previously, **t** was Gaussian distributed b/c the target was real-valued. Now the target is a binary class label (0 or 1), so likelihood is a different RV:

$$p(\mathbf{t}|\mathbf{X},\mathbf{w}) = \prod_{n=1}^{N} P(T_n = t_n|\mathbf{x}_n,\mathbf{w})$$

a binary random variable

The Likelihood

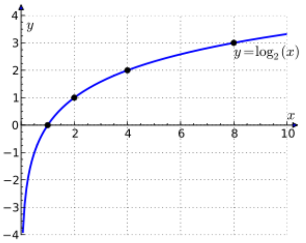
$$p(\mathbf{t}|\mathbf{X},\mathbf{w}) = \prod_{n=1}^{N} P(T_n = t_n|\mathbf{x}_n,\mathbf{w})$$

- Want likelihood to…
  - … be high if model assigns high probabilities for class 1 when we observe class 1, and high probabilities for class 0 when we observe class 0.
  - … have a maximum value of 1 where all of the training points are predicted perfectly.
- **Popular approach**: take simple linear function and pass the result through a second function that "squashes" its output, to ensure it produces a valid probability.

Logistic Likelihood

- The logistic likelihood is formally derived as a result of modeling the ***log-odds*** (aka the ***logit***):

$$\log\left(\frac{P(T_{\text{new}}=1|\mathbf{x}_{\text{new}},\mathbf{w})}{P(T_{\text{new}}=0|\mathbf{x}_{\text{new}},\mathbf{w})}\right) = \mathbf{w}^{\top}\mathbf{x}_{\text{new}}$$



- There are no constraints on this value: it can take any real value.

$$P(T_{\text{new}} = 1|\mathbf{x}_{\text{new}},\mathbf{w}) \ll P(T_{\text{new}} = 0|\mathbf{x}_{\text{new}},\mathbf{w})$$ Large *negative*

$$P(T_{\text{new}} = 1|\mathbf{x}_{\text{new}},\mathbf{w}) \gg P(T_{\text{new}} = 0|\mathbf{x}_{\text{new}},\mathbf{w})$$ Large positive

Logistic Likelihood

Example of a **generalized linear model**: linear model passed through a transformation to model a quantity of interest.

- Now, derive $P(T_{\text{new}} = 1|\mathbf{x}_{\text{new}},\mathbf{w})$

Note: $P(T_{\text{new}} = 0|\mathbf{x}_{\text{new}},\mathbf{w}) = 1 - P(T_{\text{new}} = 1|\mathbf{x}_{\text{new}},\mathbf{w})$

$$\begin{aligned}
\log\left(\frac{P(T_{new}=1|\mathbf{x}_{new},\mathbf{w})}{P(T_{new}=0|\mathbf{x}_{new},\mathbf{w})}\right) &= \mathbf{w}^{\top}\mathbf{x} \\
\frac{P(T_{new}=1|\mathbf{x}_{new},\mathbf{w})}{P(T_{new}=0|\mathbf{x}_{new},\mathbf{w})} &= \exp(\mathbf{w}^{\top}\mathbf{x}) \\
\frac{P(T_{new}=1|\mathbf{x}_{new},\mathbf{w})}{1-P(T_{new}=1|\mathbf{x}_{new},\mathbf{w})} &= \exp(\mathbf{w}^{\top}\mathbf{x}) \\
P(T_{new}=1|\mathbf{x}_{new},\mathbf{w}) &= \exp(\mathbf{w}^{\top}\mathbf{x})(1-P(T_{new}=1|\mathbf{x}_{new},\mathbf{w})) \\
P(T_{new}=1|\mathbf{x}_{new},\mathbf{w}) &= \exp(\mathbf{w}^{\top}\mathbf{x}) - \exp(\mathbf{w}^{\top}\mathbf{x})P(T_{new}=1|\mathbf{x}_{new},\mathbf{w}) \\
P(T_{new}=1|\mathbf{x}_{new},\mathbf{w}) + \exp(\mathbf{w}^{\top}\mathbf{x})P(T_{new}=1|\mathbf{x}_{new},\mathbf{w}) &= \exp(\mathbf{w}^{\top}\mathbf{x}) \\
P(T_{new}=1|\mathbf{x}_{new},\mathbf{w})(1+\exp(\mathbf{w}^{\top}\mathbf{x})) &= \exp(\mathbf{w}^{\top}\mathbf{x}) \\
P(T_{new}=1|\mathbf{x}_{new},\mathbf{w}) &= \frac{\exp(\mathbf{w}^{\top}\mathbf{x})}{1+\exp(\mathbf{w}^{\top}\mathbf{x})} \\
P(T_{new}=1|\mathbf{x}_{new},\mathbf{w}) &= \frac{1}{1+\exp(-\mathbf{w}^{\top}\mathbf{x})}
\end{aligned}$$

So the logistic likelihood is really modeling the log-odds with a linear model!

The **Logistic** (likelihood) function

Logistic Likelihood

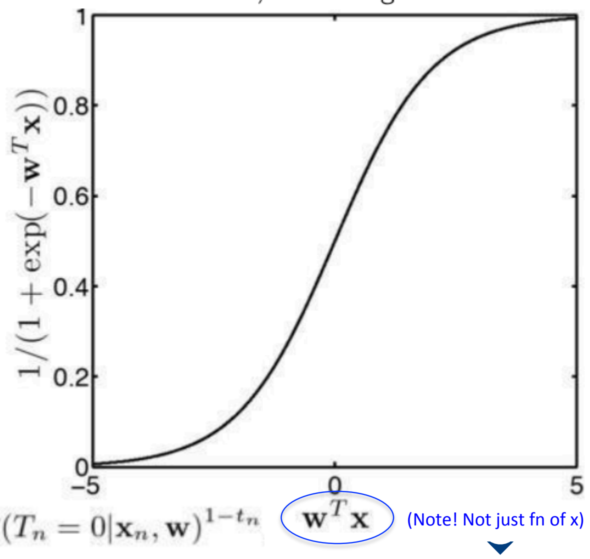
$$p(\mathbf{t}|\mathbf{X},\mathbf{w}) = \prod_{n=1}^{N} P(T_n = t_n|\mathbf{x}_n,\mathbf{w})$$

The Logistic (or Sigmoid) function

$$P(T_n = 1|\mathbf{x}_n,\mathbf{w}) = \frac{1}{1+\exp(-\mathbf{w}^\top\mathbf{x}_n)}$$

Linear component

As $\mathbf{w}^\top\mathbf{x}$ increases, the value converges to 1 as it decreases, it converges to 0.



(Note! Not just fn of x)

When target is 0:

$$\begin{aligned} P(T_n = 0|\mathbf{x}_n,\mathbf{w}) &= 1 - P(T_n = 1|\mathbf{x}_n,\mathbf{w}) \\ &= 1 - \frac{1}{1+\exp(-\mathbf{w}^\top\mathbf{x}_n)} \\ &= \frac{\exp(-\mathbf{w}^\top\mathbf{x}_n)}{1+\exp(-\mathbf{w}^\top\mathbf{x}_n)}. \end{aligned}$$

Combine both into a single probability function

$$P(T_n = t_n|\mathbf{x}_n,\mathbf{w}) = P(T_n = 1|\mathbf{x}_n,\mathbf{w})^{t_n} P(T_n = 0|\mathbf{x}_n,\mathbf{w})^{1-t_n}$$

The Likelihood

$$P(T_n = t_n|\mathbf{x}_n,\mathbf{w}) = P(T_n = 1|\mathbf{x}_n,\mathbf{w})^{t_n} P(T_n = 0|\mathbf{x}_n,\mathbf{w})^{1-t_n}$$

Substitute in the component likelihoods to get the final likelihood function

$$\begin{aligned} p(\mathbf{t}|\mathbf{X},\mathbf{w}) &= \prod_{n=1}^{N} P(T_n = t_n|\mathbf{x}_n,\mathbf{w}) \\ &= \prod_{n=1}^{N} P(T_n = 1|\mathbf{x}_n\mathbf{w})^{t_n} P(T_n = 0|\mathbf{x}_n,\mathbf{w})^{1-t_n} \\ &= \prod_{n=1}^{N} \left(\frac{1}{1+\exp(-\mathbf{w}^\top\mathbf{x}_n)}\right)^{t_n} \left(\frac{\exp(-\mathbf{w}^\top\mathbf{x}_n)}{1+\exp(-\mathbf{w}^\top\mathbf{x}_n)}\right)^{1-t_n} \end{aligned}$$

Bayesian Logistic Regression

$$\mathbf{x}_n = \begin{bmatrix} x_{n1} \\ x_{n2} \end{bmatrix}, \ \mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}, \ \mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_N^\top \end{bmatrix}$$

Want to compute the posterior density over the parameters **w** of the model

$$p(\mathbf{w}|\mathbf{t}, \mathbf{X}) = \frac{p(\mathbf{t}|\mathbf{X}, \mathbf{w})p(\mathbf{w})}{p(\mathbf{t}|\mathbf{X})}$$

$$p(\mathbf{t}|\mathbf{X}) = \int p(\mathbf{t}|\mathbf{X}, \mathbf{w})p(\mathbf{w}) \ d\mathbf{w}$$

Prior: $p(\mathbf{w}) = \mathcal{N} \ (\mathbf{0}, \ \sigma^2\mathbf{I})$

Likelihood:

$$p(\mathbf{t}|\mathbf{X}, \mathbf{w}) = \prod_{n=1}^{N} \left(\frac{1}{1 + \exp(-\mathbf{w}^\top \mathbf{x}_n)} \right)^{t_n} \left(\frac{\exp(-\mathbf{w}^\top \mathbf{x}_n)}{1 + \exp(-\mathbf{w}^\top \mathbf{x}_n)} \right)^{1-t_n}$$

Prior:

$$p(\mathbf{w}|\sigma^2) = \mathcal{N}(0, \sigma^2\mathbf{I})$$

Once we have the Posterior… $P(\mathbf{w}|\mathbf{X}, \mathbf{t}, \sigma^2)$

… can predict the response (class) of new objects by taking the expectation with respect to this density:

$$P(t_{\text{new}} = 1|\mathbf{x}_{\text{new}}, \mathbf{X}, \mathbf{t}) = \mathbf{E}_{p(\mathbf{w}|\mathbf{X},\mathbf{t},\sigma^2)} \left\{ \frac{1}{1 + \exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_{\text{new}})} \right\}$$

Problem: the posterior is not in a standard form.

The numerator is fine: just calc prior and likelihood at observations, then multiply.

It's the denominator (marginal likelihood) that is the problem: can't integrate…

$$Z^{-1} = p(\mathbf{t}|\mathbf{X}, \sigma^2) = \int p(\mathbf{t}|\mathbf{X}, \mathbf{w})p(\mathbf{w}|\sigma^2) \ d\mathbf{w}$$

$$g(\mathbf{w};\mathbf{X},\mathbf{t},\sigma^2) = \overset{\text{likelihood}}{p(\mathbf{t}|\mathbf{X},\mathbf{w})}\overset{\text{prior}}{p(\mathbf{w}|\sigma^2)}$$

$$\text{marginal likelihood}\quad Z^{-1} = p(\mathbf{t}|\mathbf{X},\sigma^2) = \int p(\mathbf{t}|\mathbf{X},\mathbf{w})p(\mathbf{w}|\sigma^2)\ d\mathbf{w}$$

$$p(\mathbf{w}|\mathbf{X},\mathbf{t},\sigma^2) = Z^{-1}g(\mathbf{w};\mathbf{X},\mathbf{t},\sigma^2)$$

Our Options

1. Find the single value of **w** that corresponds to the highest value of the posterior. As $g(\mathbf{w};\mathbf{X},\mathbf{t},\sigma^2)$ is proportional to the posterior, a maximum of *g* will also correspond to a maximum of the posterior. *Z* $^{-1}$ is not a function of **w**
2. Approximate $P(\mathbf{w}|\mathbf{X},\mathbf{t},\sigma^2)$ with some other density that we can compute analytically.
3. Sample directly from the posterior $P(\mathbf{w}|\mathbf{X},\mathbf{t},\sigma^2)$, knowing only $g(\mathbf{w};\mathbf{X},\mathbf{t},\sigma^2)$

Method 1: MAP point estimate

- While we cannot derive a direct analytic posterior density, we can compute something proportional to it:

$$g(\mathbf{w};\mathbf{X},\mathbf{t},\sigma^2) = p(\mathbf{t}|\mathbf{X},\mathbf{w})p(\mathbf{w}|\sigma^2)$$

- We will find the value of **w** that maximizes *g*
- This will correspond to the value at the maximum of the posterior.
- This will be the most likely value $\hat{\mathbf{w}}$ under the posterior.

The MAP Estimate

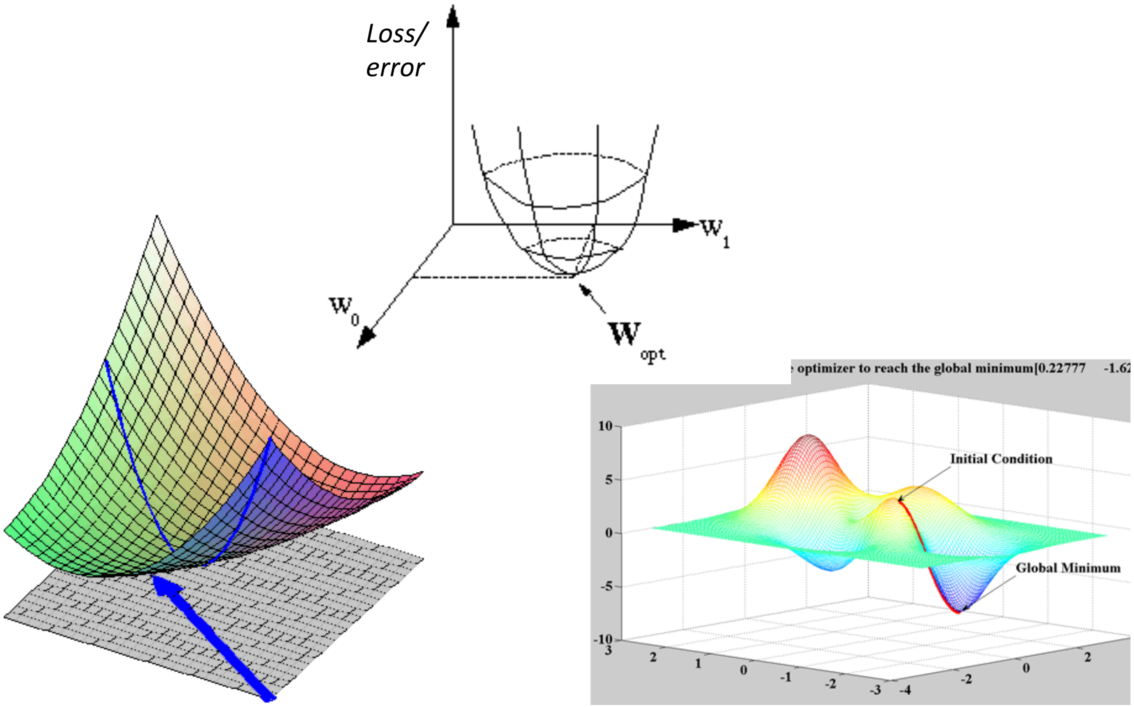
- It will again be helpful to work with the log

$$\log g(\mathbf{w}; \mathbf{X}, \mathbf{t}) = \log p(\mathbf{t}|\mathbf{X}, \mathbf{w}) + \log p(\mathbf{w}|\sigma^2)$$

- However, unlike the max likelihood solution for the linear model, we **cannot** get an exact analytical expression for **w** by differentiating this expression and setting it to 0.
- Instead, need to use an **optimization method**: guess value of **w** and apply *incremental update adjustments* to our estimate of **w** that increase log *g* until a maximum is found.

Optimization with Gradient Methods

Loss (or Error) Gradient Descent (Or Likelihood Gradient Ascent)



Solving The Normal Equations (Matrix)

- The matrix version of the squared loss, multiplied out for easier differentiation

$$\mathcal{L} = \frac{1}{N}\mathbf{w}^\top\mathbf{X}^\top\mathbf{X}\mathbf{w} - \frac{2}{N}\mathbf{w}^\top\mathbf{X}^\top\mathbf{t}$$

- Drop the constant 2/N's

$$\begin{aligned} \frac{\partial\mathcal{L}}{\partial\mathbf{w}} &= \frac{2}{N}\mathbf{X}^\top\mathbf{X}\mathbf{w} - \frac{2}{N}\mathbf{X}^\top\mathbf{t} \\ &= \mathbf{X}^\top\mathbf{X}\mathbf{w} - \mathbf{X}^\top\mathbf{t} \end{aligned}$$

The LMS update rule (Widrow-Hoff)

- The ***batch*** update version

$$
\begin{aligned}
\mathbf{w} &:= \mathbf{w} - \alpha \frac{\partial \mathcal{L}}{\partial \mathbf{w}} \\
&:= \mathbf{w} - \alpha \left(\mathbf{X}^\top \mathbf{X} \mathbf{w} - \mathbf{X}^\top \mathbf{t} \right) \\
&:= \mathbf{w} - \alpha \sum_{n=1}^{N} \left(t_n - \mathbf{w}^\top \mathbf{x}_n \right) \mathbf{x}_n
\end{aligned}
$$

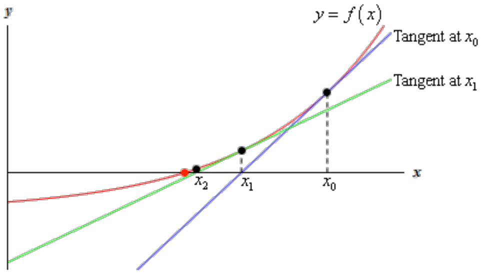
Properties:

(1) Step is in the direction of the gradient's steepest descent (negative of tangent slope)

(2) Magnitude of the update is proportional to the error term

The "algorithm"

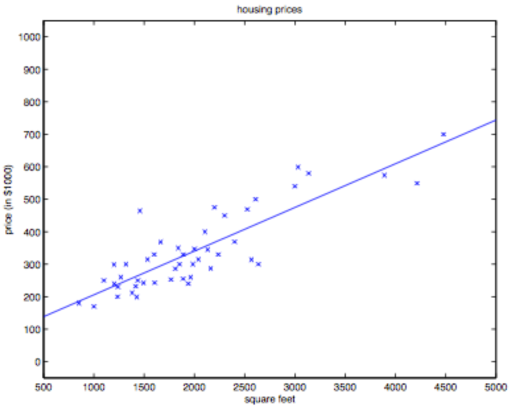
```
repeat
    $\mathbf{w} := \mathbf{w} - \alpha \sum_{n=1}^{N} \left( t_n - \mathbf{w}^\top \mathbf{x}_n \right) \mathbf{x}_n$
until $convergence$;
```

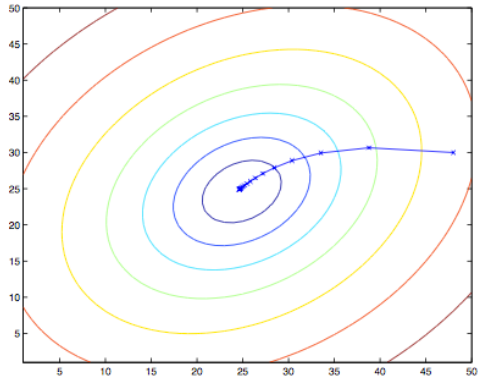


Version with design matrix computations (equivalent)

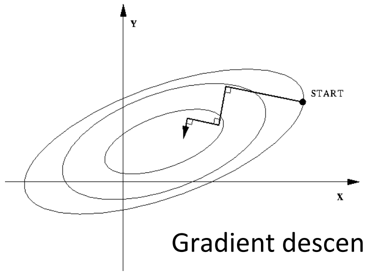
```
repeat
    $\mathbf{w} := \mathbf{w} - \alpha \left( \mathbf{X}^\top \mathbf{X} \mathbf{w} - \mathbf{X}^\top \mathbf{t} \right)$
until $convergence$;
```

Gradient Descent with line in 2D





Plot of Loss gradient in space of Parameters: w_0 and w_1



Gradient descent path is not always so "smooth"

Exploring the Landscape

- **Local Maxima**: peaks that aren't the highest point in the space
- **Plateaus:** the space has a broad flat region that gives the search algorithm no direction (random walk)
- **Ridges:** flat like a plateau, but with drop-offs to the sides; steps to the North, East, South and West may go down, but a step to the NW may go up.

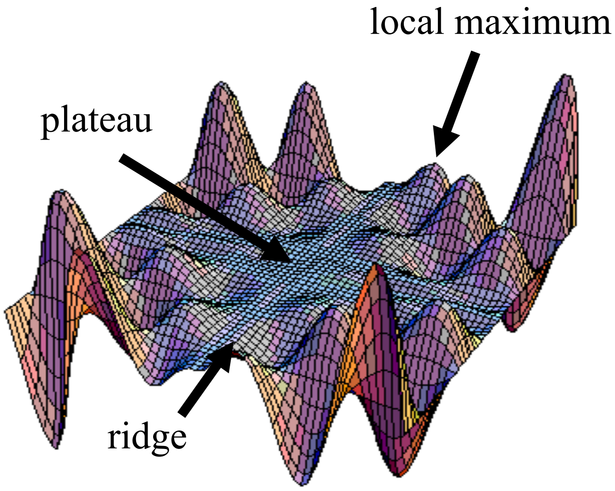


Image from: http://classes.yale.edu/fractals/CA/GA/Fitness/Fitness.html

Batch vs. Incremental (Stochastic)

- Batch gradient descent has to scan through the entire training set before taking a single step.
- Costly if *N* is large
- Stochastic gradient descent can start making progress right away, and continues to make progress with each example it looks at.

The LMS update rule (Widrow-Hoff) for *Stochastic* Gradient Descent

- The single instance version (for stochastic g.d.)

$$\begin{aligned} \mathbf{w} &:= \mathbf{w} - \alpha \frac{\partial \mathcal{L}_n}{\partial \mathbf{w}} \\ &:= \mathbf{w} - \alpha \left(t_n - \mathbf{w}^\top \mathbf{x}_n \right) \mathbf{x}_n \end{aligned}$$

(Update per **w** vector element:

$$w_j \quad := \quad w_j - \alpha \left(t_n - \mathbf{w}^\top \mathbf{x}_n \right) x_{n,j} \quad)$$

- The "algorithm":

```
repeat
    for $n = 1$ to $N$ do
        $\mathbf{w} := \mathbf{w} - \alpha \left( t_n - \mathbf{w}^\top \mathbf{x}_n \right) \mathbf{x}_n$
    end
until convergence;
```

Incremental / Stochastic Gradient Descent

- Often, stochastic gradient descent gets **w** "close" to the minimum much faster than batch gradient descent.
- NOTE: however, it may never "converge" to the minimum, and the parameters **w** will keep oscillating around the minimum of the *Loss*
- But in practice, most of the value near the minimum will be reasonably good approximations
- It is more common to run stochastic gradient descent with a fixed learning rate (alpha). However, theoretically, by slowly decreasing the learning rate to zero at the right rate, it is possible to ensure that the parameters will converge to the global minimum rather than merely oscillate around the minimum.

Isaac Joseph

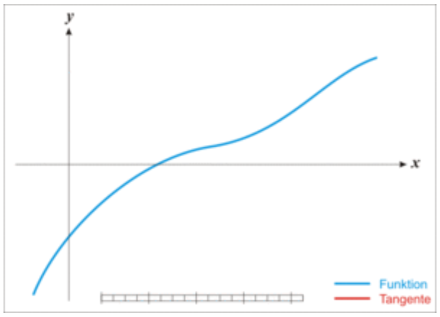
The Newton-Raphson Method

1642–1727 1648–1715

- Method for finding points where functions are equal to zero (e.g., ***roots*** of a polynomial)
- Given a current estimate of the zero point, x_n, find derivative at that point (give slope), find intersection of sloping line, then move in that direction:

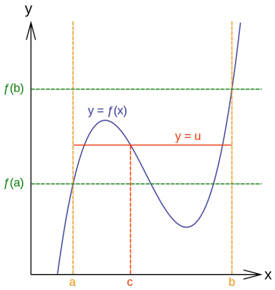
$$f'(x_n) = \frac{\Delta y}{\Delta x} = \frac{f(x_n) - 0}{x_n - x_{n+1}}.$$

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



Why does this work?
The **intermediate value theorem**!!

If *f* : [*a*,*b*] → **R** is continuous,
u is real and *f*(*a*) > *u* > f(*b*),
then ∃ *c*∈(*a*,*b*), *f*(*c*) = *u*



The Newton-Raphson Method

- But Newton's method can do more!
- Extend to find minima and maxima. These are simply points where the gradient itself passes through zero.
- So… replace *f* with *f'* and *f'* with *f''*

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

Using Newton-Raphson for MAP

$$p(\mathbf{t}|\mathbf{X},\mathbf{w}) = \prod_{n=1}^{N} \left(\frac{1}{1+\exp(-\mathbf{w}^\top \mathbf{x}_n)}\right)^{t_n} \left(\frac{\exp(-\mathbf{w}^\top \mathbf{x}_n)}{1+\exp(-\mathbf{w}^\top \mathbf{x}_n)}\right)^{1-t_n} \qquad p(\mathbf{w}|\sigma^2) = \mathcal{N}(0,\sigma^2\mathbf{I})$$

$$\log g(\mathbf{w};\mathbf{X},\mathbf{t}) = \log p(\mathbf{t}|\mathbf{X},\mathbf{w}) + \log p(\mathbf{w}|\sigma^2)$$

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

$$\mathbf{w}' = \mathbf{w} - \left(\frac{\partial^2 \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial\mathbf{w}\partial\mathbf{w}^\mathsf{T}}\right)^{-1} \frac{\partial \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial\mathbf{w}}$$

Point is a **global maximum** if Hessian is negative definite
(as was the case with max likelihood)

Take the Derivatives…

$$\mathbf{w}' = \mathbf{w} - \left(\frac{\partial^2 \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial\mathbf{w}\partial\mathbf{w}^\mathsf{T}}\right)^{-1} \frac{\partial \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial\mathbf{w}}$$

$$P_n = P(T_n = 1|\mathbf{w},\mathbf{x}_n) = \frac{1}{1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n)}$$

Recall the chain rule:

$$\frac{\partial f(g(\mathbf{w}))}{\partial\mathbf{w}} = \frac{\partial f(g(\mathbf{w}))}{\partial g(\mathbf{w})}\frac{\partial g(\mathbf{w})}{\partial\mathbf{w}}$$

$$\begin{aligned}
\log g(\mathbf{w};\mathbf{X},\mathbf{t}) &= \sum_{n=1}^{N} \log P(T_n = t_n|\mathbf{x}_n,\mathbf{w}) + \log p(\mathbf{w}|\sigma^2) \\
&= \sum_{n=1}^{N} \log \left(\frac{1}{1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n)}\right)^{t_n} \left(\frac{\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n)}{1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n)}\right)^{1-t_n} \\
&\quad + \log p(\mathbf{w}|\sigma^2)
\end{aligned}$$

$$= \log p(\mathbf{w}|\sigma^2) + \sum_{n=1}^{N} \log P_n^{t_n} + \log(1-P_n)^{1-t_n}$$

$$= -\frac{D}{2}\log 2\pi - D\log\sigma - \frac{1}{2\sigma^2}\mathbf{w}^\mathsf{T}\mathbf{w}$$ ← Log of Gaussian prior

$$+ \sum_{n=1}^{N} t_n \log P_n + (1-t_n)\log(1-P_n),$$

$$\mathbf{w}' = \mathbf{w} - \left(\frac{\partial^2 \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial \mathbf{w}\partial \mathbf{w}^\mathsf{T}}\right)^{-1} \frac{\partial \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial \mathbf{w}}$$

Take the Derivatives…

$$P_n = P(T_n = 1|\mathbf{w}, \mathbf{x}_n) = \frac{1}{1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n)}$$

Recall the chain rule:

$$\frac{\partial f(g(\mathbf{w}))}{\partial \mathbf{w}} = \frac{\partial f(g(\mathbf{w}))}{\partial g(\mathbf{w})}\frac{\partial g(\mathbf{w})}{\partial \mathbf{w}}$$

$$\log g(\mathbf{w};\mathbf{X},\mathbf{t}) = -\frac{D}{2}\log 2\pi - D\log\sigma - \frac{1}{2\sigma^2}\mathbf{w}^\mathsf{T}\mathbf{w} + \sum_{n=1}^{N} t_n \log P_n + (1-t_n)\log(1-P_n)$$

$$\frac{\partial \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial \mathbf{w}} = -\frac{1}{\sigma^2}\mathbf{w} + \sum_{n=1}^{N}\left(\frac{t_n}{P_n}\frac{\partial P_n}{\partial \mathbf{w}} + \frac{1-t_n}{1-P_n}\frac{\partial(1-P_n)}{\partial \mathbf{w}}\right)$$

$$= -\frac{1}{\sigma^2}\mathbf{w} + \sum_{n=1}^{N}\left(\frac{t_n}{P_n}\boxed{\frac{\partial P_n}{\partial \mathbf{w}}} - \frac{1-t_n}{1-P_n}\boxed{\frac{\partial P_n}{\partial \mathbf{w}}}\right),$$

Apply chain rule again…

$$\frac{\partial(1-P_n)}{\partial \mathbf{w}} = \frac{\partial(1-P_n)}{\partial P_n}\frac{\partial P_n}{\partial \mathbf{w}} = -\frac{\partial P_n}{\partial \mathbf{w}}.$$

$$\frac{\partial P_n}{\partial \mathbf{w}} = \frac{\partial\,(1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n))^{-1}}{\partial\,(1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n))}\frac{\partial\,(1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n))}{\partial \mathbf{w}}$$

$$= -\frac{1}{(1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n))^2}\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n)(-\mathbf{x}_n)$$

$$= \frac{\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n)}{(1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n))^2}\mathbf{x}_n$$

$$= \frac{1}{1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n)}\frac{\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n)}{1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x})}\mathbf{x}_n$$

$$= P_n(1-P_n)\mathbf{x}_n.$$

Finally, plug $\frac{\partial P_n}{\partial \mathbf{w}}$ back in:

$$\frac{\partial \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial \mathbf{w}} = -\frac{1}{\sigma^2}\mathbf{w} + \sum_{n=1}^{N}(\mathbf{x}_n t_n(1-P_n) - \mathbf{x}_n(1-t_n)P_n)$$

$$= -\frac{1}{\sigma^2}\mathbf{w} + \sum_{n=1}^{N}\mathbf{x}_n(t_n - t_nP_n - P_n + t_nP_n)$$

$$= -\frac{1}{\sigma^2}\mathbf{w} + \sum_{n=1}^{N}\mathbf{x}_n(t_n - P_n).$$

$$\mathbf{w}' = \mathbf{w} - \left(\frac{\partial^2 \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial \mathbf{w}\partial \mathbf{w}^\mathsf{T}}\right)^{-1} \frac{\partial \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial \mathbf{w}}$$

Take the Derivatives…

$$P_n = P(T_n = 1|\mathbf{w}, \mathbf{x}_n) = \frac{1}{1+\exp(-\mathbf{w}^\mathsf{T}\mathbf{x}_n)}$$

Recall the chain rule:

$$\frac{\partial f(g(\mathbf{w}))}{\partial \mathbf{w}} = \frac{\partial f(g(\mathbf{w}))}{\partial g(\mathbf{w})}\frac{\partial g(\mathbf{w})}{\partial \mathbf{w}}$$

$$\frac{\partial \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial \mathbf{w}} = -\frac{1}{\sigma^2}\mathbf{w} + \sum_{n=1}^{N}\mathbf{x}_n(t_n - P_n)$$

Now, for the Hessian:

$$\frac{\partial^2 \log g(\mathbf{w};\mathbf{X},\mathbf{t})}{\partial \mathbf{w}\partial \mathbf{w}^\mathsf{T}} = -\frac{1}{\sigma^2}\mathbf{I} - \sum_{n=1}^{N}\mathbf{x}_n\frac{\partial P_n}{\partial \mathbf{w}^\mathsf{T}}$$

$$\frac{\partial P_n}{\partial \mathbf{w}} = P_n(1-P_n)\mathbf{x}_n$$

$$\frac{\partial P_n}{\partial \mathbf{w}^\mathsf{T}} = \left(\frac{\partial P_n}{\partial \mathbf{w}}\right)^\mathsf{T}$$

$0 \leq P_n \leq 1$ So, Hessian is negative definite!

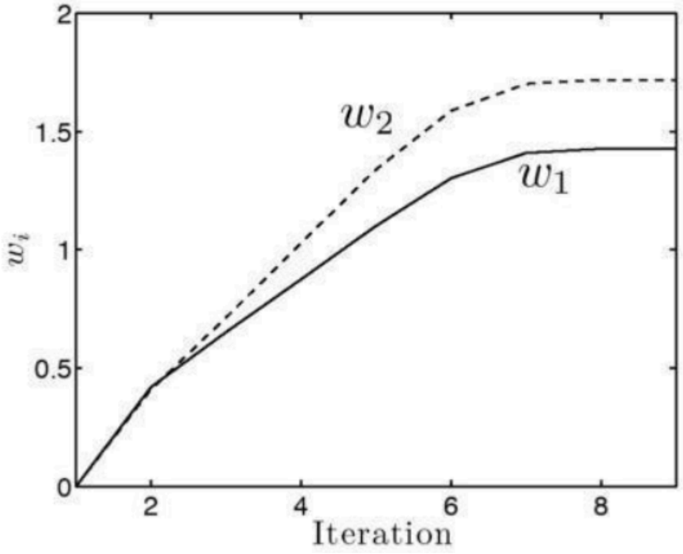
There can only be one optimum, must be a maximum.
Whatever value of **w** the Newton-Raphson procedure converges to must correspond to the highest value of the posterior density.
This is a choice of our particular prior and likelihood
Changing either may result in harder posterior density to optimize

Estimating w

Starting from:

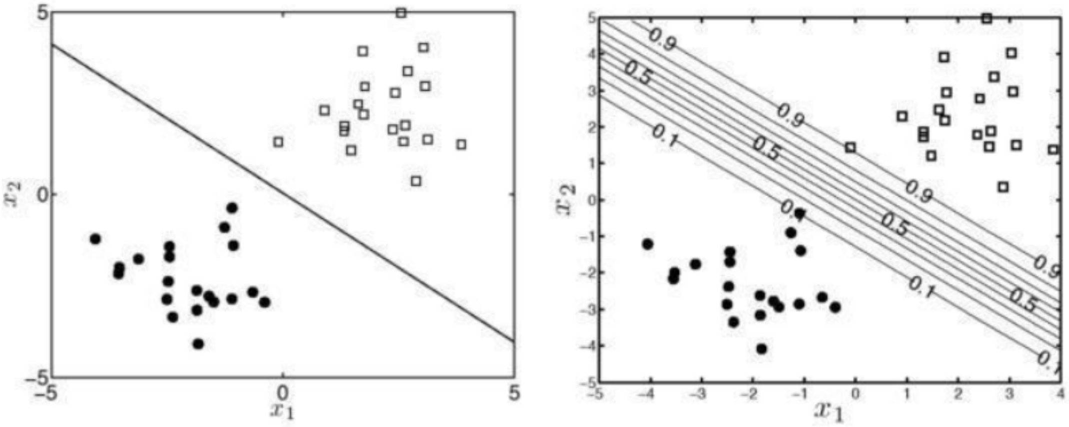
$$\mathbf{w} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\sigma^2 = 10$$



Using w to compute prob of response

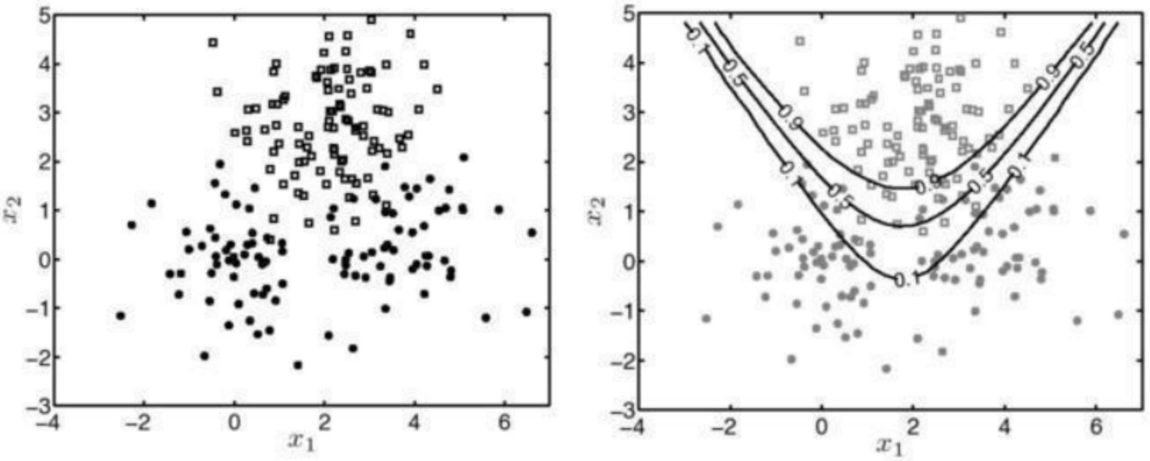
$$P(T_{\text{new}} = 1 | \mathbf{x}_{\text{new}}, \widehat{\mathbf{w}}) = \frac{1}{1 + \exp(-\widehat{\mathbf{w}}^{\mathsf{T}} \mathbf{x}_{\text{new}})}$$



Nonlinear Decision Functions

$$\log\left(\frac{P(T_{\text{new}}=1|\mathbf{x}_{\text{new}},\mathbf{w})}{P(T_{\text{new}}=0|\mathbf{x}_{\text{new}},\mathbf{w})}\right) = w_0 + w_1x_1 + w_2x_2 + w_3x_1^2 + w_4x_2^2$$

Find $\widehat{\mathbf{w}}$ by MAP



x_1
x_2
−4
−2
0
2
4
6
−3
−2
−1
0
1
2
3
4
5
0.1
0.5
0.9