Lecture 6/11

- Recap Generative Models ⇒ Supervised Classification
  - Last time Gen Model (Gaussian) ⇒ GDA
  - Gen Models (?other dist) ⇒ ? Discriminant Analysis

- Naive Bayes
  - Smoothing estimates.

- HW2B demo

Generative model: fit the curve(params) to data $X_i \Rightarrow P(X)$
distrib

$P(X)$ ● Gaussian $\mathcal{N}(\mu, \Sigma)$: find μ, Σ such that $X \sim \mathcal{N}(\mu, \Sigma)$

$\mathcal{N}(\mu, \Sigma)$ generator for datapoints ⟸ appears distas...

INDEP N times, prob observing $X_1, X_2, \ldots X_N$ datapoints
given data

is as much as possible.

(Last time) $\Rightarrow \mu = \text{Mean}[X_{i=1:N}]$ $\Sigma = \text{COVAR}(X)$
sigma DXD

$P(X)$ ● model = Poisson $\Rightarrow$ #events per fixed interval (assuming rough constant)

λ param = expected AVG

" #bikes at stop per hour" $\rightarrow$ avg = 6

X = #events in an interval

" #Trump tweets a day"

$$P[X=k] = \frac{\lambda^k / k!}{e^{\lambda}}$$ normalizer

" # accidents in Boston per week"

$\sum_{k=0}^{\infty} \frac{\lambda^k}{k!}$ Taylor e^{λ} ⇒ need to divide with e^{λ} to sum=1

- multi-dim Poisson (D dim) ?

P(x) • exponential

P(x) • multinomial Roll die K faces (1, 2, 3 ... K) N throws

prop $p_1\ p_2\ p_3 \dots p_K$ $\sum p_i = 1$

outcome x_1 x_2 ... x_K observed $\sum x_k = N$

X #1 #2 #K

$$P(x_1, x_2 \dots x_K) = \binom{N}{x_1\ x_2 \dots x_K} p_1^{x_1} \cdot p_2^{x_2} \dots p_K^{x_K}$$

face counts

Multinom $\frac{N!}{x_1!\, x_2! \dots x_K!}$

Check

$$\sum_{x_1 + x_2 + \dots + x_K = N} \binom{N}{x_1 \dots x_K} \prod_{t=1}^{K} p_t^{x_t} = 1$$

Fit curve (param) to data ⇔ Find best params.

- gaussian : μ, Σ
- multinom : $p_1, p_2 \dots p_k$
- poisson : $\lambda_1, \lambda_2 \dots \lambda_k$

→ model of density

data density

$P(x_i)$ ← datapoint = probability to observe that point

$P(X) \bullet$ MIXTURE (Gaussian, Multinom / Poisson)

fixed weights

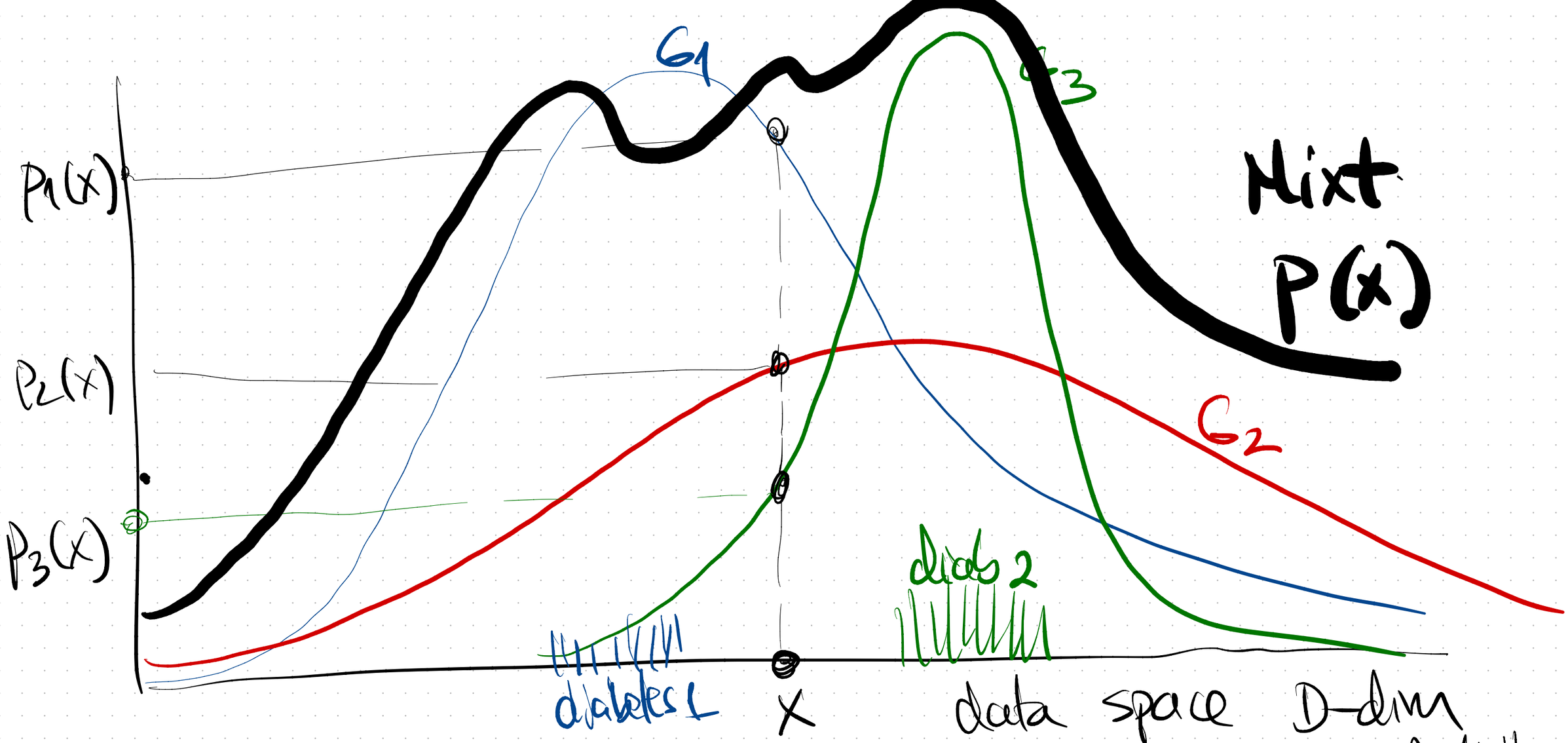
hw 3: Mixture of Gaussians

$k=3$ gaussians each D-dim

$$P(x) = \boxed{w_1} \bullet \underbrace{N_1(\mu_1, \Sigma_1)}_{P_1(x \mid \mu_1, \Sigma_1)} + \boxed{w_2} \underbrace{N_2(\mu_2, \Sigma_2)}_{P_2(x \mid \mu_2, \Sigma_2)} + \boxed{w_3} \underbrace{N_3(\mu_3, \Sigma_3)}_{P_3(x \mid \mu_3, \Sigma_3)}$$

D-dim

20% 50% 30%



Mixture adv: ability to fit data with multiple "hills" "bumps"

Generative Models ⇒ supervised classification.

① For each class $y_1, y_2 \cdots y_L$ (L labels)

fit a model on X that class

$P_1(x) = p(x|y=1)$ model for class $y=1$

$P_2(x) = p(x|y=2)$ model for class $y=2$ → fit model only on data $X_{(y=2)}$

$P_L(x) = p(x|y=L)$ model for class $y=L$

→ not cond dist. X = filtered by class

gauss ← not. nec. same model → exp

models trained per class

② z = test point ⇒ predict label $y_z \in \{1, 2, \ldots L\}$

$$P(y|z) = \frac{P(z|y) \cdot P(y)}{P(z)}$$

Predict ARGMAX

prior (labels)

calculate for each $y = l$

$$P(y=l|z) = \frac{P(z|y_l) \cdot P(y=y_l)}{\text{normalized}}$$

predict: $P(Y=\ell|z)$ = confidence label weighted by $Y=\ell$

over all $\ell=1:L$ classes

Naive Bayes = same prediction method

naive assumption about D-dim.

D-dim z = test point = $(z^1 z^2 \dots z^D)$

prediction $P(Y=c|z) = \frac{P(z|Y) \cdot P(Y)}{\text{normalized } P(z)}$ → prior

$P(Y)$ → prior

normalized $P(z)$

⇒ need $P(z|Y) = P(z^1, z^2, \dots z^D | Y)$

[before $P(z^1 z^2 \dots z^D | Y) \overset{\text{Fit}}{\sim} \mathcal{N}(z^{D\text{-dim}} | \mu, \Sigma)$

D-dim curve

NAÏVE

Now: assume feat 1,2..D relatively (??) independent

$P(z^1 z^2 \dots z^D | Y) = P(z^1|Y)\, P(z^2|Y) \dots P(z^D|Y)$

ex: z = person z^1 = height z^2 = weight z^3 = shoe size

Independent? NO **assumption wrong!**

but classifier still works ⇒ produces output

Fit $P(z \mid y=c)$ ⇒ Fit separately every feature (1-dim)

class $y=c$ — D-dim

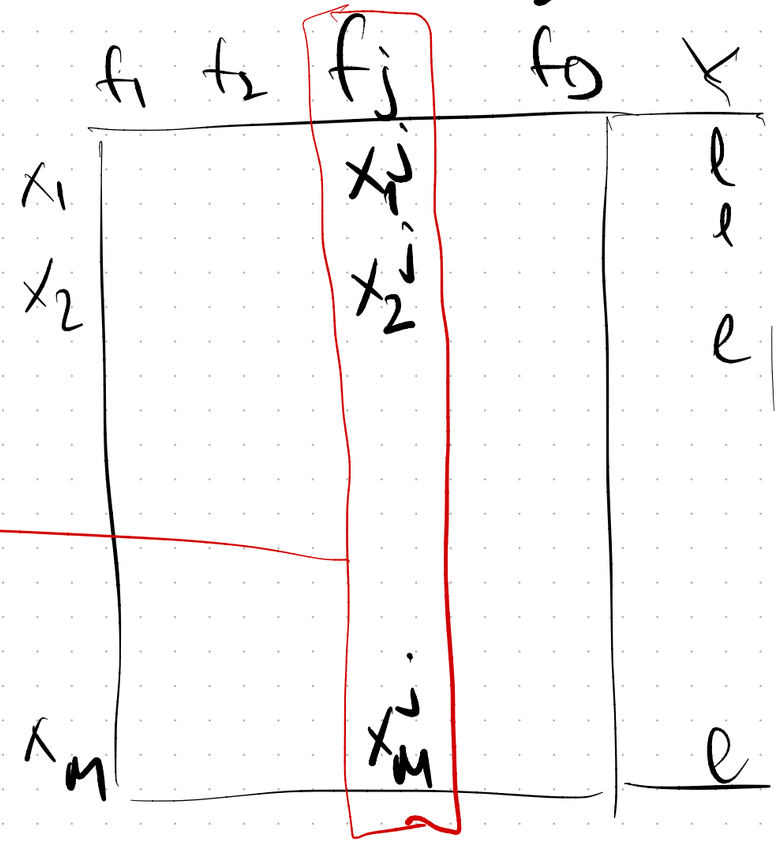
TRAINING: (like before, separate for each class)

class $y=c$ ⇒ FILTER X data $= X_{(y=c)}$

- for each feature $j = 1:D$ → separately

Fit 1-dim $P(x^j) \simeq$

vector $x^j_{1:M}$



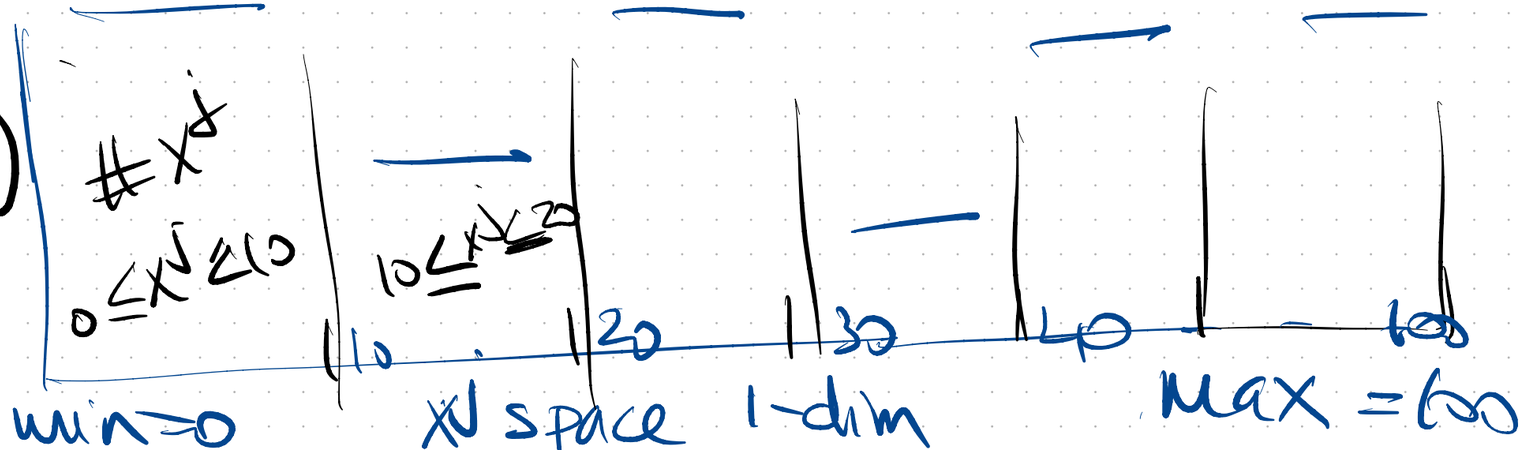
2 options

→ fit parametric curve (1-dim)

$P(x^j) \simeq \mathcal{N}(\mu, \sigma^2)$ 1-dim

no assumptions → fit histogram

- buckets ($B=10$)
- count per bucket
- normalized



ex: x^1 = weight

buckets	B_1: 0–40LB	B_2: 40LB–60LB	B_3: 60–90	— — —	B_{last} ⩾ 250LB
	count 1 / num	count 2 / num	counts		count last

More granular ⇒ more buckets
requires more data for accuracy

product for Z = test point = $(z^1, z^2 \dots z^D)$

$$P(Y|Z) = \frac{P(Z|Y)\,P(Y)}{P(Z)} = \frac{P(z^1|Y)\,P(z^2|Y) \dots P(z^D|Y) \cdot P(Y)}{P(Z)}$$

trained

num product very small

use logs

$$\log(P(Y|Z)) = \log(P(Y)) + \sum_{d=1}^{D} \log(P(z^d|Y)) - \log P(Z)$$

Naive Bayes for Text using Bag-of-Words representation.

- class $y=e$ as filter $X = X_{[y=e]}$ data with label $y=e$ "china"
- want fit $P(x) = P(x \mid y=e)$

$$P(x) = P(x^1) \cdot P(x^2) \cdot \ldots \cdot P(x^j) \cdot \ldots \cdot P(x^D)$$

- fit each separately

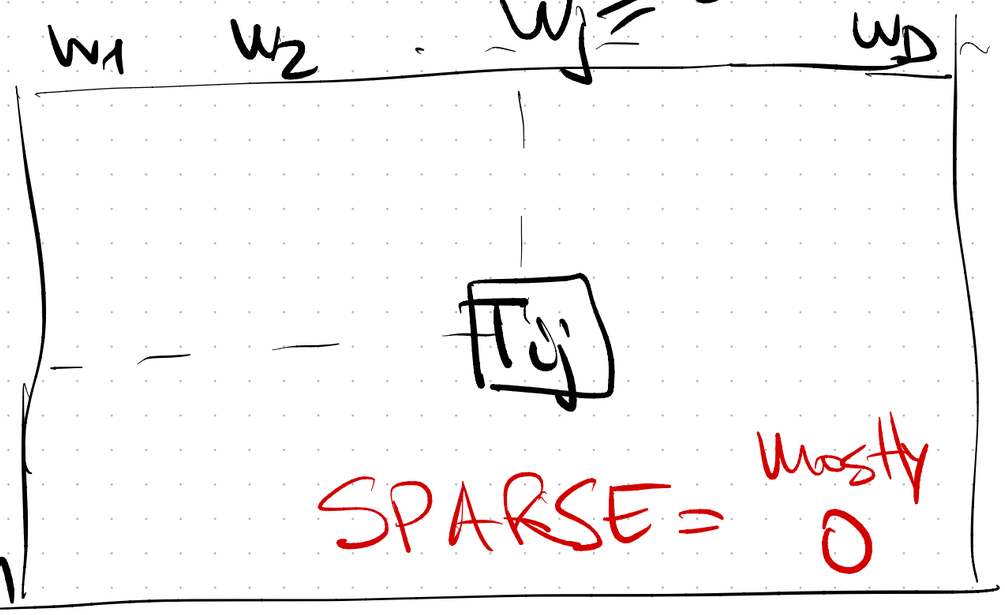
For feature word$_j$ = "china"

counts $\Rightarrow$ distribution (1-dim)

x_1^j
x_2^j
x_M^j

$$\text{OBVIOUS} = P(x_i^j) = \frac{TF_{ij}}{\#\text{words in doc } i}$$

$$= \frac{TF}{\text{DOCLENGTH}}$$



$T_{ij} = TF_{ij} = \text{term freq}$ (word j in doc i)

$= \#\text{occ of word}_j$

useless words: "for" "a" "about" "the" "in"

"STOPWORDS" $\Rightarrow$ exclude

VISUALIZE FOR ADOC x_i

	w_1	w_2		w_j		w_D
x_i:	x_i^1	x_i^2	...	x_i^j		x_i^D

$\Rightarrow$

term freq in doc x_i			
$\frac{x_i^1}{\lvert x_i \rvert}$	$\frac{x_i^2}{\lvert x_i \rvert}$	...	$\frac{x_i^D}{\lvert x_i \rvert}$

$\text{doclength}(x_i) = |x_i| = \#\text{words in doc}$

discrete (counts)

very sparse (many 0)

- dont like prob = 0 ! Fix = "smoothing"

	w^1	w^2		w^j		w^D
$x_i =$	x_i^1	x_i^2				x_i^D

$\Rightarrow$

LAPLACE: add +1 numerators, add +D denomin

$\frac{x_i^1+1}{\lvert x_i \rvert + D}$	$\frac{x_i^2+1}{\lvert x_i \rvert + D}$	...	$\frac{x_i^D+1}{x_i+D}$

LAPLACE
- do not have prob = 0 (min prob = $\frac{1}{D}$)
- prior for words in doc if they do not actually occur $\frac{1}{|x_i|+D}$
- deal with words w_{new} new in TEST SET (never seen in TRAIN)

- math nice, including for loops
- general: $\frac{+\varepsilon \text{ to numerator}}{+D\varepsilon \text{ to denominators}}$ => ε = strength of prior
  $\varepsilon = 1$ default in text
- still a probab (sum = 1)

---

prod = 0 => BAD

- $p(x) = p(x^1) \cdot p(x^2) \cdot \ldots \cdot \boxed{p(x^i)} \cdot \ldots \cdot p(x^D) = 0$
  → makes all other features irrelevant.

- never seen x^i word in training
  $\frac{0}{0}$ in overall formula for each class Y_e.