Dimensionality

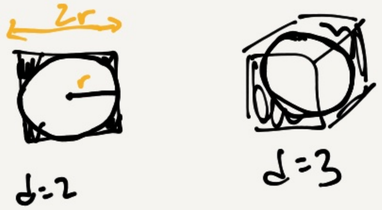
The quality of BDR depends on the class conditional densities (CCD) estimates.

- How does it work in high-dimension?

"High dimensional spaces are weird."

Do not trust your intuition.

Example 1) consider a hypercube & an inscribed hypersphere in $\mathbb{R}^d$.



Volume of hypersphere: $V_d(r) = \frac{\pi^{d/2} r^d}{\Gamma(\frac{d}{2}+1)}$

$\Gamma(n) = \int_0^\infty e^{-x} x^{n-1} dx$ (Gamma function)

$\approx$ factorial for real numbers.

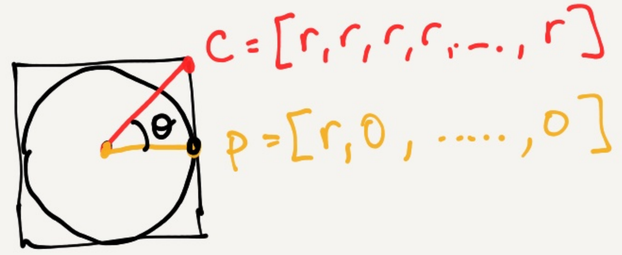
$\Gamma(n+1) = n!$

Volume of hypercube: $(2r)^d$

let $f_d = \left(\frac{\text{volume of h-sphere}}{\text{volume of h-cube}}\right) = \frac{\pi^{d/2}}{2^d \Gamma(\frac{d}{2}+1)}$

as d increases, $f_d \to 0$.

$\therefore$ the volume of the corners increases.



$\|c\|^2 = d r^2$

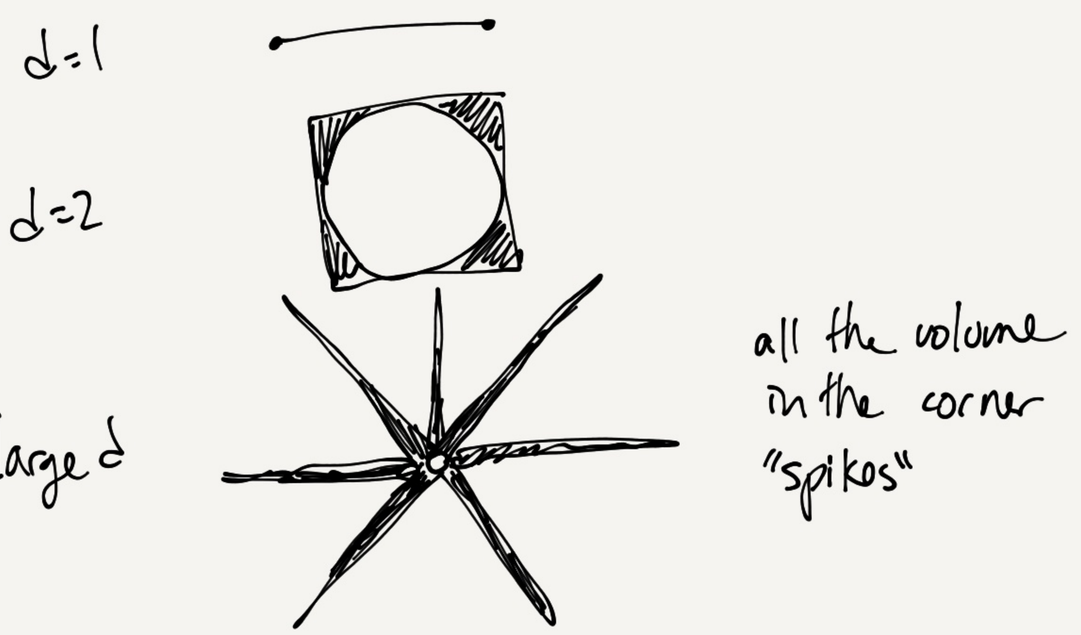
$\|p\|^2 = r^2$

$\cos\theta = \frac{c^T p}{\|c\|\|p\|} = \frac{r^2}{\sqrt{d}\cdot r\, r} = \frac{1}{\sqrt{d}}$

As d increases, then $\cos\theta \to 0$

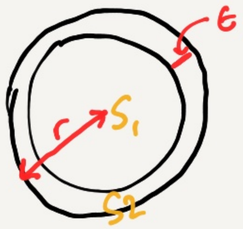
i.e. $c \perp p$ (corner is orthogonal to the axis)

corner ↑ axes ↑



all the volume in the corner "spikes"

Example 2) consider a hypersphere shell of thickness ϵ.



volume of hypershell

$V_{shell} = V(S_2) - V(S_1)$

$= \left(1 - \frac{V(S_1)}{V(S_2)}\right) V(S_2)$

$\frac{V(S_1)}{V(S_2)} = \cdots\cdots = \left(1 - \frac{\epsilon}{r}\right)^d$ ($1 - \frac{\epsilon}{r} < 1$)

Suppose $0 < \epsilon < r$: As d increases, $\frac{V(S_1)}{V(S_2)} \to 0$

Thus: $V_{shell} \to V(S_2)$

"all the volume is in the shell!!!" (?)

Example 3: high-dim Gaussian.

m.v. Gaussian $X \sim N(0, \sigma^2 I_d)$

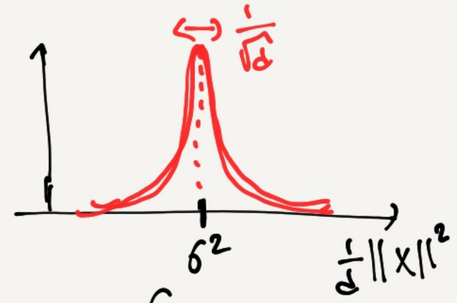
thus $x_i \sim N(0, \sigma^2)$ (iid r.v)

Then: $E\left[\frac{1}{d}\|X\|^2\right] = \frac{1}{d} E\left[\underbrace{x_1^2 + x_2^2 + \cdots x_d^2}_{d \text{ terms}}\right] = \underbrace{\sigma^2}$

Note: $\|X\|^2 = \underbrace{\sum_i \underbrace{x_i^2}_{\text{r.v.}}}_{\text{sum of r.v.}} \to$ sum of iid r.v.

So by the central limit theorem, $\|X\|^2$ is concentrated around the mean as $d \to \infty$

$\Rightarrow \frac{1}{d}\|X\|^2 \sim N(\sigma^2, \frac{1}{d})$

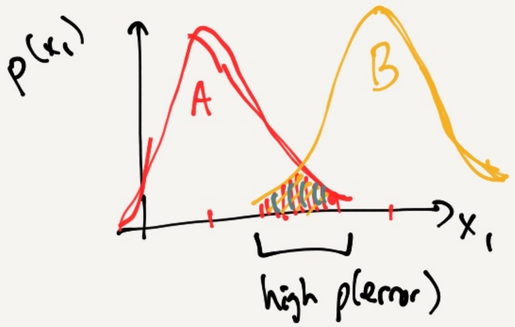


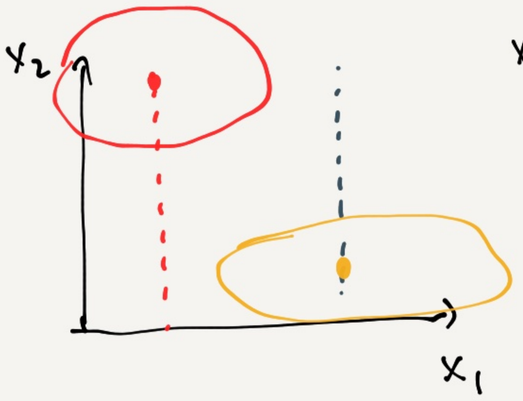
In high-dim, a Gaussian is a shell of radius $\sigma\sqrt{d}$ — most of the distribution is inside the shell.

(the max density value is still the mean)

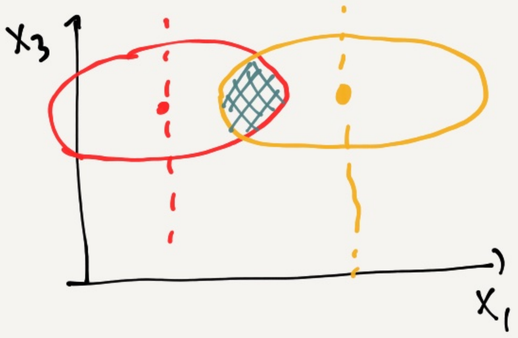
Curse of dimensionality

In theory, adding new features will not hurt the classifier (i.e. increase p(error))





x_2 is an informative feature
→ p(error) decrease



x_3 is uninformative,
→ p(error) is the same as before.

But: in practice, for BDR the error increases as we add more features (add dimensions to the input)

The problem: quality of the CD estimates.

density estimates in high-dim require more training data!

e.g. histogram in d-dim over unit cube $[0,1]^d$
⇒ 10 bins per dimension ⇒ 10^d bins overall

to just have 1 sample per bin, we need 10^d samples → increases exponentially w/ the dimension.

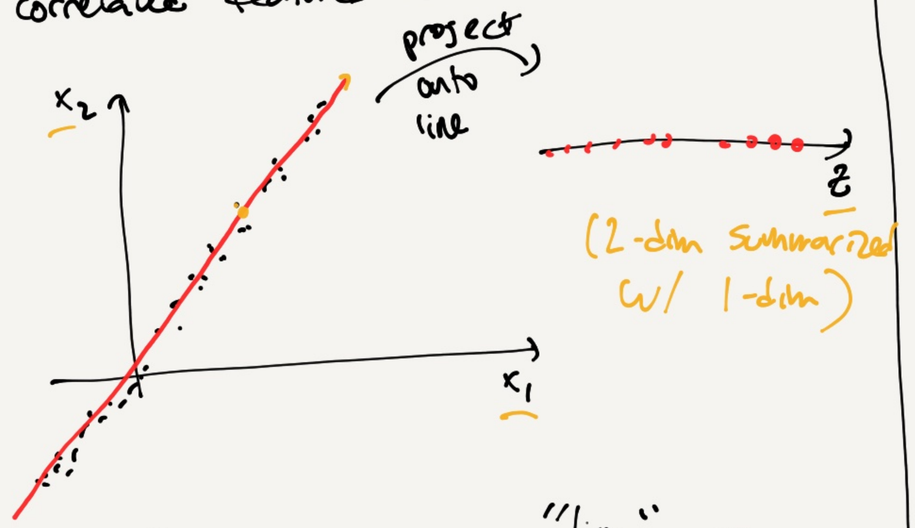
Solutions: 1) Reduce # parameters (complexity of model)
e.g. full cov → diagonal cov.

✓ 2) Reduce # of features (dimensionality reduction)
⇒ implicitly reduce # of parameters.

3) "Create" more data
a) virtual samples (Bayesian estimation)
b) data augmentation (apply x-forms to the data that keep the class the same)

Linear Dimensionality Reduction

- summarize correlated features w/ a set of fewer features.
- What do correlated features look like?



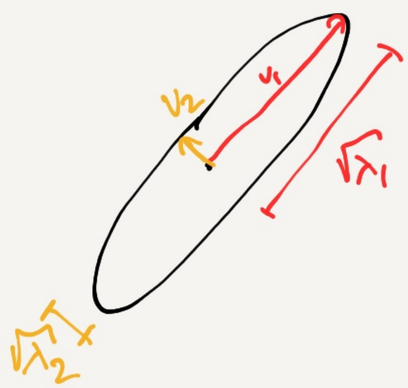
- the correlated features live on a "line" or subspace (w/ some noise)

Principal Component Analysis (PCA)

Idea: if the data lives in a low-dim subspace, then it should be "flat" in the full space.

⇒ if we fit a Gaussian → it will be highly skewed (skinny ellipses)

- let (v_i, λ_i) be an eigenvector/eigenvalue pair of cov Σ.
  - each v_i defines an axis of ellipse
  - each λ_i defines the width



Thus the eigenvalues tell us which directions are flat.

⇒ select v_i w/ largest eigenvalue for the projection

PCA: Given dataset $\{x_1, \dots, x_N\}$ of dim k.

Learning:

1) calculate Gaussian:

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i \quad , \quad \Sigma = \frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)(x_i-\mu)^T$$

2) eigen decomposition of $\Sigma = V \Lambda V^T$

$V = [v_1 \cdots v_d]$ = matrix of eigenvectors

$\Lambda = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_d \end{bmatrix}$ = diagonal matrix of eigenvalues.

3) Sort eigenvalues: $\lambda_1 > \lambda_2 > \lambda_3 > \dots \geq 0$

4) Select top-k eigenvectors: $\Phi = [v_1 \cdots v_k]$ ← Principal components

dim. reduction:

5) project point x onto Φ space:

$$z = \Phi^T(x-\mu)$$ ← "PCA coefficients"

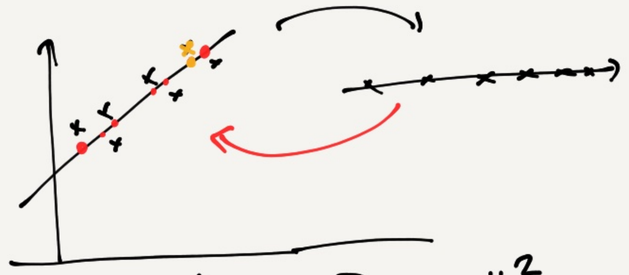
z is the new feature, apply BDR as before

Notes:

This selection of Φ w/ $\Phi^T\Phi = I$:

1) maximize the variance of the projected training data
i.e $\sum_i \|z_i\|^2$ (PS 7.3)

2) minimizes the reconstruction error of the training data (PS 7.2)



$$\sum_i \|x_i - \Phi z_i + \mu\|^2$$

reconstruction $\hat{x}_i$

3) for high-dim data, can use SVD for an efficient implementation. (PS7-4)

4) Selecting k: 1) pick a value of k (that works for your problem)

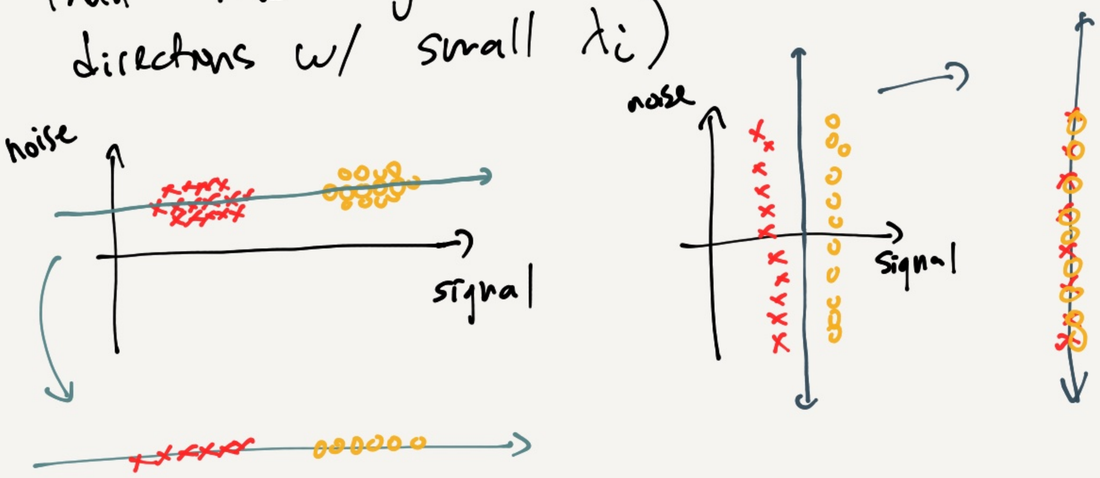
2) pick k to preserve p% of the variance

$$p = \frac{\sum_{i=1}^{k}\lambda_i}{\sum_{i=1}^{d}\lambda_i}$$

$\sum_{i=1}^{k}\lambda_i$ ← variance preserved

$\sum_{i=1}^{d}\lambda_i$ ← total variance

5) assume the "noise" variance is smaller than the "signal" variance. (throw away directions w/ small λ_i)



⇒ has consequences on classification.

6) PCA is optimal for representation (preserving variance) but not necessarily classification.

- there is no way to fix this because we don't have the classes!

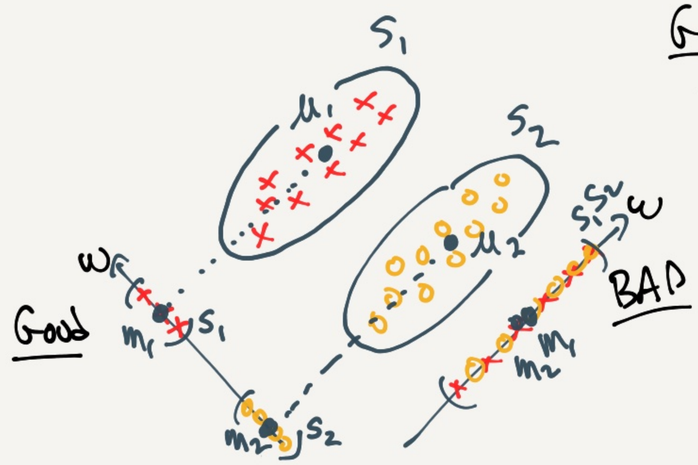
⇒ Fix this using the class information when doing dim. reduction.

Fisher's Linear Discriminant (FLD)

(Linear discriminant analysis (LDA))

Goal: find the projection that best separates the classes.

$z = w^T x$



Class Statistics

	original space	1-d space
class mean:	$\mu_j = \frac{1}{N_j} \sum_{x_i \in C_j} x_i$	$m_j = w^T \mu_j$
scatter matrix:	$S_j = \sum_{x_i \in C_j} (x_i - \mu_j)(x_i - \mu_j)^T$	$s_j = w^T S_j w$

Idea: maximize the distance between the projected means: $(m_1 - m_2)^2 = (w^T(\mu_1 - \mu_2))^2$

problem: w is unconstrained → need normalization.

Fisher's Idea

$$w^* = \underset{w}{\operatorname{argmax}} \frac{(m_1 - m_2)^2}{S_1 + S_2}$$

$(m_1 - m_2)^2$: between-class scatter

$S_1 + S_2$: within-class scatter

$$= \underset{w}{\operatorname{argmax}} \frac{w^T S_B w}{w^T S_W w}$$

$$\begin{cases} S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T \\ S_W = S_1 + S_2 \end{cases}$$

⋮ generalized eigenvalue problem (tutorial)

← Fisher's Linear Discriminant

$$w^* = (S_1 + S_2)^{-1}(\mu_1 - \mu_2)$$

Note: this also defines the decision boundary of a Gaussian classifier w/ cov $\frac{1}{N}(S_1 + S_2)$.

$\Rightarrow$ FLD is optimal when the 2 classes are Gaussian w/ equal covariance.