DS-GA 1003: Machine Learning

Lecture 6: Probabilistic Modeling & MLE

Slides adapted from material from David Rosenberg.

Logistics & Announcements

Midterm next week. Usual class time, 2 hours.

All details on the Midterm section of course website, including review sheet.

Lab this week is midterm review session (might go 30min - 45min over to review more).

Midcourse survey on Ed. Anonymized and sent to CDS admin, sent back to us in two weeks.

Nick and Sam's first time teaching this course — any feedback much appreciated!

Project groups. Form on Ed if you don't have a group yet.

Spring break. Almost here! Week after midterm (no office hours, etc. that week).

Outline

Probabilistic Modeling

Review: Maximum Likelihood Estimation

Conditional Probability Model: Linear Regression

Conditional Probability Model: Logistic Regression

Generalized Linear Models

Generative Model: Naive Bayes

Probabilistic Modeling

Motivation

Up to now, machine learning involved:

1. Choosing an appropriate hypothesis class (e.g. linear/affine functions).

2. Choosing an appropriate loss function to setup the ERM problem:

$$\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell(h(x^{(i)}), y^{(i)}).$$

3. Apply optimization (e.g. gradient descent) to the ERM problem to solve and get a hypothesis $\hat{h}$.

Probabilistic Modeling

Motivation

In the probabilistic modeling viewpoint:

Model our *belief* in the data-generating distribution $P_{\mathcal{X} \times \mathcal{Y}}$.

Learning as **statistical inference**.

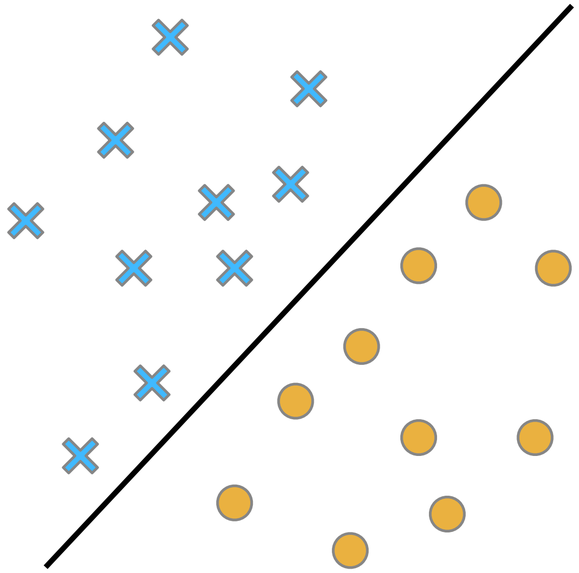
Can recover/give an alternative view to methods we've already seen (logistic regression, linear regression).

Optimization problem changes from ERM to **Maximum Likelihood Estimation (MLE)**.

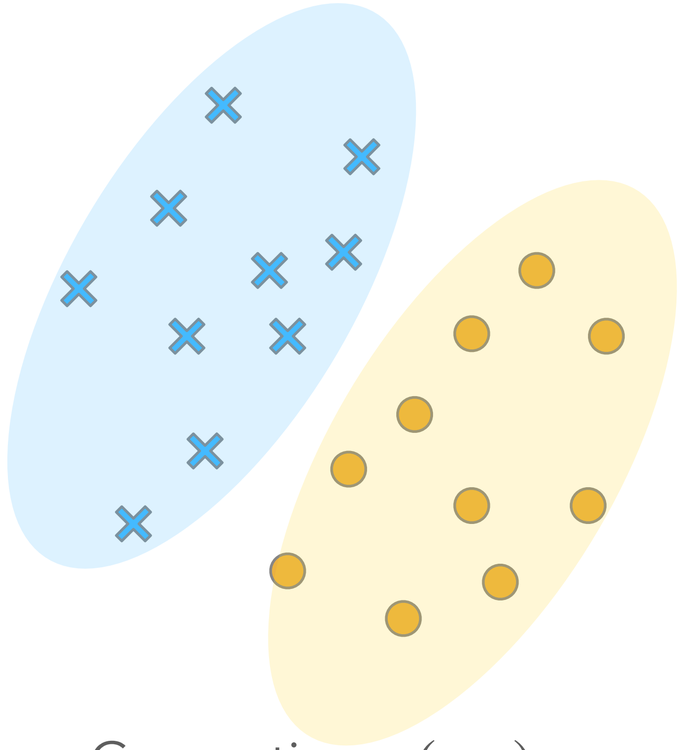
How data is generated

Conditional vs. Generative Models

Our goal will be to *model* how the data are generated (through $P_{\mathcal{X}\times\mathcal{Y}}$) and apply MLE.



Conditional: $p(y \mid x)$



Generative: $p(x, y)$

Outline

Probabilistic Modeling

Review: Maximum Likelihood Estimation

Conditional Probability Model: Linear Regression

Conditional Probability Model: Logistic Regression

Generalized Linear Models

Generative Model: Naive Bayes

Parametric Estimation

Definition

A parametric model is a class of functions of the form:

$$\mathcal{F} := \{f(z;\theta) : \theta \in \Theta\},$$

where $\Theta \subseteq \mathbb{R}^k$ is the parameter space and $\theta = (\theta_1, \ldots, \theta_k)$ are the model parameters.

Example. The parameter space for the Gaussian distribution $N(\mu, \sigma^2)$ is

$$\Theta = \{(\mu, \sigma) : \mu \in \mathbb{R}, \sigma > 0\}.$$

Example. The parameter space for the Bernoulli distribution $\mathrm{Ber}(p)$ is

$$\Theta = \{p : 0 \leq p \leq 1\}.$$

Maximum Likelihood Estimation

Intuition

One way to do *parametric estimation* given i.i.d. data $z_1, \ldots, z_n$ is maximum likelihood estimation.

We assume that $z_1, \ldots, z_n$ are from a distribution with PDF $p(z; \theta)$ and parameter space $\Theta \subseteq \mathbb{R}^k$.

"Assume that the data come from a Gaussian with $p(z; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{ -\frac{1}{2\sigma^2}(z-\mu)^2 \right\}$*"*

We consider the likelihood function which maps from parameters Θ to some positive number: the "likelihood" of those parameters explaining the data.

Maximum Likelihood Estimation

Intuition

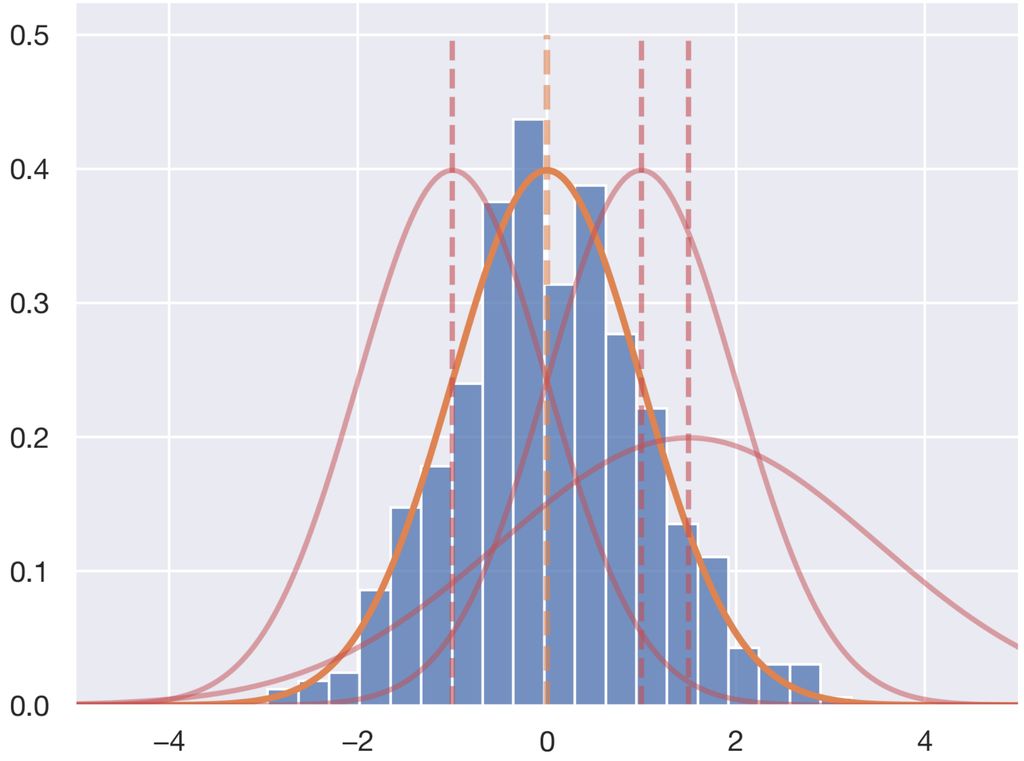
One way to do *parametric estimation* given i.i.d. data $z_1, \ldots, z_n$ is maximum likelihood estimation.

We assume that $z_1, \ldots, z_n$ are from a distribution with PDF $p(z; \theta)$ and parameter space $\Theta \subseteq \mathbb{R}^k$.

"Assume that the data come from a Gaussian with

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

We consider the likelihood function which maps from parameters Θ to some positive number: the "likelihood" of those parameters explaining the data.



Maximum Likelihood Estimation

Definition

Consider the parametric model

$$\mathcal{F} := \{f(z;\theta) : \theta \in \Theta\}.$$

Let $z_1, \ldots, z_n$ be **i.i.d. (independent, identically dist.)** random variables. The likelihood function is:

$$L_n(\theta) := \prod_{i=1}^{n} f(z_i;\theta).$$

Note that $z_1, \ldots, z_n$ are fixed here, so this is *just* a function of θ.

"How well does θ describe my data $z_1, \ldots, z_n$?"

Maximum Likelihood Estimation

Why log-likelihood?

The log-likelihood function is the function defined by:

$$\mathscr{L}_n(\theta) := \log L_n(\theta) = \sum_{i=1}^{n} \log f(z_i; \theta).$$

The maximum likelihood estimator $\hat{\theta}_{MLE}$ is the value of θ that maximizes $L_n(\theta)$.

$$\hat{\theta}_{MLE} = \arg\max_{\theta} L_n(\theta) = \arg\max_{\theta} \mathscr{L}_n(\theta).$$

$$\hat{\theta}_{MLE} = \arg\min_{\theta} -L_n(\theta) = \arg\min_{\theta} -\mathscr{L}_n(\theta)$$

$\log(\cdot)$ is a *monotonic* function, so the maximizer of $\log f$ corresponds to the maximizer of f.

MLE for Bernoulli

Step 1: Write down likelihood function

Example. Suppose $z_1, \ldots, z_n \sim \mathrm{Ber}(p)$ over $\{0,1\}$, so our parametric model is:

$$\mathscr{F} = \left\{ f(z;p) = p^z(1-p)^{1-z} : p \in [0,1] \right\}$$

$\Theta = \{p : 0 \leq p \leq 1\}$. The unknown parameter θ is p.

Likelihood function. The likelihood function is

$$L_n(\theta) = L_n(p) = \prod_{i=1}^{n} f(z_i;p) = \prod_{i=1}^{n} p^{z_i}(1-p)^{1-z_i} = p^{\sum_{i=1}^n z_i}(1-p)^{n-\sum_{i=1}^n z_i}.$$

Denote $S := \sum_{i=1}^{n} z_i$, and the likelihood function is: $L_n(p) = p^S(1-p)^{n-S}$

MLE for Bernoulli

Step 2: Simplify using log-likelihood

Example. Suppose $z_1, \ldots, z_n \sim \mathrm{Ber}(p)$, so our parametric model is:

$$\mathcal{F} = \left\{ f(z; p) = p^z (1-p)^{1-z} : p \in [0,1] \right\}$$

$\Theta = \{p : 0 \leq p \leq 1\}$. The unknown parameter θ is p.

Likelihood function. Denote $S := \sum_{i=1}^{n} z_i$, and the likelihood function is: $L_n(p) = p^S (1-p)^{n-S}$

Log-likelihood function. The log-likelihood is

$\mathcal{L}_n(p) = S \log p + (n - S) \log(1-p)$. Now optimize this with respect to p!

MLE for Bernoulli

Step 3: Optimize log-likelihood using calculus

Example. Suppose $z_1, \ldots, z_n \sim \mathrm{Ber}(p)$, so our parametric model is:

$$\mathscr{F} = \left\{ f(z; p) = p^z (1-p)^{1-z} : p \in [0,1] \right\}$$

$\Theta = \{p : 0 \leq p \leq 1\}$. The unknown parameter θ is p.

Optimizing the negative log-likelihood. We need to solve the optimization problem:

$$\underset{p \in [0,1]}{\text{minimize}} \quad -\mathscr{L}_n(p) = -S \log p + (S-n) \log(1-p).$$

First order condition: $\nabla_p \mathscr{L}_n(p) = -\dfrac{S}{p} - \dfrac{S-n}{1-p} = 0.$

Solving for p, we get: $\hat{p}_{MLE} = \dfrac{S}{n} = \dfrac{1}{n} \displaystyle\sum_{i=1}^{n} z_i.$

Maximum Likelihood Estimation

Example: Bernoulli

Example. Suppose $z_1, \ldots, z_n \sim \text{Ber}(p)$, so our parametric model is:

$$\mathscr{F} = \left\{ f(z; p) = p^z (1-p)^{1-z} : p \in [0,1] \right\}$$

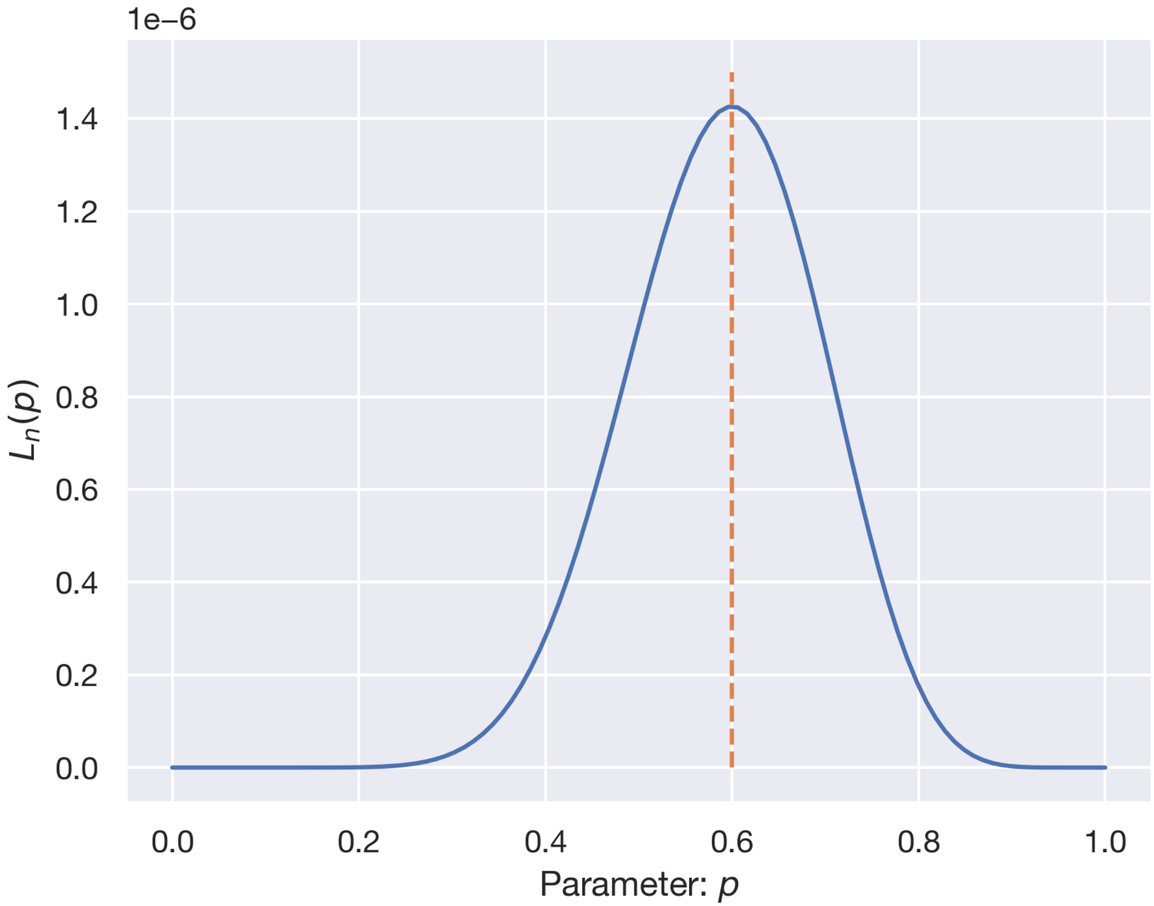
$\Theta = \{p : 0 \leq p \leq 1\}$. The unknown parameter θ is p.

The [likelihood function](#) is:

$$L_n(p) = p^{\sum_{i=1}^n X_i} (1-p)^{n - \sum_{i=1}^n X_i}$$

The [maximum likelihood estimator](#) of the estimand p is:

$$\hat{p}_{MLE} = \frac{S}{n} = \frac{1}{n} \sum_{i=1}^{n} X_i.$$



Outline

Probabilistic Modeling

Review: Maximum Likelihood Estimation

Conditional Probability Model: Linear Regression

Conditional Probability Model: Logistic Regression

Generalized Linear Models

Generative Model: Naive Bayes

Linear Regression

Review

Predict real-valued output $y \in \mathbb{R}$ from features $x \in \mathbb{R}^d$.

Input space: $\mathcal{X} = \mathbb{R}^d$; Output space: $\mathcal{Y} = \mathbb{R}$.

Hypothesis class: $\mathcal{H} = \{h(x) = w^\top x : w \in \mathbb{R}^d\}$ (linear functions).

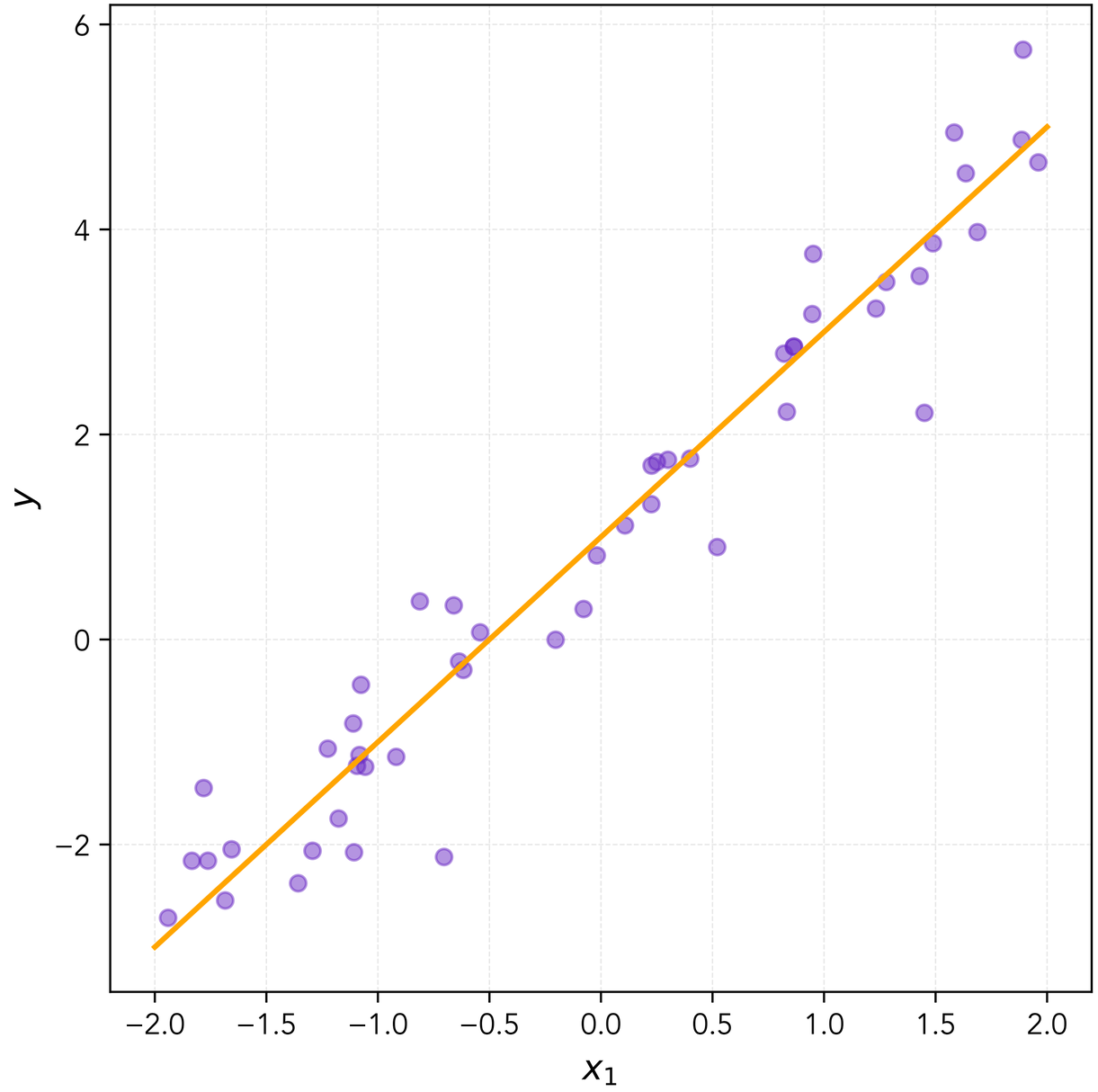
Loss function: $\ell(\hat{y}, y) = (\hat{y} - y)^2$ (squared loss).

$$\hat{R}_n(w) = \frac{1}{n}\sum_{i=1}^{n}(w^\top x^{(i)} - y^{(i)})^2 \text{ or } \hat{R}_n(w) = \frac{1}{n}\|Xw - y\|^2 \text{ with } X \in \mathbb{R}^{n\times d},\ y \in \mathbb{R}^n.$$

Linear Regression

Example: $d = 1$

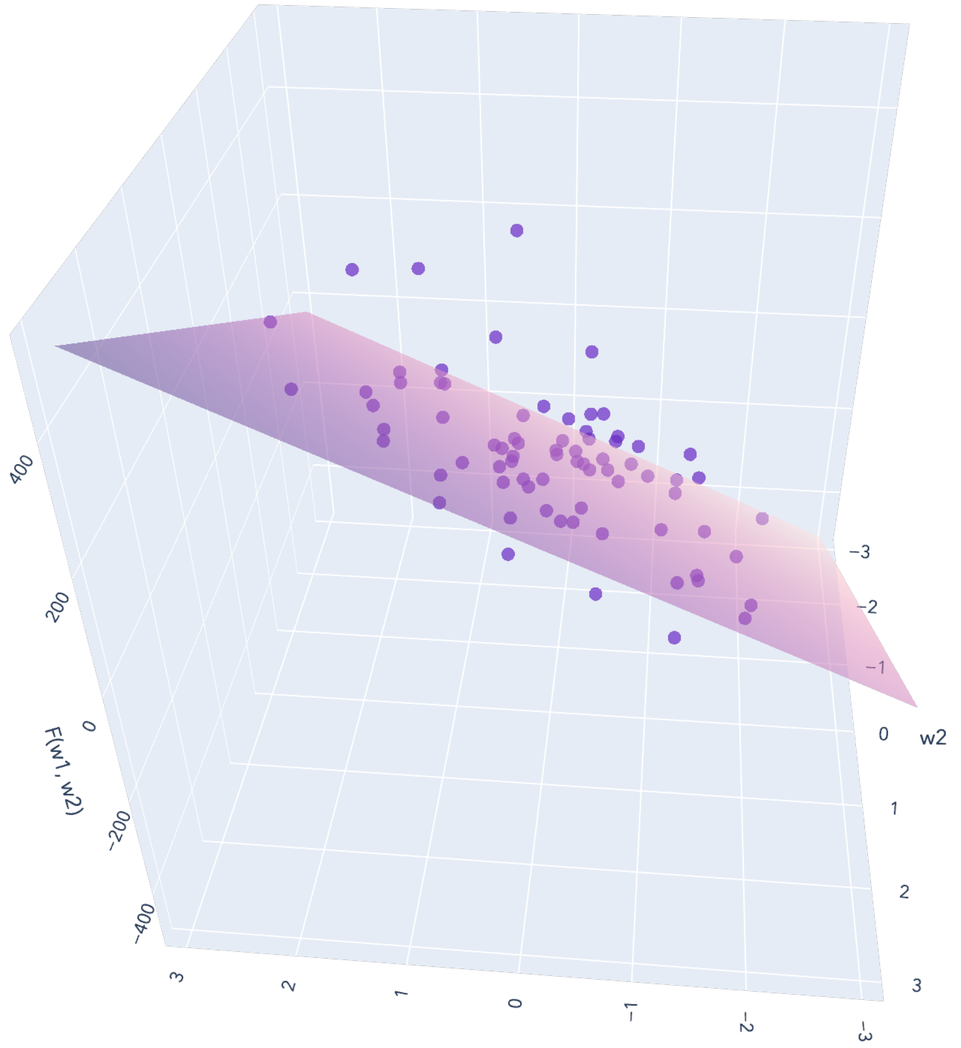
$$X = \begin{bmatrix} \vdots \\ -0.58 \\ 1.36 \\ 1.30 \\ -0.86 \\ \vdots \end{bmatrix} \quad y = \begin{bmatrix} \vdots \\ -0.30 \\ 3.16 \\ 3.29 \\ -1.75 \\ \vdots \end{bmatrix}$$



Linear Regression

Example: $d = 2$

$$X = \begin{bmatrix} \vdots & \vdots \\ 0.51 & -0.53 \\ -0.56 & -1.72 \\ -0.57 & -0.99 \\ 1.54 & 0.36 \\ \vdots & \vdots \end{bmatrix} \quad y = \begin{bmatrix} \vdots \\ -85.35 \\ -121.2 \\ -46.14 \\ 154.72 \\ \vdots \end{bmatrix}$$



Linear Regression

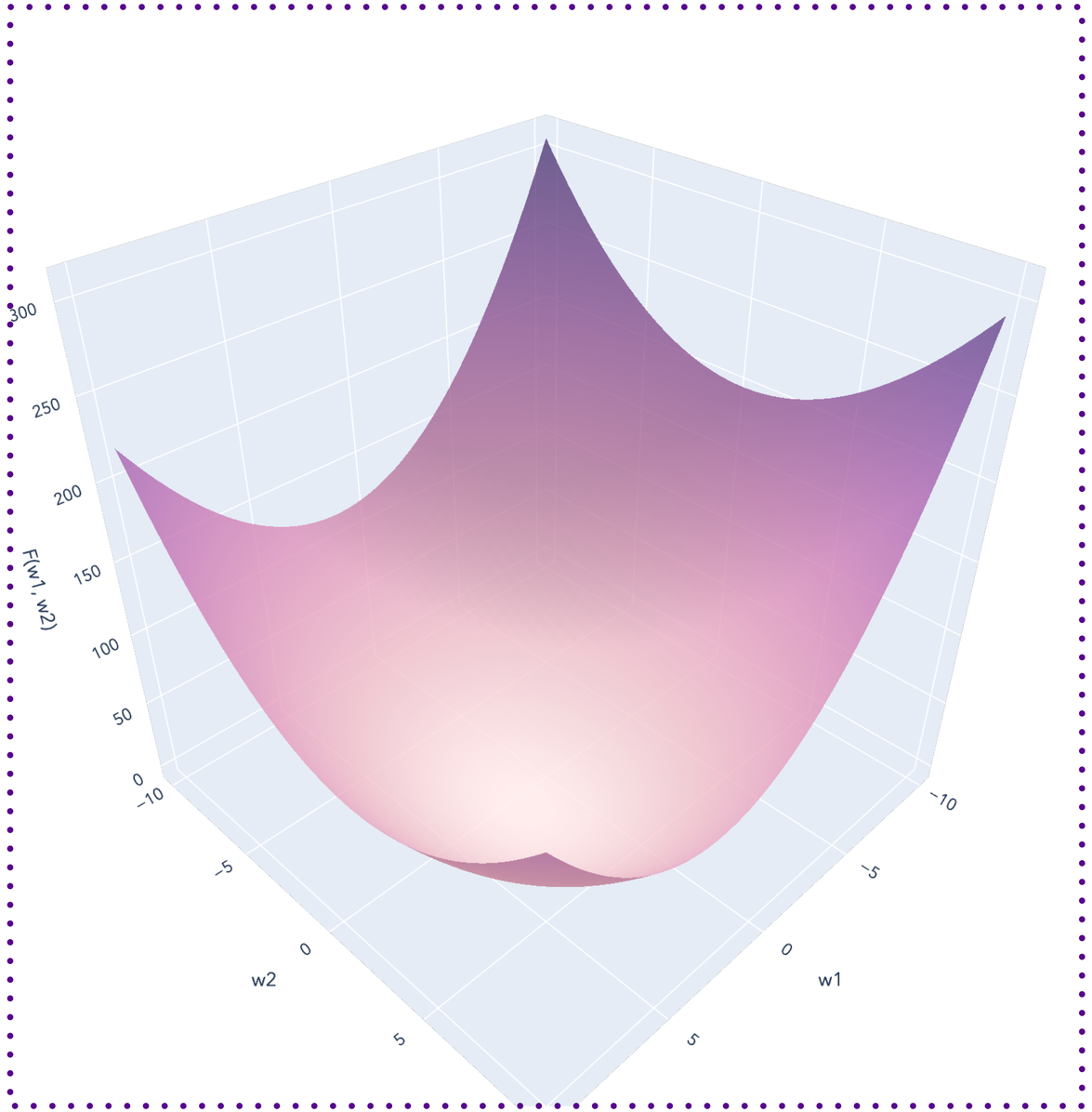
Review

Given $D_n := \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$ we want to minimize the empirical risk:

$$\hat{R}_n(w) = \frac{1}{n}\|Xw - y\|^2$$

Closed-form solution: $(X^\top X)^{-1} X^\top y$.

GD: $w^{(t)} \leftarrow w^{(t-1)} - \eta \cdot \frac{2}{n} X^\top (Xw - y)$



Linear Regression

Motivation for Probabilistic Perspective

Questions:

What could be a reason that squared loss is a reasonable loss function?

What assumptions are we making on the *data*?

$$\hat{R}_n(w) = \frac{1}{n} \sum_{i=1}^{n} (w^\top x^{(i)} - y^{(i)})^2$$

Probabilistic modeling perspective:

For each x, the conditional distribution $p(y \mid x)$ is Gaussian!

Linear Regression

Probabilistic Assumptions

Assume: x and y are related through a linear function:

$$y = x^\top w^* + \epsilon, \text{ where } \epsilon \text{ is the residual error.}$$

View ϵ as capturing the effects not captured by linear model (e.g. noise).

The errors are i.i.d. (independently and identically distributed):

$$\epsilon \sim N(0, \sigma^2).$$

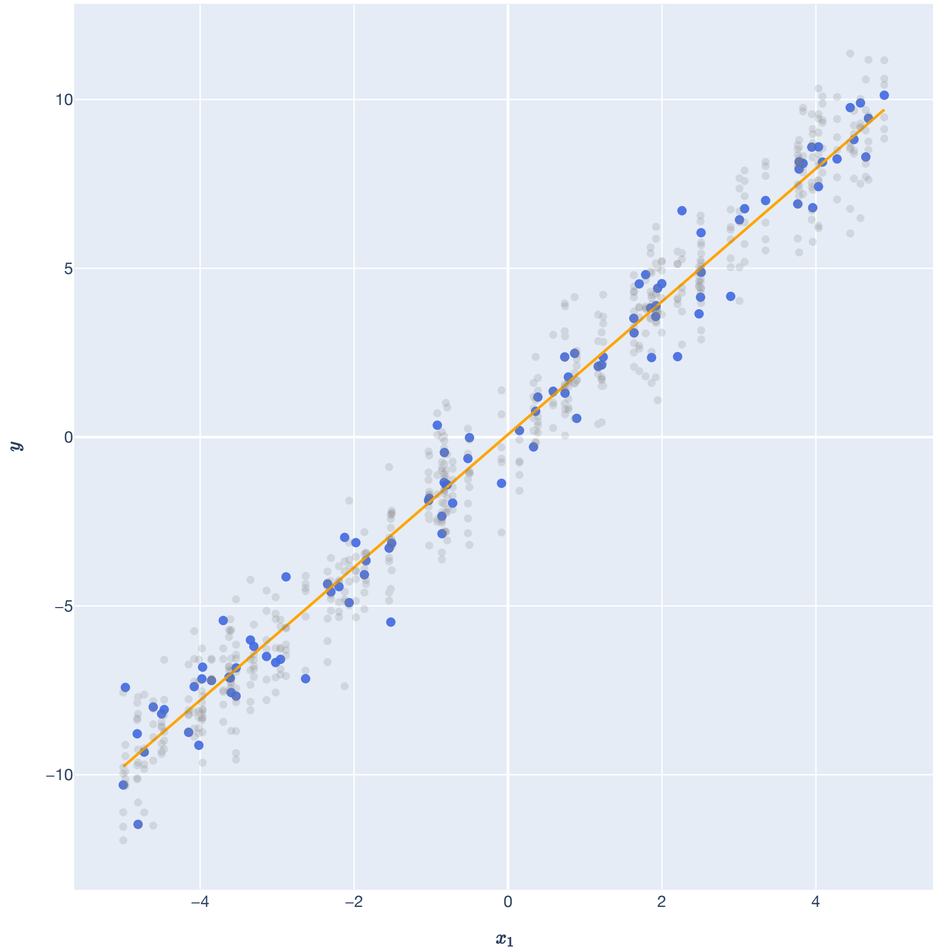
What's the distribution of $y \mid x$?

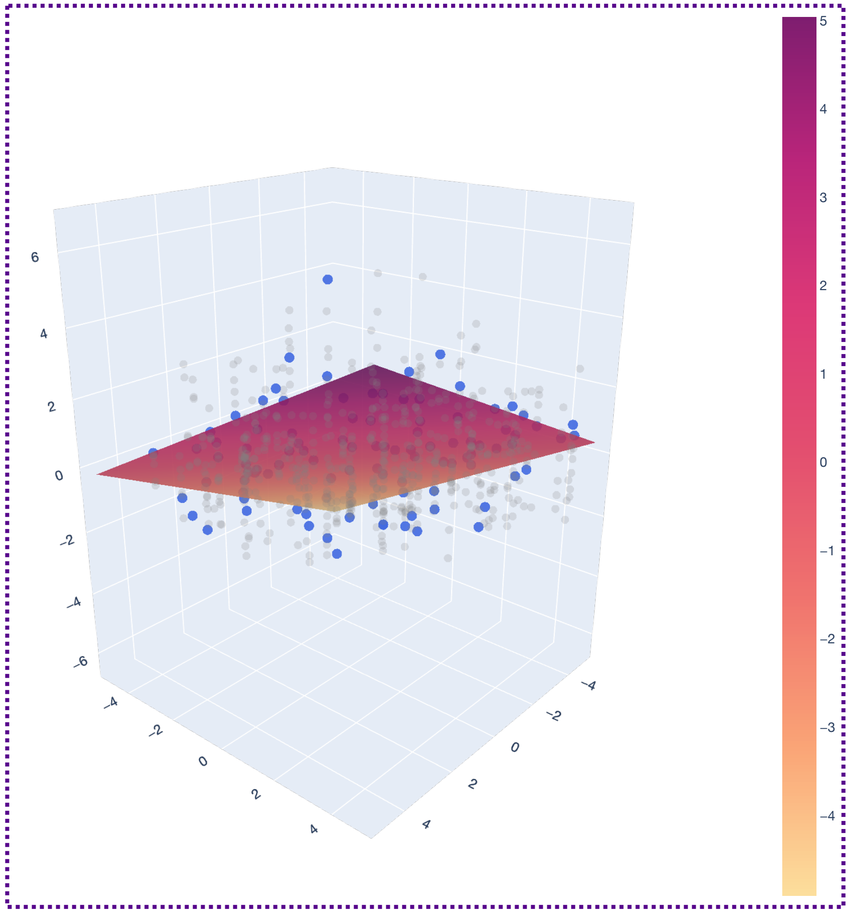
$$p(y \mid x; w^*, \sigma) = N(x^\top w^*, \sigma^2).$$

Linear Regression

Probabilistic Assumptions

$$p(y \mid x; w^*, \sigma) = N(x^\top w^*, \sigma^2)$$





MLE for Linear Regression

Likelihood and log-likelihood function

$$p(y \mid x; w^*, \sigma) = N(x^\top w^*, \sigma^2).$$

Given a dataset $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})$, the (conditional) likelihood function is:

$$L_n(\theta) := \prod_{i=1}^{n} f(z_i; \theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; \theta).$$

Our parameters are $\theta = (w, \sigma^2)$. The log-likelihood function is:

$$\mathscr{L}_n(\theta) = \log L_n(\theta)$$

$$\mathscr{L}_n(w, \sigma^2) = \log L_n(w, \sigma^2) = \log \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; w, \sigma^2) = \sum_{i=1}^{n} \log p(y^{(i)} \mid x^{(i)}; w, \sigma^2)$$

The Gaussian Distribution

Review: PDF of Gaussian

A random variable z has a Gaussian/Normal distribution with parameters μ and σ, denoted $z \sim N(\mu, \sigma^2)$ if it has PDF:

$$p(z) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2\sigma^2}(z-\mu)^2\right\}, \text{ for all } z \in \mathbb{R}.$$

This random variable has mean $\mathbb{E}[z] = \mu$ and variance $\mathrm{Var}(z) = \sigma^2$.

MLE for Linear Regression

Likelihood and log-likelihood function

$$p(y \mid x; \mu, \sigma) = N(x^\top w, \sigma^2).$$

$$\mathscr{L}_n(w, \sigma^2) = \sum_{i=1}^{n} \log p(y^{(i)} \mid x^{(i)}; w, \sigma^2)$$

$$= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y^{(i)} - w^\top x^{(i)})^2}{2\sigma^2} \right)$$

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

MLE for Linear Regression

Maximizing likelihood

Maximizing the log-likelihood involves finding:

$$\max_{w \in \mathbb{R}^d} \mathscr{L}_n(w, \sigma^2)$$

$$\max_{w \in \mathbb{R}^d} \left(N \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y^{(i)} - w^\top x^{(i)})^2 \right)$$

$$= \min_{w \in \mathbb{R}^d} \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y^{(i)} - w^\top x^{(i)})^2$$

So the $w \in \mathbb{R}^d$ maximizing the likelihood is exactly the $w \in \mathbb{R}^d$ solving ERM with squared loss!

MLE for Linear Regression

Gradient of the likelihood

We can also compute the *gradient* of the likelihood with respect to $w \in \mathbb{R}^d$.

$$\mathscr{L}_n(w, \sigma) = N \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y^{(i)} - w^\top x^{(i)})^2$$

The jth partial derivative (for $j \in \{1, \ldots, d\}$ is):

$$\frac{\partial \mathscr{L}_n(w, \sigma)}{\partial w_j} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} (y^{(i)} - w^\top x^{(i)}) x_j^{(i)}$$

Outline

Probabilistic Modeling

Review: Maximum Likelihood Estimation

Conditional Probability Model: Linear Regression

Conditional Probability Model: Logistic Regression

Generalized Linear Models

Generative Model: Naive Bayes

Logistic Regression

Review: Optimization Perspective

Hypothesis class: $\mathscr{H} = \{h_w(x) = w^\top x : w \in \mathbb{R}^d\}$; $\mathscr{X} = \mathbb{R}^d$ and $\mathscr{Y} = \{-1, +1\}$.

Loss: $\ell_{\log}(m) := \log(1 + e^{-m})$ (logistic loss)

For a any hypothesis $h(\cdot)$, the margin on (x, y) is $m = yh(x)$.

Empirical risk minimization:

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \log(1 + \exp(-y^{(i)} w^\top x^{(i)}))$$

Minimizing this objective is known as logistic regression (a linear **classification** method).

Logistic Regression

Probabilistic Perspective

Binary classification setting: let $\mathcal{Y} = \{0,1\}$ (more convenient than $\mathcal{Y} = \{-1, +1\}$).

What's a good assumption for the distribution of $y \mid x$ for each x?

Model $p(y \mid x)$ as a Bernoulli (coin flip) distribution:

$$p(y \mid x) = h(x)^y (1 - h(x))^{1-y} \text{ for each } x \in \mathbb{R}^d.$$

Logistic Regression

Probabilistic Perspective

$$p(y \mid x) = h(x)^y (1 - h(x))^{1-y}.$$

How should we parameterize $h(x)$?

What is $p(y = 1 \mid x)$ and $p(y = 0 \mid x)$? $h(x) \in (0,1)$.

What is the mean of $p(y \mid x)$? $h(x)$

Need a function to map the linear predictor $w^\top x$ in $\mathbb{R}$ to $(0,1)$:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$ (sigmoid/logistic function).

Sigmoid Function

Assumption for Logistic Regression

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

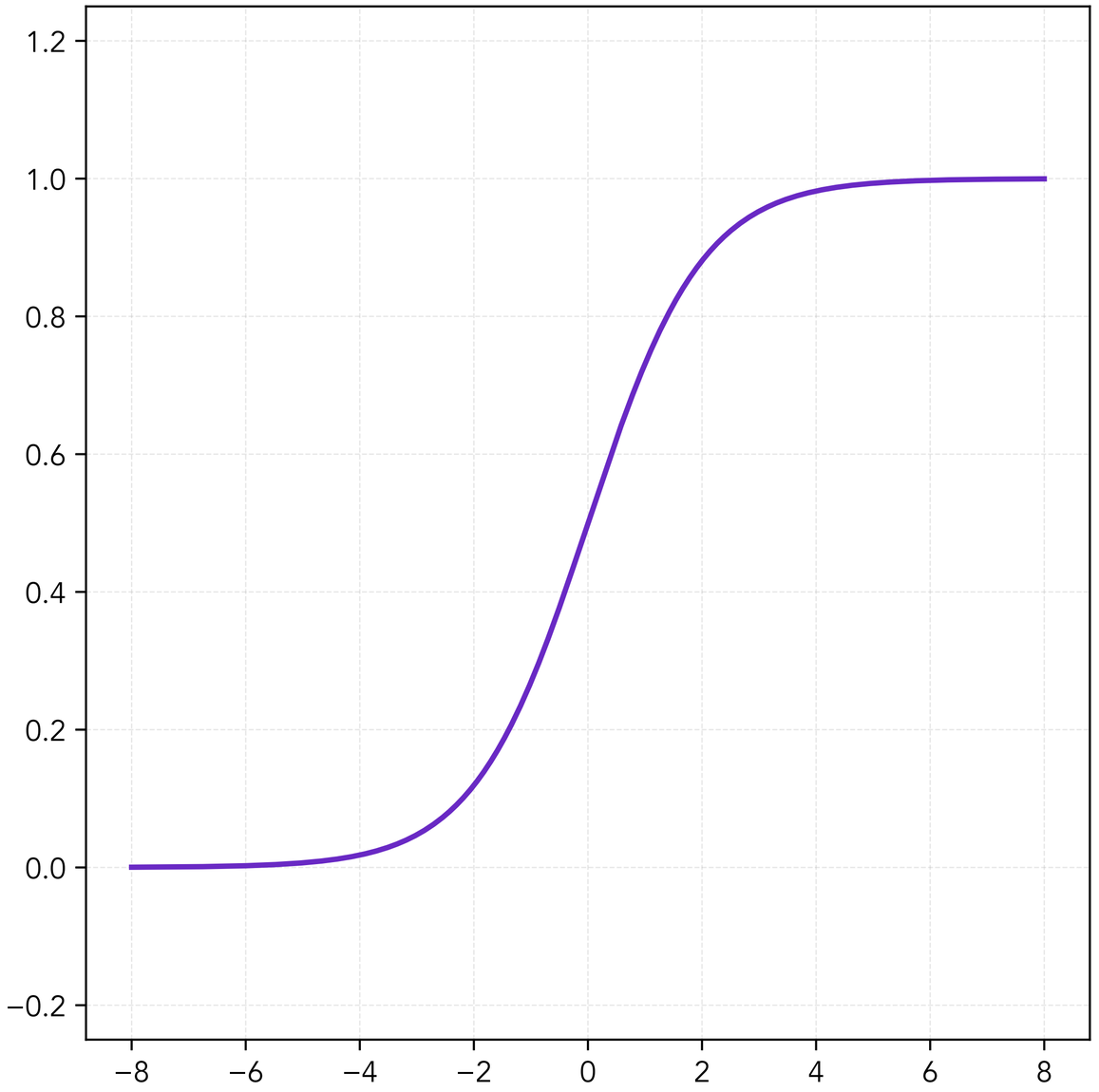
Main assumption:

$$p(y \mid x) = \text{Bernoulli}(\sigma(x^\top w^*))$$

Can show that the log odds is:

$$\log\left(\frac{p(y = 1 \mid x)}{p(y = 0 \mid x)}\right) = x^\top w^*$$

This means we have a *linear decision boundary*.



Sigmoid Function

Assumption for Logistic Regression

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

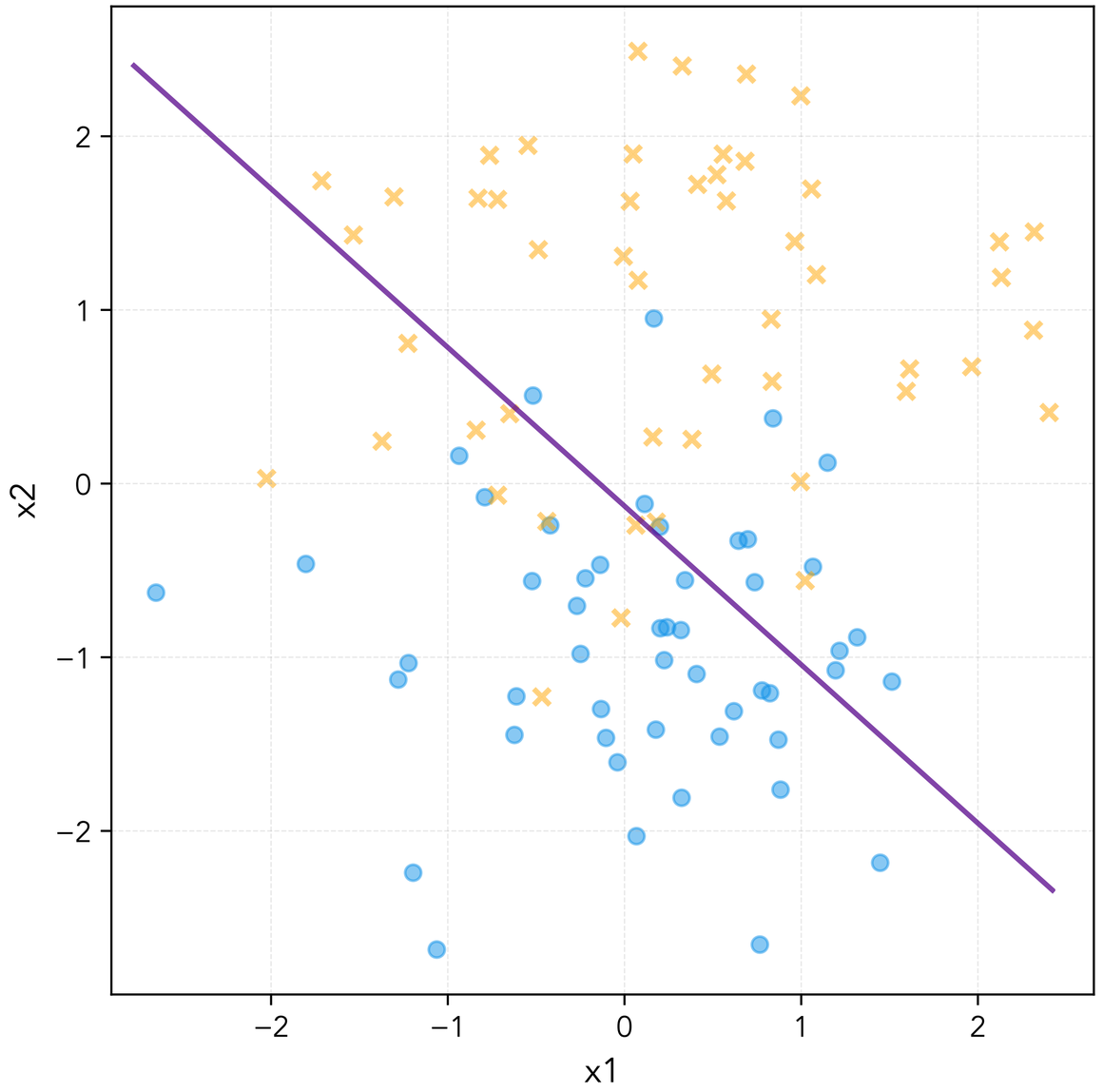
Main assumption:

$$p(y \mid x) = \text{Bernoulli}(\sigma(x^\top w^*))$$

Can show that the log odds is:

$$\log\left(\frac{p(y = 1 \mid x)}{p(y = 0 \mid x)}\right) = x^\top w^*$$

This means we have a *linear decision boundary*.



MLE for Logistic Regression

Likelihood and Log-Likelihood

$$p(y \mid x) = \text{Bernoulli}(\sigma(w^\top x)).$$

Given a dataset $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})$, the (conditional) likelihood function is:

$$L_n(\theta) := \prod_{i=1}^{n} f(z_i; \theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; \theta).$$

Our parameter is $\theta = w$. The log-likelihood function is:

$$\mathscr{L}_n(w) = \log L_n(w) = \log \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; w) = \sum_{i=1}^{n} \log p(y^{(i)} \mid x^{(i)}; w)$$

$$= \sum_{i=1}^{n} y^{(i)} \log \sigma(w^\top x^{(i)}) + (1 - y^{(i)}) \log(1 - \sigma(w^\top x^{(i)}))$$

MLE for Logistic Regression

Gradient of log-likelihood

$$\mathscr{L}_n(w) = \sum_{i=1}^{n} y^{(i)} \log \sigma(w^\top x^{(i)}) + (1 - y^{(i)}) \log(1 - \sigma(w^\top x^{(i)}))$$

$\sigma(z) = (1 + e^{-z})^{-1}$; One useful rule: $\sigma'(z) = \sigma(z)(1 - \sigma(z))$.

Take derivative of the likelihood for example $i \in [n]$ and w.r.t. $j \in [d]$:

$$\frac{\partial \ell^{(i)}}{\partial w_j} = \frac{\partial \ell^{(i)}}{\partial \sigma^{(i)}} \frac{\partial \sigma^{(i)}}{\partial w_j} = \left(\frac{y^{(i)}}{\sigma^{(i)}} - \frac{1 - y^{(i)}}{1 - \sigma^{(i)}} \right) \frac{\partial \sigma^{(i)}}{\partial w_j}$$ chain rule and derivative of $\log z$

$$= \left(\frac{y^{(i)}}{\sigma^{(i)}} - \frac{1 - y^{(i)}}{1 - \sigma^{(i)}} \right) (\sigma^{(i)}(1 - \sigma^{(i)}) x_j^{(i)})$$ derivative of $\sigma(z)$

$$= (y^{(i)} - \sigma^{(i)}) x_j^{(i)}$$

MLE for Logistic Regression

Gradient of log-likelihood

$$\mathscr{L}_n(w) = \sum_{i=1}^{n} y^{(i)} \log \sigma(w^\top x^{(i)}) + (1 - y^{(i)}) \log(1 - \sigma(w^\top x^{(i)}))$$

Therefore:

$$\frac{\partial \ell^{(i)}}{\partial w_j} = (y^{(i)} - \sigma^{(i)}) x_j^{(i)}$$

and the full gradient has entries $j \in \{1, \ldots, d\}$:

$$\frac{\partial \mathscr{L}_n}{\partial w_j} = \sum_{i=1}^{n} (y^{(i)} - \sigma(w^\top x^{(i)})) x_j^{(i)}$$

MLE for Logistic Regression

Gradient ascent for solution

$$\mathscr{L}_n(w) = \sum_{i=1}^{n} y^{(i)} \log \sigma(w^\top x^{(i)}) + (1 - y^{(i)}) \log(1 - \sigma(w^\top x^{(i)}))$$

Logistic regression *does not* have an analytic solution (unlike linear regression).

But logistic regression objective is concave.

In order to *maximize* the logistic regression objective, we must perform **gradient ascent**:

$$w^{(t+1)} \leftarrow w^{(t)} + \eta \nabla \mathscr{L}_n(w)$$

$$w^{(t+1)} \leftarrow w^{(t)} + \eta \cdot \sum_{i=1}^{n} (y^{(i)} - \sigma(w^\top x^{(i)})) x^{(i)}$$

Outline

Probabilistic Modeling

Review: Maximum Likelihood Estimation

Conditional Probability Model: Linear Regression

Conditional Probability Model: Logistic Regression

Generalized Linear Models

Generative Model: Naive Bayes

A closer look at gradients

Comparing linear and logistic regression

Logistic regression log-likelihood & gradient:

$$\mathscr{L}_n(w) = \sum_{i=1}^{n} y^{(i)} \log \sigma(w^\top x^{(i)}) + (1 - y^{(i)}) \log(1 - \sigma(w^\top x^{(i)}))$$

$$\frac{\partial \mathscr{L}_n}{\partial w_j} = \sum_{i=1}^{n} (y^{(i)} - \sigma(w^\top x^{(i)})) x_j^{(i)}$$

Linear regression log-likelihood & gradient:

$$\mathscr{L}_n(w, \sigma) = N \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y^{(i)} - w^\top x^{(i)})^2$$

$$\frac{\partial \mathscr{L}_n}{\partial w_j} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} (y^{(i)} - w^\top x^{(i)}) x_j^{(i)}$$

Call the $f : \mathbb{R} \to \mathbb{R}$ defined as $f(w^\top x)$ a transfer function.

Linear vs. Logistic Regression

Probabilistic Perspective

Linear regression	Logistic regression
Inputs combined with $w^\top x$ (linear model).	**Inputs** combined with $w^\top x$ (linear model).
Outputs are real-valued $\mathcal{Y} = \mathbb{R}$.	**Outputs** are binary-valued $\mathcal{Y} = \{0,1\}$.
$p(y \mid x)$ is assumed Gaussian.	$p(y \mid x)$ is assumed Bernoulli.
Transfer function is identity function $f(z) = z$.	**Transfer function** is sigmoid $f(z) = \sigma(z)$.
Mean is $\mathbb{E}[y \mid x] = f(w^\top x)$.	Mean is $\mathbb{E}[y \mid x] = f(w^\top x)$.

x enters through a linear function.

Main difference is due to different assumed conditional distributions.

Generalized Linear Models

Overview

Task: Given x, predict $p(y \mid x)$.

Modeling:

Choose a parametric family of distributions $p(y; \theta)$ with parameters $\theta \in \Theta$.

Choose a transfer function that maps a linear function to Θ

$$\underbrace{x}_{\mathbb{R}^d} \mapsto \underbrace{w^\top x}_{\mathbb{R}} \mapsto \underbrace{f(w^\top x)}_{\Theta} = \theta$$

Learning: MLE: $\hat{\theta} \in \operatorname{argmax}_{\theta \in \Theta} \log p(D_n; \theta)$ where D_n is the dataset (optimize with gradient ascent).

Prediction: $x \mapsto f(w^\top x)$.

Poisson Regression

Modeling $p(y \mid x)$

Say we want to predict the number of people entering a restaurant in NYC during lunch.

What features would be useful to collect (for $x \in \mathbb{R}^d$)?

What's a good model for the number of visitors given those features $p(y \mid x)$?

A possible model is the Poisson distribution.

$$p(y = k; \lambda) = \frac{\lambda^k e^{-\lambda}}{k!}, \text{ where } \lambda > 0 \text{ and } \mathbb{E}[y] = \lambda \text{ and } k \in \{0,1,2,\ldots,\}.$$

Poisson distribution usually models number of events occurring during fixed period of time.

Poisson Regression

Modeling transfer function

$$p(y = k; \lambda) = \frac{\lambda^k e^{-\lambda}}{k!}, \text{ where } \lambda > 0 \text{ and } \mathbb{E}[y] = \lambda.$$

A transfer function needs to map $w^\top x$ to the parameter:

$$\underbrace{x}_{\mathbb{R}^d} \mapsto \underbrace{w^\top x}_{\mathbb{R}} \mapsto \underbrace{f(w^\top x)}_{(0,\infty)} = \lambda$$

We can take $f(w^\top x) = \exp(w^\top x)$, which maps into the correct range $(0,\infty)$.

Poisson Regression

Likelihood Function

$$p(y = k; \lambda) = \frac{\lambda^k e^{-\lambda}}{k!}, \text{ where } \lambda > 0 \text{ and } \mathbb{E}[y] = \lambda.$$

$$\underbrace{x}_{\mathbb{R}^d} \mapsto \underbrace{w^\top x}_{\mathbb{R}} \mapsto \underbrace{\exp(w^\top x)}_{(0,\infty)}$$

Log-likelihood of dataset $D_n = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$:

$$\log p(y^{(i)}; \lambda) = y^{(i)} \log \lambda_i - \lambda_i - \log(y^{(i)}!), \text{ where } \lambda_i = \exp(w^\top x^{(i)}).$$

$$\log p(D_n; w) = \sum_{i=1}^{n} y^{(i)} \log(\exp(w^\top x^{(i)})) - \exp(w^\top x^{(i)}) - \log(y^{(i)}!)$$

$$= \sum_{i=1}^{n} y^{(i)} w^\top x^{(i)} - \exp(w^\top x^{(i)}) - \log(y^{(i)}!)$$

Multinomial Logistic Regression

Modeling

Say we want to get the predicted *categorial distribution* over k categories for a given $x \in \mathbb{R}^d$.

First, compute the scores $\in \mathbb{R}^k$ and then their softmax:

$$x \mapsto (w_1^\top x, \ldots, w_k^\top x) \mapsto \theta = \left(\frac{\exp(w_1^\top x)}{\sum_{i=1}^k \exp(w_i^\top x)}, \ldots, \frac{\exp(w_k^\top x)}{\sum_{i=1}^k \exp(w_i^\top x)} \right), \text{ each } w_y \in \mathbb{R}^d.$$

The conditional probability for $y \in \{1, \ldots, k\}$ is modeled as:

$$p(y \mid x; w) = \frac{\exp(w_y^\top x)}{\sum_{i=1}^k \exp(w_i^\top x)}.$$

Outline

Probabilistic Modeling

Review: Maximum Likelihood Estimation

Conditional Probability Model: Linear Regression

Conditional Probability Model: Logistic Regression

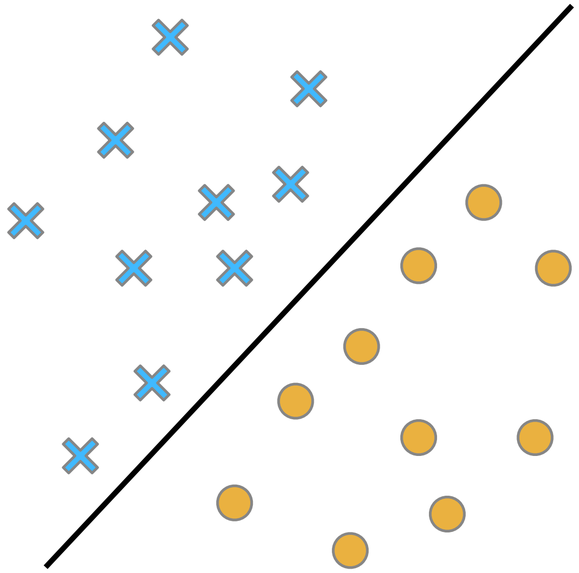
Generalized Linear Models

Generative Model: Naive Bayes

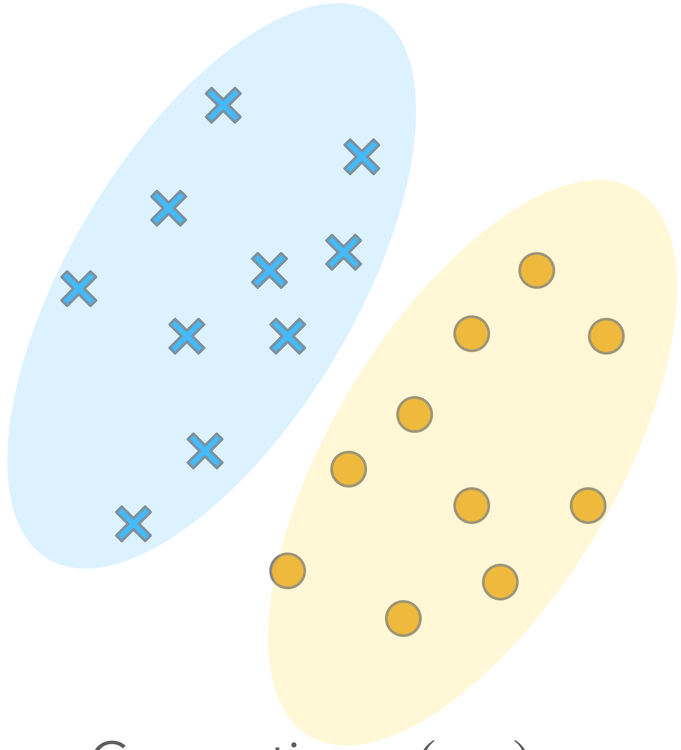
How data is generated

Conditional vs. Generative Models

Our goal will be to *model* how the data are generated (through $P_{\mathcal{X}\times\mathcal{Y}}$) and apply MLE.



Conditional: $p(y \mid x)$



Generative: $p(x, y)$

How data is generated

Conditional vs. Generative Models

Our goal will be to *model* how the data are generated (through $P_{\mathcal{X}\times\mathcal{Y}}$) and apply MLE.

Just saw: Model the *conditional* distribution $p(y \mid x; \theta)$ using generalized linear models.

Previously in class: Directly map x to y (e.g. SVM).

Instead, we'll model the joint distribution $p(x, y; \theta)$.

To predict the label for x, we take $\operatorname{argmax}_{y\in\mathcal{Y}} p(x, y; \theta)$.

Generative Modeling

Bayes Rule

Training:

$$p(x, y) = p(x \mid y)p(y)$$

Test:

$$p(y \mid x) = \frac{p(x \mid y)p(y)}{p(x)} \text{ by Bayes rule.}$$

$$\underset{y \in \mathcal{Y}}{\text{argmax}}\ p(y \mid x) = \underset{y \in \mathcal{Y}}{\text{argmax}} p(x \mid y)p(y)$$

Naive Bayes Model

Example of Generative Modeling

Problem: Binary text classification (e.g. spam vs. genuine email).

Features will be from **bag-of-words** representation of documents:

Example document: [“new” , “job” , “opportunity” , “lets”, “connect”]

$x \in \mathbb{R}^d$ where $x_j \in \{0,1\}$ denotes whether jth word in vocabulary exists in input.

What’s the probability of document x? Use chain rule:

$$
\begin{aligned}
p(x \mid y) &= p(x_1, \ldots, x_d \mid y) \\
&= p(x_1 \mid y) p(x_2 \mid y, x_2) p(x_3 \mid y, x_2, x_1) \ldots p(x_d \mid y, x_{d-1}, \ldots, x_1) \\
&= \prod_{j=1}^{d} p(x_j \mid y, x_{<j})
\end{aligned}
$$

Naive Bayes Model

Assumption

$$p(x \mid y) = \prod_{j=1}^{d} p(x_j \mid y, x_{<j})$$

Problem: $p(x_j \mid y, x_{<j})$ is hard to model (and estimate), especially for large j.

Naive Bayes Assumption:

Features are *conditionally independent* given the label (i.e. $p(x_k \mid x_j, y) = p(x_k \mid y)$ for $k \neq j$).

$$p(x \mid y) = \prod_{j=1}^{d} p(x_j \mid y)$$

Naive Bayes Model

Parameterize $p(x_i \mid y)$ and $p(y)$

$$p(x \mid y) = \prod_{j=1}^{d} p(x_j \mid y)$$

For binary x_j, assume $p(x_j \mid y)$ follows Bernoulli distributions:

$$\theta_{j,1} := p(x_j = 1 \mid y = 1) \text{ and } \theta_{j,0} := p(x_j = 1 \mid y = 0)$$

Similarly, model $p(y)$ as Bernoulli:

$$\theta_1 := p(y = 1)$$

Therefore: $p(x, y) = p(x \mid y)p(y) = p(y)\prod_{j=1}^{d} p(x_j \mid y) = p(y)\prod_{j=1}^{d} \left(\theta_{j,y}\mathbf{1}\{x_j = 1\} + (1 - \theta_{j,y})\mathbf{1}\{x_j = 0\} \right).$

Naive Bayes Model

MLE for Naive Bayes

$$\mathscr{L}_n(\theta) = \sum_{i=1}^{n} \left(\log p(y^{(i)}; \theta) + \sum_{j=1}^{d} \log \left(\theta_{j,y^{(i)}} \mathbf{1}\{x_j^{(i)} = 1\} + \left(1 - \theta_{j,y^{(i)}}\right) \mathbf{1}\{x_j^{(i)} = 0\} \right) \right)$$

What parameters are there? $\theta_{j,0}, \theta_{j,1}$ for each $j \in [d]$ and θ_1.

$$\frac{\partial \mathscr{L}_n}{\partial \theta_{j,1}} = \frac{\partial}{\partial \theta_{j,1}} \sum_{i=1}^{n} \log \left(\theta_{j,y^{(i)}} \mathbf{1}\{x_j^{(i)} = 1\} + \left(1 - \theta_{j,y^{(i)}}\right) \mathbf{1}\{x_j^{(i)} = 0\} \right)$$

$$= \sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 1 \wedge x_j^{(i)} = 1\} \frac{1}{\theta_{j,1}} + \mathbf{1}\{y^{(i)} = 1 \wedge x_j^{(i)} = 0\} \frac{1}{1 - \theta_{j,1}} \quad \text{(zero terms where } y^{(i)} = 0\text{)}$$

Naive Bayes Model

MLE for Naive Bayes (solving for $\theta_{j,1}$)

Set $\frac{\partial \mathscr{L}_n}{\partial \theta_{j,1}}$ equal to zero:

$$0 = \sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 1 \wedge x_j^{(i)} = 1\} \frac{1}{\theta_{j,1}} + \mathbf{1}\{y^{(i)} = 1 \wedge x_j^{(i)} = 0\} \frac{1}{1 - \theta_{j,1}}$$

$$\implies \theta_{j,1} = \frac{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 1 \wedge x_j^{(i)} = 1\}}{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 1\}}$$

In practice, this is just counting words. If $j \in [d]$ corresponds to “free,” this is:

$$\frac{\text{number of spam emails containing "free"}}{\text{number of spam emails}}$$

Naive Bayes Model

MLE for all parameters

$$\implies \theta_{j,1} = \frac{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 1 \wedge x_j^{(i)} = 1\}}{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 1\}}$$

Can also show that the other parameters are:

$$\theta_{j,0} = \frac{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 0 \wedge x_j^{(i)} = 1\}}{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 0\}}$$

$$\theta_1 = \frac{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 1\}}{n}$$

Naive Bayes Model

MLE for all parameters

$$\theta_{j,1} = \frac{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 1 \wedge x_j^{(i)} = 1\}}{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 1\}}$$

$$\theta_{j,0} = \frac{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 0 \wedge x_j^{(i)} = 1\}}{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 0\}}$$

$$\theta_1 = \frac{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = 1\}}{n}$$

Together, these specify the joint distribution, under the Naive Bayes assumption:

$$p(x, y; \theta) = p(y) \prod_{j=1}^{d} \Big(\theta_{j,y} \mathbf{1}\{x_j = 1\} + (1 - \theta_{j,y}) \mathbf{1}\{x_j = 0\} \Big).$$

Naive Bayes Model

Summary

Together, these specify the joint distribution, under the Naive Bayes assumption:

$$p(x, y; \theta) = p(y) \prod_{j=1}^{d} \Big(\theta_{j,y} \mathbf{1}\{x_j = 1\} + (1 - \theta_{j,y}) \mathbf{1}\{x_j = 0\} \Big).$$

Naive Bayes assumption: features are *conditionally independent*, given label.

Recipe for **learning** Naive Bayes model:

1. Choose $p(x_j \mid y)$, e.g. Bernoulli distribution for binary x_j.
2. Choose $p(y)$, often some categorical distribution.
3. Estimate parameters via MLE (same strategy as conditional models).

Naive Bayes Model

Summary

Recipe for **learning** Naive Bayes model:

1. Choose $p(x_j \mid y)$, e.g. Bernoulli distribution for binary x_j.
2. Choose $p(y)$, often some categorical distribution.
3. Estimate parameters via MLE (same strategy as conditional models).

Recipe for **prediction** with Naive Bayes model:

$$p(y \mid x) = \frac{p(x \mid y)p(y)}{p(x)} \text{ by Bayes rule} \implies \operatorname*{argmax}_{y \in \mathcal{Y}} p(y \mid x) = \operatorname*{argmax}_{y \in \mathcal{Y}} p(x, y)$$

Compare: $p(x, y = 1; \theta) = p(y = 1) \prod_{j=1}^{d} \left(\theta_{j,1} \mathbf{1}\{x_j = 1\} + (1 - \theta_{j,1}) \mathbf{1}\{x_j = 0\} \right)$ vs. $p(x, y = 0; \theta)$.

Outline

Probabilistic Modeling

Review: Maximum Likelihood Estimation

Conditional Probability Model: Linear Regression

Conditional Probability Model: Logistic Regression

Generalized Linear Models

Generative Model: Naive Bayes