Lecture Three
Supervised Learning: Classification

KMA Solaiman
Fall 2023

Adapted From
Chris Re′
Stanford ML

Supervised Learning and Classification

- Linear Regression via a Probabilistic Interpretation
- Logistic Regression
- Optimization Method: Newton's Method

> We'll learn the maximum likelihood method (a probabilistic interpretation) to generalize from linear regression to more sophisticated models.

A Justification for Least Squares?

- **Given** a training set $\{(x^{(i)}, y^{(i)})$ for $i = 1, \dots, n\}$ in which $x^{(i)} \in \mathbb{R}^{d+1}$ and $y^{(i)} \in \mathbb{R}$.
- **Do** find $\theta \in \mathbb{R}^{d+1}$ s.t. $\theta = \operatorname{argmin}_\theta \sum_{i=1}^{n} (h_\theta(x^{(i)}) - y^{(i)})^2$ in which $h_\theta(x) = \theta^T x$.

A Justification for Least Squares?

- **Given** a training set $\{(x^{(i)}, y^{(i)})$ for $i = 1, \ldots, n\}$ in which $x^{(i)} \in \mathbb{R}^{d+1}$ and $y^{(i)} \in \mathbb{R}$.
- **Do** find $\theta \in \mathbb{R}^{d+1}$ s.t. $\theta = \text{argmin}_\theta \sum_{i=1}^{n} (h_\theta(x^{(i)}) - y^{(i)})^2$ in which $h_\theta(x) = \theta^T x$.

Where did this model come from?

One way to view is via a *probabilistic interpretation (helpful throughout the course)*.

A Justification for Least Squares?

We make an assumption (common in statistics) that the data are *generated* according to some model (that may contain random choices). That is,

$$y^{(i)} = \theta^T x^{(i)} + \varepsilon^{(i)}.$$

Here, $\varepsilon^{(i)}$ is a random variable that captures "noise" that is, unmodeled effects, measurement errors, etc.

A Justification for Least Squares?

We make an assumption (common in statistics) that **the data are *generated* according to some model** (that may contain random choices). That is,

$$y^{(i)} = \theta^T x^{(i)} + \varepsilon^{(i)}.$$

Here, $\varepsilon^{(i)}$ is a random variable that captures "noise" that is, unmodeled effects, measurement errors, etc.

Please keep in mind: this is just a model! As they say, all models are wrong but some models are *useful.* This model has been *shockingly* useful.

What do we expect of the noise?

What properties should we expect from $\varepsilon^{(i)}$

$$y^{(i)} = \theta^T x^{(i)} + \varepsilon^{(i)}.$$

Again, it's a model and $\varepsilon^{(i)}$ is a random variable:

- $\mathbb{E}[\varepsilon^{(i)}] = 0$ – the noise is unbiased.

What do we expect of the noise?

What properties should we expect from $\varepsilon^{(i)}$

$$y^{(i)} = \theta^T x^{(i)} + \varepsilon^{(i)}.$$

Again, it's a model and $\varepsilon^{(i)}$ is a random variable:

- $\mathbb{E}[\varepsilon^{(i)}] = 0$ – the noise is unbiased.
- The errors for different points are *independent* and *identically distributed* (called, **iid**)

$$\mathbb{E}[\varepsilon^{(i)}\varepsilon^{(j)}] = \mathbb{E}[\varepsilon^{(i)}]\mathbb{E}[\varepsilon^{(j)}] \text{ for } i \neq j.$$

and

$$\mathbb{E}\left[\left(\varepsilon^{(i)}\right)^2\right] = \sigma^2$$

Here σ^2 is some measure of *how noisy* the data are.

What do we expect of the noise?

What properties should we expect from $\varepsilon^{(i)}$

$$y^{(i)} = \theta^T x^{(i)} + \varepsilon^{(i)}.$$

Again, it's a model and $\varepsilon^{(i)}$ is a random variable:

- $\mathbb{E}[\varepsilon^{(i)}] = 0$ – the noise is unbiased.
- The errors for different points are *independent* and *identically distributed* (called, **iid**)

$$\mathbb{E}[\varepsilon^{(i)}\varepsilon^{(j)}] = \mathbb{E}[\varepsilon^{(i)}]\mathbb{E}[\varepsilon^{(j)}] \text{ for } i \neq j.$$

and

$$\mathbb{E}\left[\left(\varepsilon^{(i)}\right)^2\right] = \sigma^2$$

Here σ^2 is some measure of *how noisy* the data are. Turns out, this effectively defines the *Gaussian or Normal distribution.*

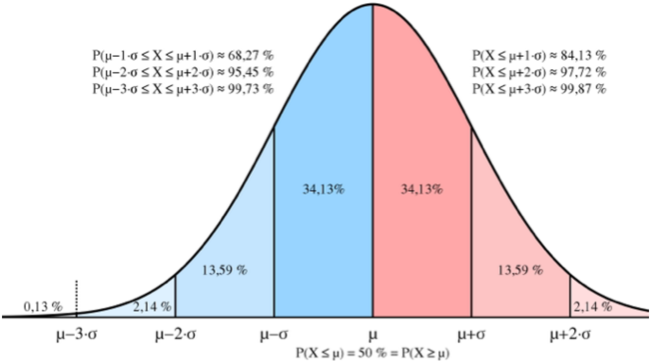
Notation for the Gaussian

We write $z \sim \mathcal{N}(\mu, \sigma^2)$ and read these symbols as

> *z is distributed as a normal with mean μ and standard deviation σ^2.*

or equivalently the probability density function -

$$P(z) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(z-\mu)^2}{2\sigma^2}\right\} \quad \text{.............} \; (10.1)$$



Notation for Guassians **in our Problem**

Recall in our model,

$$y^{(i)} = \theta^T x^{(i)} + \varepsilon^{(i)} \text{ in which } \varepsilon^{(i)} \sim \mathcal{N}(0, \sigma^2) \text{ .... } (11.1)$$

or more compactly notation:

$$y^{(i)} \mid x^{(i)}; \theta \sim \mathcal{N}(\theta^T x, \sigma^2) \text{.............. } (11.2)$$

equivalently, **Probability distribution** over $y^{(i)}$, given $x^{(i)}$ and parameterized by θ

$$P\left(y^{(i)} \mid x^{(i)}; \theta\right) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(y^{(i)} - x^{(i)}\theta)^2}{2\sigma^2}\right\} \text{ ...... } (11.3)$$

- We **condition** on $x^{(i)}$.
- In contrast, θ **parameterizes** or "picks" a distribution.

We use bar (|) versus semicolon (;) notation above.

How did we calculate Probability Distribution of $y^{(i)}$ in 11.3?

Using our $error\ term$ in place of z, we get

$$\frac{1}{\sigma\sqrt{2\pi}} exp\left\{-\frac{\left(\varepsilon^{(i)}\ - 0\right)^2}{2\sigma^2}\right\}$$

Now if we replace this with values from 11.1, we get

$$\frac{1}{\sigma\sqrt{2\pi}} exp\left\{-\frac{\left(y^{(i)}\ - x^{(i)}\theta\right)^2}{2\sigma^2}\right\}$$

This term gives us the probability distribution over $y^{(i)}$, but we must add $x^{(i)}$ as a given, since we will see it as input, so by fiat we consider this as, $P(y^{(i)}|x^{(i)};\theta)$ which is not conditioned on θ, as it isn't Random Variable

(Log) Likelihoods!

Intuition: among many distributions, pick the one that agrees with the data the most (is most "likely").

$$L(\theta) = p(y|X;\theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)};\theta) \qquad \text{iid assumption}$$

(Log) Likelihoods!

Intuition: among many distributions, pick the one that agrees with the data the most (is most "likely").

$$L(\theta) = p(y|X;\theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)};\theta) \qquad \text{iid assumption}$$

$$= \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(x^{(i)}\theta - y^{(i)})^2}{2\sigma^2}\right\}$$

(Log) Likelihoods!

Intuition: among many distributions, pick the one that agrees with the data the most (is most "likely").

$$L(\theta) = p(y|X;\theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)};\theta) \qquad \text{iid assumption}$$

$$= \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{ -\frac{(x^{(i)}\theta - y^{(i)})^2}{2\sigma^2} \right\}$$

For convenience, we use the *Log Likelihood* $\ell(\theta) = \log L(\theta)$.

$$\ell(\theta) = \sum_{i=1}^{n} \log \frac{1}{\sigma\sqrt{2\pi}} - \frac{(x^{(i)}\theta - y^{(i)})^2}{2\sigma^2}$$

(Log) Likelihoods!

Intuition: among many distributions, pick the one that agrees with the data the most (is most "likely").

$$L(\theta) = p(y|X;\theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)};\theta) \qquad \text{iid assumption}$$

$$= \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{ -\frac{(x^{(i)}\theta - y^{(i)})^2}{2\sigma^2} \right\}$$

For convenience, we use the *Log Likelihood* $\ell(\theta) = \log L(\theta)$.

$$\ell(\theta) = \sum_{i=1}^{n} \log \frac{1}{\sigma\sqrt{2\pi}} - \frac{(x^{(i)}\theta - y^{(i)})^2}{2\sigma^2}$$

$$= n \log \frac{1}{\sigma\sqrt{2\pi}} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (x^{(i)}\theta - y^{(i)})^2 = C(\sigma, n) - \frac{1}{\sigma^2} J(\theta)$$

where $C(\sigma, n) = n \log \frac{1}{\sigma\sqrt{2\pi}}$.

(Log) Likelihoods!

So we've shown that finding a θ to maximize $L(\theta)$ is the same as *maximizing*

$$\ell(\theta) = C(\sigma, n) - \frac{1}{\sigma^2} J(\theta)$$

Or minimizing, $J(\theta)$ directly (why?)

Takeaway: "Under the hood," solving least squares *is* solving a maximum likelihood problem for a particular probabilistic model.

This view shows a path to generalize to new situations!

Summary of Least Squares

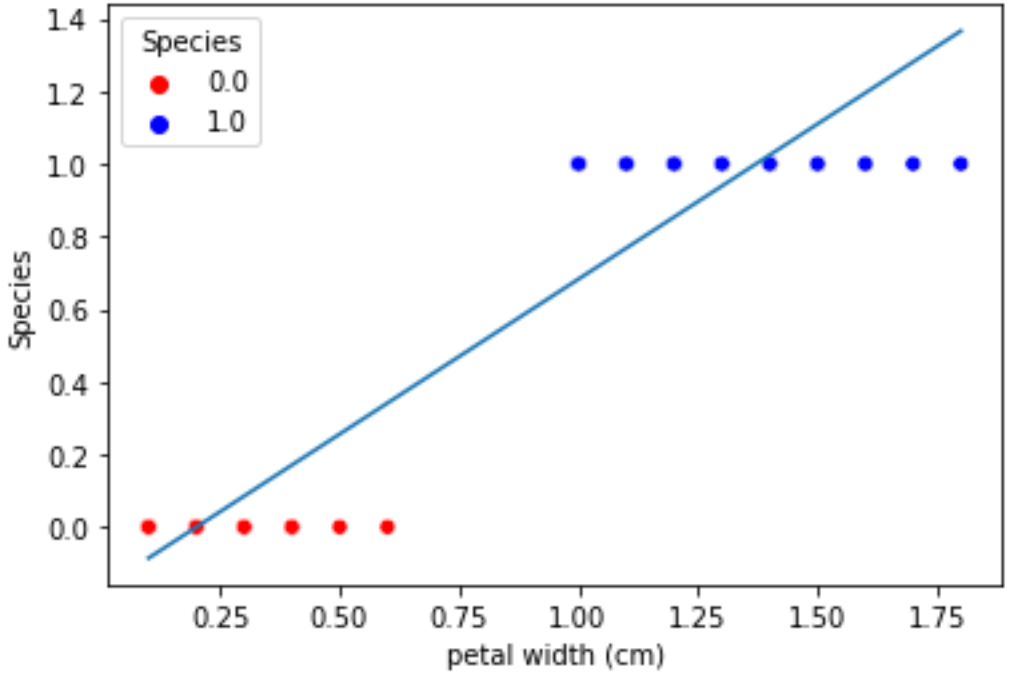
- We introduced the Maximum Likelihood framework–super powerful (next lectures)
- We showed that least squares was actually a version of maximum likelihoods.
- We learned some notation that will help us later in the course. . .

Classification

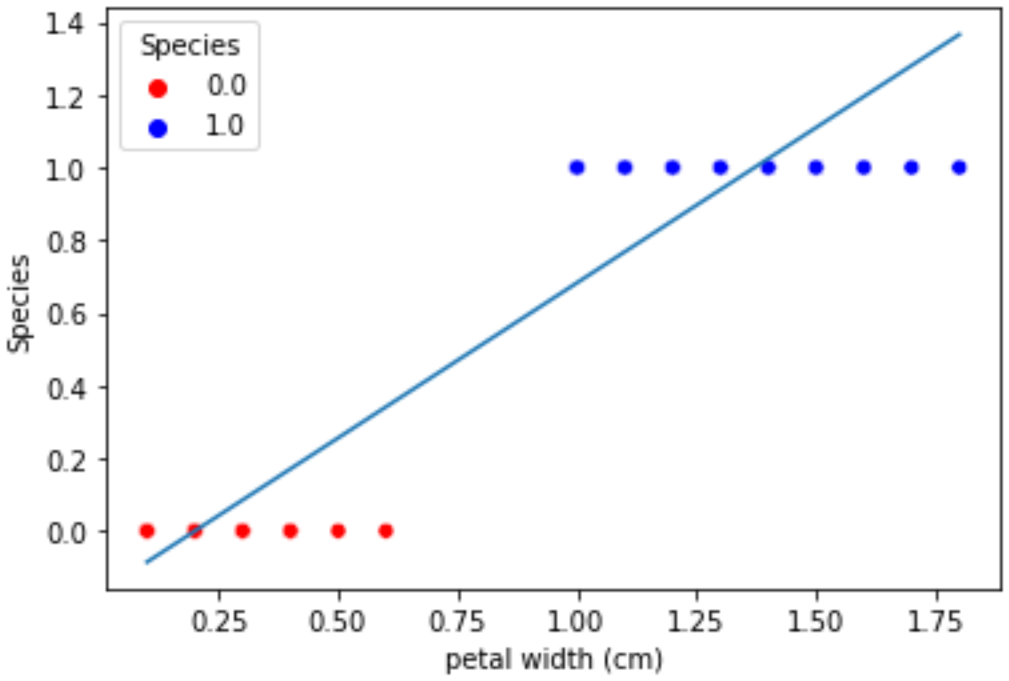
Given a training set $\{(x^{(i)}, y^{(i)})$ for $i = 1, \ldots, n\}$ let $y^{(i)} \in \{0, 1\}$.
Why not use regression, say least squares? A picture ...

	petal width (cm)	Species
0	0.2	0.0
1	0.2	0.0
2	0.2	0.0
3	0.2	0.0
4	0.2	0.0
...	...	...
95	1.2	1.0
96	1.3	1.0
97	1.3	1.0
98	1.1	1.0
99	1.3	1.0

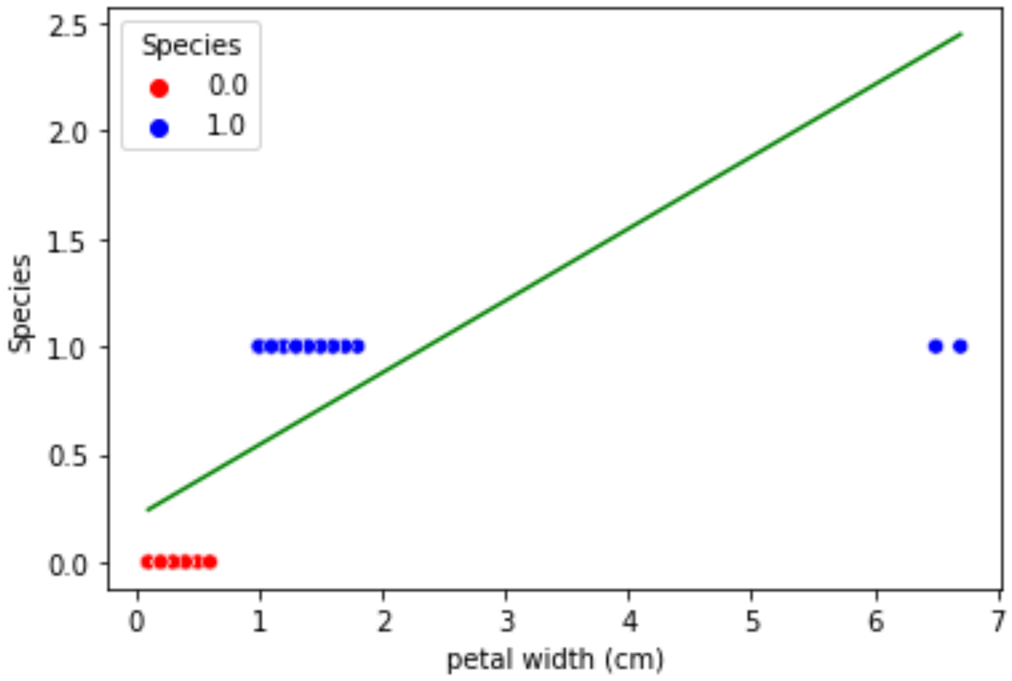
Iris Flower Dataset



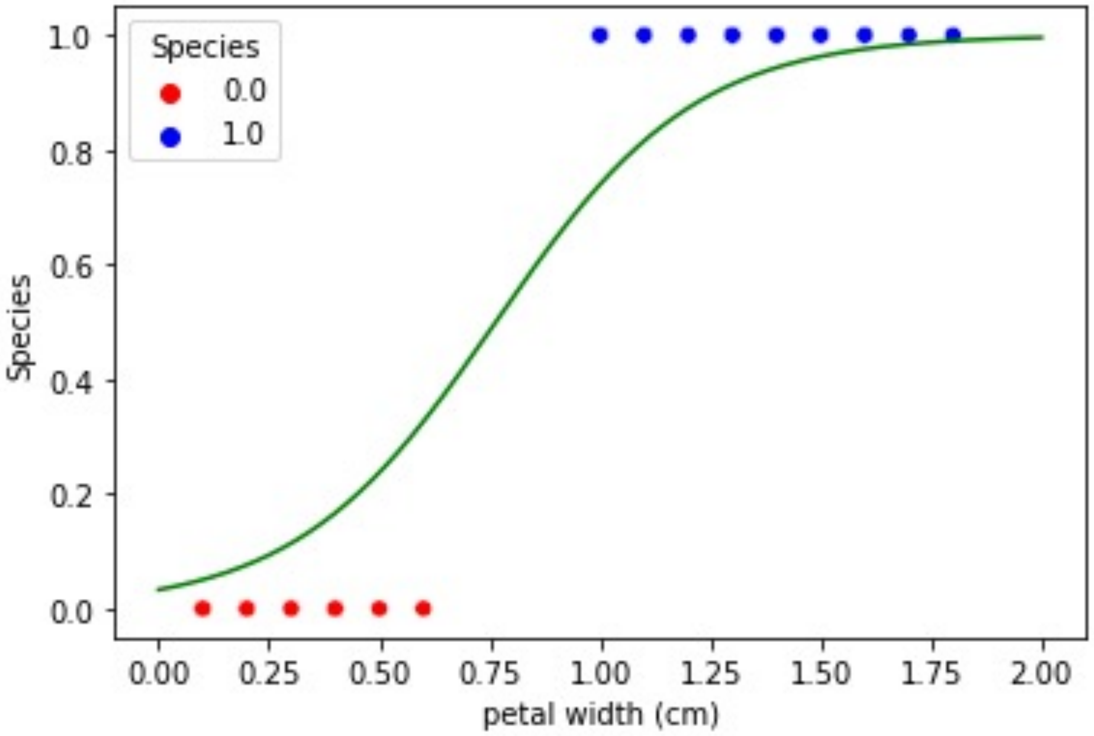
Graph of Iris Dataset with linear regression



Graph of Iris Dataset with linear regression



Graph of Iris Dataset(with outliers) with linear regression



Graph of Iris Dataset with logistic regression

Logistic Regression: Link Functions

Given a training set $\{(x^{(i)}, y^{(i)}) \text{ for } i = 1, \ldots, n\}$ let $y^{(i)} \in \{0, 1\}$. Want $h_\theta(x) \in [0, 1]$. Let's pick a smooth function:

$$h_\theta(x) = g(\theta^T x)$$

Here, g is a link function. There are *many*...

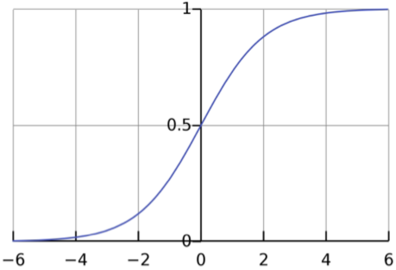
Logistic Regression: Link Functions

Given a training set $\{(x^{(i)}, y^{(i)})$ for $i = 1, \ldots, n\}$ let $y^{(i)} \in \{0, 1\}$. Want $h_\theta(x) \in [0, 1]$. Let's pick a smooth function:

$$h_\theta(x) = g(\theta^T x)$$

Here, g is a link function. There are *many*... but we'll pick one!

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



Logistic Regression: Link Functions

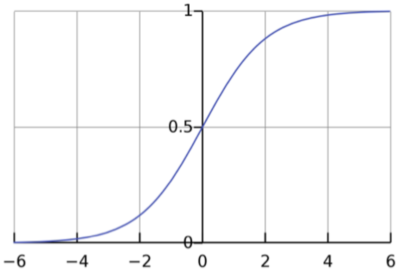
Given a training set $\{(x^{(i)}, y^{(i)}) \text{ for } i = 1, \ldots, n\}$ let $y^{(i)} \in \{0, 1\}$.
Want $h_\theta(x) \in [0, 1]$. Let's pick a smooth function:

$$h_\theta(x) = g(\theta^T x)$$

Here, g is a link function. There are *many*... but we'll pick one!

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

SIGMOID



How do we interpret $h_\theta(x)$?

$$P(y = 1 \mid x; \theta) = h_\theta(x)$$
$$P(y = 0 \mid x; \theta) = 1 - h_\theta(x)$$

Logistic Regression: Link Functions

Let's write the Likelihood function. Recall:

$$P(y = 1 \mid x; \theta) = h_\theta(x)$$
$$P(y = 0 \mid x; \theta) = 1 - h_\theta(x)$$

Then,

$$L(\theta) = P(y \mid X; \theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; \theta)$$

Logistic Regression: Link Functions

Let's write the Likelihood function. Recall:

$$
\begin{aligned}
P(y = 1 \mid x; \theta) &= h_\theta(x) \\
P(y = 0 \mid x; \theta) &= 1 - h_\theta(x)
\end{aligned}
$$

Then,

$$L(\theta) = P(y \mid X; \theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; \theta)$$

How do we go to a cost function from P (y I X; θ) ?

We need to go back to Maximum Likelihood Estimation that we saw before at the beginning of this lecture.

Logistic Regression: Link Functions

Let's write the Likelihood function. Recall:

$$\begin{aligned} P(y = 1 \mid x; \theta) =& h_\theta(x) \\ P(y = 0 \mid x; \theta) =& 1 - h_\theta(x) \end{aligned}$$

Then,

$$\begin{aligned} L(\theta) =& P(y \mid X; \theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; \theta) \\ =& \prod_{i=1}^{n} h_\theta(x^{(i)})^{y^{(i)}} (1 - h_\theta(x^{(i)}))^{1-y^{(i)}} \quad \text{exponents encode "if-then"} \end{aligned}$$

Logistic Regression: Link Functions

Let's write the Likelihood function. Recall:

$$
\begin{aligned}
P(y = 1 \mid x; \theta) =& h_\theta(x) \\
P(y = 0 \mid x; \theta) =& 1 - h_\theta(x)
\end{aligned}
$$

Then,

$$
\begin{aligned}
L(\theta) =& P(y \mid X; \theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; \theta) \\
=& \prod_{i=1}^{n} h_\theta(x^{(i)})^{y^{(i)}} (1 - h_\theta(x^{(i)}))^{1-y^{(i)}} \quad \text{exponents encode “if-then”}
\end{aligned}
$$

Taking logs to compute the log likelihood $\ell(\theta)$ we have:

$$
\ell(\theta) = \log L(\theta) = \sum_{i=1}^{n} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))
$$

Now to solve it...

$$\ell(\theta) = \log L(\theta) = \sum_{i=1}^{n} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))$$

We maximize for θ but we already saw how to do this! Just compute derivative, run (S)GD and you're done with it!

Takeaway: This is *another* example of the max likelihood method: we setup the likelihood, take logs, and compute derivatives.

Time Permitting: There is magic in the derivative...

Even more, the batch update can be written in a *remarkably familiar* form:

$$\theta^{(t+1)} = \theta^{(t)} + \sum_{j \in B} (y^{(j)} - h_\theta(x^{(j)}))x^{(j)}.$$

We sketch why (you can check!) We drop superscripts to simplify notation and examine a single data point:

$$\begin{aligned} & y \log h_\theta(x) + (1-y)\log(1-h_\theta(x)) \\ = & -y\log(1+e^{-\theta^T x}) + (1-y)(-\theta^T x) - (1-y)\log(1+e^{-\theta^T x}) \\ = & -\log(1+e^{-\theta^T x}) - (1-y)(\theta^T x) \end{aligned}$$

We used $1 - h_\theta(x) = \frac{e^{-\theta^T x}}{1-e^{-\theta^T x}}$. We now compute the derivative of this expression wrt θ and get:

$$\frac{e^{-\theta^T x}}{1+e^{-\theta^T x}} x - (1-y)x = (y - h_\theta(x))x$$

Perceptron Learning Algorithm

- Modify link function to output either 0 or 1.
- Make g to be a threshold function
- Then use same $h_\theta(x) = g(\theta^T x)$ using this g
- Follow the same update rule for θ

$$g(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0 \end{cases}$$

Summary of Introduction to Classification

- We used the principle of maximum likelihood (and a probabilistic model) to extend to classification.

Summary of Introduction to Classification

- We used the principle of maximum likelihood (and a probabilistic model) to extend to classification.
- We developed logistic regression from this principle.
  - Logistic regression is *widely* used today.

Summary of Introduction to Classification

- We used the principle of maximum likelihood (and a probabilistic model) to extend to classification.
- We developed logistic regression from this principle.
  - Logistic regression is *widely* used today.
- We noticed a familiar pattern: take derivatives of the likelihood, and the derivatives had this (hopefully) intuitive *"misprediction form"*

Newton's Method

Given $f : \mathbb{R}^d \to \mathbb{R}$ find x s.t. $f(x) = 0$.

Newton's Method

Given $f : \mathbb{R}^d \to \mathbb{R}$ find x s.t. $f(x) = 0$.

We apply this with $f(\theta) = \nabla_\theta \ell(\theta)$, the likelihood function

Newton's Method (Drawn in Class)

Given $f : \mathbb{R}^d \to \mathbb{R}$ find x s.t. $f(x) = 0$.

Newton's Method Summary

Given $f : \mathbb{R}^d \to \mathbb{R}$ find x s.t. $f(x) = 0$.

- This is the update rule in 1d

$$x^{(t+1)} = x^{(t)} - \frac{f(x^{(t)})}{f'(x^{(t)})}$$

- It may converge *very* fast (quadratic local convergence!)
- For the likelihood, i.e., $f(\theta) = \nabla_\theta \ell(\theta)$ we need to generalize to a vector-valued function which has:

$$\theta^{(t+1)} = \theta^{(t)} - \left(H(\theta^{(t)})\right)^{-1} \nabla_\theta \ell(\theta^{(t)}).$$

in which $H_{i,j}(\theta) = \frac{\partial}{\partial\theta_i \partial\theta_j} \ell(\theta)$.

Optimization Method Summary

Method	Compute per Step	Number of Steps to convergence
SGD	$\theta(d)$	$\approx \epsilon^{-2}$
Minibatch SGD		
GD	$\theta(nd)$	$\approx \epsilon^{-1}$
Newton	$\Omega(nd^2)$	$\approx \log(1/\epsilon)$

- In classical stats, d is small (< 100), n is often small, and *exact parameters matter*
- In modern ML, d is huge (billions, trillions), n is huge (trillions), and parameters used *only* for prediction
  - These are approximate number of computing steps
  - Convergence happens when loss settles to within an error range around the final value.
  - Newton would be very fast, where SGD needs a lot of step, but individual steps are fast, makes up for it
- As a result, (minibatch) SGD is the *workhorse* of ML.

Classification Lecture Summary

- We saw the differences between classification and regression.
- We learned about a principle for probabilistic interpretation for linear regression and classification: **Maximum Likelihood**.
  - We used this to derive logistic regression.
  - The Maximum Likelihood principle will be used again next lecture (and in the future)
- We saw Newton's method, which is classically used models (more statistics than ML–it's not used in most modern ML)