

CS5014 Machine Learning

Lecture 4 Maximum Likelihood Estimation (MLE)

Lei Fang

School of Computer Science, University of St Andrews

Spring 2021



University
of
St Andrews

Motivation

Objective: **probabilistic perspective** of linear regression

- justify least squared error: $(\mathbf{y} - \mathbf{X}\boldsymbol{\theta})^T (\mathbf{y} - \mathbf{X}\boldsymbol{\theta})$
- maximum likelihood estimator: $\boldsymbol{\theta}_{\text{ML}}$
- BUT nothing new: $\boldsymbol{\theta}_{\text{ML}} = \boldsymbol{\theta}_{\text{LS}} = (\mathbf{X}\mathbf{X})^{-1}\mathbf{X}\mathbf{y}$

So why bother ?

- MLE: very general model
- lots of ML algorithms fit in MLE category
 - linear regression, logistic regression, k-means, mixture model, neural nets, discriminant analysis, naive Bayes ...
- large number theory for MLE (next time)
 - $P(\boldsymbol{\theta}_{\text{ML}})$? or *sampling distribution*
 - does $\boldsymbol{\theta}_{\text{ML}}$ change much given another $\mathcal{D}_k = \{\mathbf{X}_k, \mathbf{y}_k\}$?

Motivation

Objective: **probabilistic perspective** of linear regression

- justify least squared error: $(\mathbf{y} - \mathbf{X}\boldsymbol{\theta})^T (\mathbf{y} - \mathbf{X}\boldsymbol{\theta})$
- maximum likelihood estimator: $\boldsymbol{\theta}_{\text{ML}}$
- BUT nothing new: $\boldsymbol{\theta}_{\text{ML}} = \boldsymbol{\theta}_{\text{LS}} = (\mathbf{X}\mathbf{X})^{-1} \mathbf{X}\mathbf{y}$

So why bother ?

- MLE: very general model
- lots of ML algorithms fit in MLE category
 - linear regression, logistic regression, k-means, mixture model, neural nets, discriminant analysis, naive Bayes ...
- large number theory for MLE (next time)
 - $P(\boldsymbol{\theta}_{\text{ML}})$? or *sampling distribution*
 - does $\boldsymbol{\theta}_{\text{ML}}$ change much given another $\mathcal{D}_k = \{\mathbf{X}_k, \mathbf{y}_k\}$?

Topics of today

Review of probability theory

- univariate Gaussian

Maximum likelihood estimation in general

- MLE for Gaussian
- MLE for Bernoulli/Binomial

Linear regression revisit: MLE

Logistic regression and MLE

Review: Random variable

Random variable X

- opposite to deterministic variable: X can take a range of value associated with some probability $P(X)$
- discrete r.v.: if X can only take discrete values
 - e.g. $X \in \{T, F\}$, $X \in \{1, 2, 3, \dots\}$ etc.
- otherwise X is continuous r.v.
 - e.g. $X \in [0, 1]$, $X \in \mathbb{R}^2$

Random variable - discrete r.v.

If r.v. X 's target space \mathcal{T} is discrete

- X is a **discrete random variable**
- the probability distribution P is called **probability mass function** (p.m.f.)
- and

$$0 \leq P(X = x) \leq 1, \text{ and } \sum_{x \in \mathcal{T}} P(X = x) = 1$$

Example - discrete r.v.

Bernoulli distribution Tossing a coin , $\mathcal{T} = 1, 0$ (1 is H, 0 is T),

$$P(X = 1) = p, P(X = 0) = 1 - p, 0 \leq p \leq 1$$

or

$$P(X = x) = p^x(1 - p)^{1-x}$$

Example - discrete r.v.

Multinoulli distribution

X can take $\{1, 2, \dots, k\}$, its probability mass function is

$$P(X) = \begin{cases} p_1 & X = 1 \\ p_2 & X = 2 \\ \vdots & \\ p_k & X = k \end{cases} \quad P(x) = \prod_{i=1}^k p_i^{I(x=i)}$$
$$I(x = i) = 1 \text{ if } x = i \text{ or } 0 \text{ if } x \neq i$$

E.g. throw a fair 6-facet die, $\mathcal{T} = 1, 2, \dots, 6$, the distribution is

$$P(X = i) = 1/6$$

Random variable - continuous r.v.

If r.v. X 's target space \mathcal{T} is continuous

- X is a **continuous random variable**
- the probability distribution p is called **probability density function** (p.d.f.): note we use p
- and satisfies

$$p(x) \geq 0, \text{ and } \int_{x \in \mathcal{T}} p(x) dx = 1$$

- pdf is not probability as $p(x)$ can be greater 1;
- calculate probability over an interval: e.g.

$$0 \leq P(X \in [a, b]) = \int_a^b p(x) dx \leq 1$$

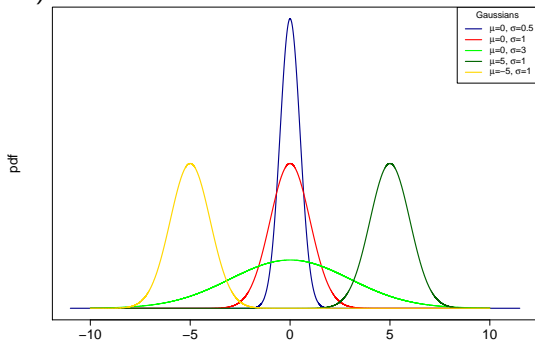
- for $\forall a \in \mathcal{T}$ $P(X = a) = P(X \in [a, a]) = \int_a^a p(x) dx = 0$

Example - continuous r.v.

Gaussian distribution $\mathcal{T} = R$, or $X \in R$ the pdf is

$$p(x) = \mathcal{N}\left(x; \mu, \sigma^2\right) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

$\left(\frac{x-\mu}{\sigma}\right)^2$ is a distance measure: how far x is away from μ (measured by σ as a unit)



Joint distribution

Random variable $\mathbf{X} = [X_1, X_2, \dots, X_n]^T$ can be multidimensional (each X_i is r.v.)

- essentially a *random vector*

Still satisfies the same requirements

$$\forall \mathbf{x}, 0 < P(\mathbf{X} = \mathbf{x}) < 1, \sum_{x_1} \sum_{x_2} \dots \sum_{x_n} P(\mathbf{X} = [x_1, x_2, \dots, x_n]) = 1$$

- means the probability that $\mathbf{X} = \mathbf{x}$ is jointly true

For bivariate case, i.e. $n = 2$, X_1, X_2 are **independent** (e.g. rolling two dice independently) if

$$P(\mathbf{X}) = P(X_1)P(X_2)$$

Example: joint distribution

The joint distribution of X snow or not, $Y \in \{\text{spring, summer, autumn, winter}\}$ represents the season that x belongs to :

	$y = \text{Spring}$	$y = \text{Summer}$	$y = \text{Autumn}$	$y = \text{winter}$
$x = F$	0.05	0.25	0.075	0
$x = T$	0.2	0	0.175	0.25

It is easy to verify that

$$\sum_x \sum_y p(x, y) = 1$$

Probability rules

There are only two probability rules (integration for continuous r.v.):

1. product rule:

$$p(x, y) = p(y|x)p(x) = p(x|y)p(y)$$

2. sum rule (marginalisation):

$$p(x) = \sum_y p(x, y), \quad p(y) = \sum_x p(x, y)$$

Conditional probability

Conditional probability distribution (by product rule):

$$p(x|y) = \frac{p(x, y)}{p(y)}$$

- probability distribution of x conditional on the value of y

	$y = \text{Spring}$	$y = \text{Summer}$	$y = \text{Autumn}$	$y = \text{winter}$
$x = F$	0.05	0.25	0.075	0
$x = T$	0.2	0	0.175	0.25

- $P(Y = \text{Spring})$? use sum rule

$$P(Y = \text{Spring}) = \sum_{x=\{T, F\}} P(X = x, Y = \text{Spring}) = 0.05 + 0.2 = \frac{1}{4}$$

- $P(X = T | Y = \text{Spring})$?

$$P(X = T | y = \text{Spring}) = \frac{P(x=T, y=\text{Spring})}{P(y=\text{Spring})} = \frac{0.2}{0.25} = 0.8$$

Parameter estimation problem

Given dataset $\mathcal{D} = \{y^{(1)}, y^{(2)}, \dots, y^{(m)}\}$, and assume

$$y^{(i)} \sim P(y^{(i)}|\theta), \quad i = 1, \dots, m$$

- *parameter estimation*: given \mathcal{D} , what is θ ?

For example, throw the same coin n times and record value $y^{(i)} \in \{1, 0\}, i = 1, \dots, m$

$$P(y^{(i)}|\theta) = \text{Ber}(\theta)$$

- $y^{(i)} \stackrel{iid}{\sim} \text{Ber}(\theta)$: independent and identically distributed
- θ : the probability that head turns up

Maximum Likelihood Estimation

Likelihood function: $P(\mathcal{D}|\theta) = \prod_i^m p(y^{(i)}|\theta)$

- the probability of observing data \mathcal{D} given θ
- it is not a probability distribution for θ : $\int p(\mathcal{D}|\theta)d\theta \neq 1$
- but it is a function of θ (given \mathcal{D})

Maximum likelihood estimation:

$$\theta_{ML} = \underset{\theta}{\operatorname{argmax}} P(\mathcal{D}|\theta)$$

- the value θ most likely to have generated the data

We usually deal with log-likelihood, denoted as $\mathcal{L}(\theta)$

$$\theta_{ML} = \underset{\theta}{\operatorname{argmax}} \underbrace{\log P(\mathcal{D}|\theta)}_{\mathcal{L}(\theta)} = \underset{\theta}{\operatorname{argmax}} P(\mathcal{D}|\theta)$$

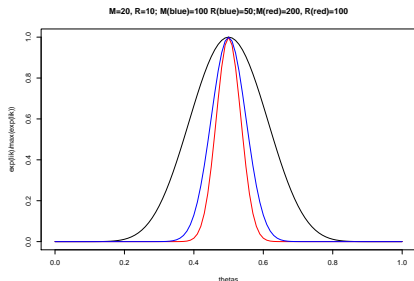
MLE for Bernoulli

For the Bernoulli case: $y^{(i)} \in \{1, 0\}$

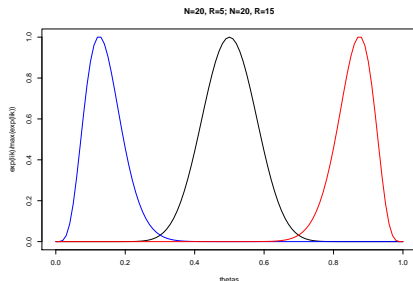
$$\begin{aligned}\mathcal{L}(\theta) &= \log P(\mathcal{D}|\theta) = \log \prod_{i=1}^m P(y^{(i)}; \theta) \\ &= \log \prod_{i=1}^m \theta^{y^{(i)}} (1 - \theta)^{1-y^{(i)}} \\ &= \log(\theta^{\sum_{i=1}^m y^{(i)}} (1 - \theta)^{\sum_{i=1}^m (1-y^{(i)})}) \\ &= \sum_{i=1}^m y^{(i)} \log \theta + (m - \sum_{i=1}^m y^{(i)}) \log(1 - \theta) \\ &= R \log \theta + (m - R) \log(1 - \theta)\end{aligned}\tag{1}$$

- $R = \sum_i y^{(i)}$: the total count of heads
- we will use the likelihood function eq.(1) for logistic regression later

Some plots of (scaled) likelihood



$$m = 20; R = \sum x_i = 10$$
$$m = 100; R = \sum x_i = 50$$
$$m = 200; R = \sum x_i = 100$$



$$m = 40; R = \sum x_i = 20$$
$$m = 40; R = \sum x_i = 5$$
$$m = 40; R = \sum x_i = 35$$

MLE for Bernoulli

Take the derivative $\frac{d\mathcal{L}(\theta)}{d\theta}$ and set it to zero

$$\mathcal{L}(\theta) = R \log \theta + (m - R) \log(1 - \theta)$$

$$\frac{d\mathcal{L}}{d\theta} = \frac{R}{\theta} - \frac{m - R}{1 - \theta} = 0$$

$$\Rightarrow \theta_{ML} = \frac{R}{m}$$

- note $R = \sum_{i=1}^m y^{(i)}$ is the count of heads;
- m is the total count
- θ_{ML} is just the relative frequency

Gradient ascent (descent) ?

We can also apply gradient **ascent** (why ascent?):
loop until converge:

$$\theta_{t+1} \leftarrow \theta_t + \alpha \nabla_{\theta} \mathcal{L}(\theta_t)$$

- where

$$\nabla_{\theta} \mathcal{L}(\theta) = \frac{R}{\theta} - \frac{m - R}{1 - \theta}$$

or gradient descent with negative log likelihood $N\mathcal{L}(\theta) = -\mathcal{L}(\theta)$:

$$\theta_{t+1} \leftarrow \theta_t - \alpha \nabla_{\theta} (N\mathcal{L}(\theta_t))$$

- but $\theta \in [0, 1]$: constrained optimisation
- the gradient $\nabla_{\theta} \mathcal{L}(\theta)$ is not defined at $\theta = 0, 1$!
- difficult to converge if step outside: $\theta_t \geq 1; \theta_t \leq 0$

Reparameterisation trick for gradient descent (ascent)

Reparameterisation trick: find f

$$\theta = f(\beta), \text{ such that}$$

- $\beta \in R$ and write $\mathcal{L}(\theta) = \mathcal{L}(f(\beta))$
- use chain rule to find $\nabla_{\beta}\mathcal{L}(\beta) = \nabla_{\theta}\mathcal{L} \cdot \nabla_{\beta}f(\beta)$
- gradient ascent against β ; then transform back

$$\beta_{t+1} \leftarrow \beta_t + \alpha \nabla_{\beta}\mathcal{L}(\beta_t); \quad \theta_{t+1} \leftarrow f(\beta_{t+1})$$

For example, if $\theta > 0$, then

$$\theta = f(\beta) = e^{\beta}, \text{ the new gradient is then}$$

$$\nabla_{\beta}\mathcal{L}(\beta) = \nabla_{\theta}\mathcal{L} \cdot e^{\beta}$$

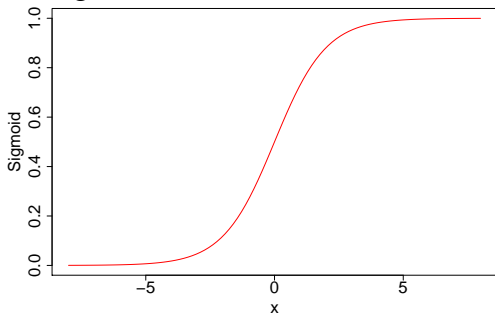
Reparameterisation trick for Bernoulli MLE

For $\theta \in [0, 1]$, such a function is sigmoid:

$$\sigma(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1};$$

The derivative:

$$\frac{d\sigma(x)}{dx} = \sigma(x)(1 - \sigma(x))$$



Reparameterisation trick for Bernoulli MLE

For the Bernoulli case, reparameterize θ :

$$\theta = \sigma(\beta);$$

Rewrite the log likelihood \mathcal{L} as a function of β :

$$\mathcal{L}(\beta) = \log \prod_{i=1}^m \theta^{y^{(i)}} (1 - \theta)^{1-y^{(i)}} = \log \prod_{i=1}^m \sigma(\beta)^{y^{(i)}} (1 - \sigma(\beta))^{1-y^{(i)}}$$

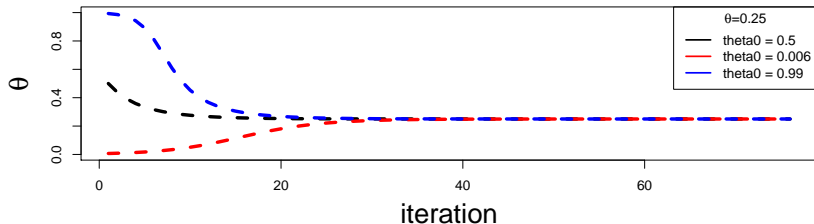
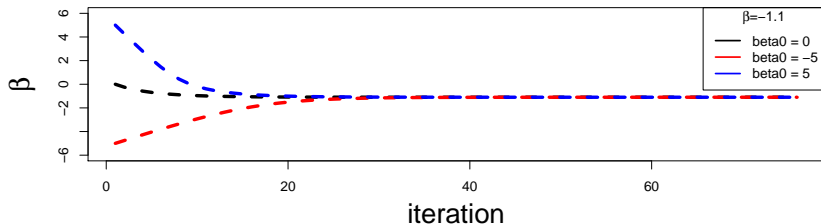
The gradient of L w.r.t β is

$$\nabla_{\beta} \mathcal{L}(\beta) = \nabla_{\theta} \mathcal{L} \cdot \nabla_{\beta} \theta = \left(\frac{R}{\sigma} - \frac{m - R}{1 - \sigma} \right) \sigma(1 - \sigma)$$

Code (R like syntax)

```
grad <- function(m,r,beta){  
  sig <- sigmoid(beta)  
  g <- (r/sig - (m-r)/(1-sig))*sig*(1-sig)  
  return(g)  
}  
  
berGAscent <- function(alpha, iter, m, r, beta0){  
  betas <- vector(mode="numeric", length = iter+1)  
  betas[1] <- beta <- beta0  
  for(i in 1:iter){  
    g <- grad(m,r,beta)  
    betas[i+1] <- beta <- beta + alpha*g  
  }  
  return(betas)  
}
```


Example with $m = 100, R = 25, \theta_{ML} = 0.25, \alpha = 0.01$



MLE for Gaussian

Similarly, for Gaussian $\mathcal{D} = \{y^{(1)}, \dots, y^{(m)}\}$, the parameters are $\theta = \{\mu, \sigma^2\}$ and

$$p(y^{(i)}; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left\{ -\frac{1}{2\sigma^2} (y^{(i)} - \mu)^2 \right\}$$

therefore, the log likelihood for $y^{(i)}$ is:

$$\log p(y^{(i)}; \mu, \sigma^2) = -\frac{1}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \underbrace{(y^{(i)} - \mu)^2}_{\text{squared error!}}$$

$$\begin{aligned}
\mathcal{L}(\mu, \sigma^2) &= \log p(\mathcal{D}|\mu, \sigma^2) = \log \prod_{i=1}^m p(y^{(i)}; \mu, \sigma^2) = \sum_{i=1}^m \log p(y^{(i)}; \mu, \sigma^2) \\
&= \sum_{i=1}^m \left(-\frac{1}{2} \log(2\pi\sigma^2) - \frac{(y^{(i)} - \mu)^2}{2\sigma^2} \right) \\
&= -\frac{m}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \underbrace{\sum_{i=1}^m (y^{(i)} - \mu)^2}_{\text{sum of squared error}}
\end{aligned}$$

Take (partial) derivative and set to zero (verify yourself!):

$$\frac{\partial \mathcal{L}}{\partial \mu} = \frac{1}{\sigma^2} \left(\sum_{i=1}^m (y^{(i)} - \mu) \right) = 0; \quad \frac{\partial \mathcal{L}}{\partial \sigma^2} = -\frac{m}{2\sigma^2} + \frac{\sum_{i=1}^m (y^{(i)} - \mu)^2}{2(\sigma^2)^2} = 0$$

$$\Rightarrow \begin{cases} \mu_{ML} = \frac{1}{m} \sum_{i=1}^m y^{(i)} & \leftarrow \text{sample mean!} \\ \sigma_{ML}^2 = \frac{1}{m} \sum_{i=1}^m (y^{(i)} - \mu_{ML})^2 \end{cases}$$

Linear regression: revisit

Linear regression model:

$$y^{(i)} = \boldsymbol{\theta}^T \mathbf{x}^{(i)} + e^{(i)}$$

- $i = 1, \dots, m$: index of data samples (row index),
- $\mathbf{x}^{(i)} = [1, x_1^{(i)}, \dots, x_n^{(i)}]^T$ is a $(n+1) \times 1$ vector:
 - n : number of predictors (columns)
- $\boldsymbol{\theta}$ is the model parameter
- $e^{(i)}$ is the prediction difference

Assume

$$e^{(i)} \sim \mathcal{N}(0, \sigma^2)$$

- the prediction error is Gaussian distributed
- the mean of the error is 0
- the variance is σ^2 , which needs to be estimated

Linear regression: revisit

$$e^{(i)} \sim \mathcal{N}(0, \sigma^2)$$

\Downarrow

$$y^{(i)} = \boldsymbol{\theta}^T \mathbf{x}^{(i)} + e^{(i)} \sim \mathcal{N}(\boldsymbol{\theta}^T \mathbf{x}^{(i)}, \sigma^2)$$

\Downarrow

$$p(y^{(i)} | \boldsymbol{\theta}, \sigma^2, \mathbf{x}^{(i)}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y^{(i)} - \boldsymbol{\theta}^T \mathbf{x}^{(i)})^2}{2\sigma^2}\right)$$

\Downarrow

$$\mathcal{L}(\boldsymbol{\theta}, \sigma^2) = \log p(\mathcal{D} | \boldsymbol{\theta}, \sigma^2) = \log p(\mathbf{y} | \boldsymbol{\theta}, \sigma^2, \mathbf{X}) = \log \prod_{i=1}^m p(y^{(i)}; \boldsymbol{\theta}, \mathbf{x}^{(i)})$$

Linear regression: maximum likelihood estimation

The log likelihood function is:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}, \sigma^2) &= \log \prod_{i=1}^m p(y^{(i)}; \boldsymbol{\theta}, \mathbf{x}^{(i)}) \\ &= \sum_{i=1}^m \log p(y^{(i)}; \boldsymbol{\theta}, \mathbf{x}^{(i)}) \\ &= -\frac{m}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^m (y^{(i)} - \boldsymbol{\theta}^T \mathbf{x}^{(i)})^2\end{aligned}$$

Maximising \mathcal{L} w.r.t $\boldsymbol{\theta}$ is the same as minimising loss function

$$\begin{aligned}L(\boldsymbol{\theta}) &= \sum_{i=1}^m (y^{(i)} - \boldsymbol{\theta}^T \mathbf{x}^{(i)})^2 \\ \Rightarrow \boldsymbol{\theta}_{ML} &= \boldsymbol{\theta}_{LS}\end{aligned}$$

Logistic regression

Let's consider binary classification $y^{(i)} \in \{1, 0\}$, assume Bernoulli likelihood

$$P(y^{(i)} = 1) = \sigma(\boldsymbol{\theta}^T \mathbf{x}^{(i)})$$

- $i = 1, \dots, m$: index of data samples (row index)
- $\boldsymbol{\theta}^T \mathbf{x}^{(i)} = \theta_0 + \theta_1 x_1^{(i)} + \dots + \theta_n x_n^{(i)} \in \mathbb{R}$
- $\sigma(x) \in [0, 1]$
- $\boldsymbol{\theta}$ is the model parameter

The log likelihood function is

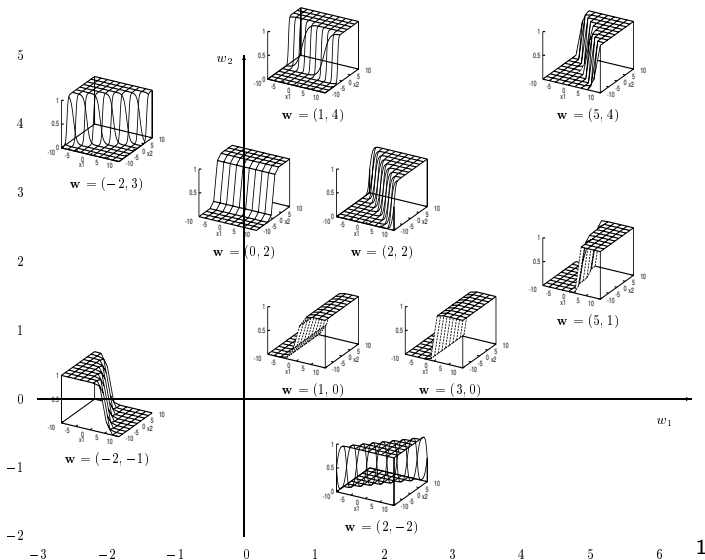
$$\mathcal{L}(\boldsymbol{\theta}) = \underbrace{\log \prod_{i=1}^m \sigma^{y^{(i)}} (1 - \sigma)^{1-y^{(i)}}}_{\text{the same as Bernoulli model with } \theta}$$

- replace single parameter $\sigma(\beta)$ with $\sigma(\boldsymbol{\theta}^T \mathbf{x}^{(i)})$
- β serves the same purpose as θ_0 : the intercept (cf. page 22)

Logistic regression: geometric view

$$\sigma(\boldsymbol{\theta}^T \mathbf{x})$$

- $\boldsymbol{\theta}^T \mathbf{x}$ is a hyperplane
- $\sigma(\boldsymbol{\theta}^T \mathbf{x})$ squeeze the plane between $(0, 1)$
- $\boldsymbol{\theta}$ determines the direction of surface facing
- $\|\boldsymbol{\theta}\|_2^2$ determines the steepness



¹Information theory, inference and learning algorithms, David MacKay

Summary

Maximum likelihood estimation

- gives rise to squared error loss function for regression
 - sample mean is the simplest kind of linear regression where $x^{(i)} = 1$ for all $i = 1, \dots, m$
- gives rise to logistic error (cross-entropy) for classification
 - relative frequency is the simplest kind of logistic regression where $x^{(i)} = 1$ for all $i = 1, \dots, m$

Suggested reading and exercises

Reading

- MLAPP 2.2, 2.3.1, 2.3.2, 2.4.1, 7.3, 8.1-8.3.1
- DL 3, 5.5, 5.7.1
- Information theory, inference and learning algorithms by David MacKay, chapter 2, 22.1, 39.1, 39.2

Exercise

- go through the equations
- write gradient descent for Gaussian model's likelihood function
 - generate some artificial data
 - workout the gradients
 - use the reparameterisation trick to treat $\sigma^2 > 0$
 - check whether they converge
- derive the gradient for logistic regression's log likelihood function

Next time

- Large number theory of MLE

$$\theta_{ML} \rightarrow \mathcal{N}(\theta, I_m^{-1}(\theta))$$

- ML estimator can recover the true parameter θ
 - as data size $m \rightarrow \infty$
- gradient descent of logistic regression
- Newton's method for optimisation

*Random variable: formal aspects

Formally, r.v. X is a mapping from *sample space* Ω to *target space* \mathcal{T}

- Ω : all possible outcomes of an experiment
- \mathcal{T} : possible values X can take
- events $E \subseteq \Omega$
- $X(\omega) \in \mathcal{T}, \forall \omega \in \Omega$
- X^{-1} defines a partition of Ω

Example: toss a fair coin twice, r.v. X : # of heads turned up

- the *sample space* is $\Omega = \{HH, TT, HT, TH\}$
- *target space* is $\mathcal{T} = \{0, 1, 2\}$
- $X(HH) = 2; X(HT) = X(TH) = 1; X(TT) = 0$
- $X^{-1} = \{E_0, E_1, E_2\}$ defines a partition of Ω :
 $E_0 = \{TT\}, E_1 = \{TH, HT\}, E_2 = \{HH\}$
 - disjoint: $E_0 \cap E_1 = E_0 \cap E_2 = E_1 \cap E_2 = \emptyset$
 - complete: $E_0 \cup E_1 \cup E_2 = \Omega$