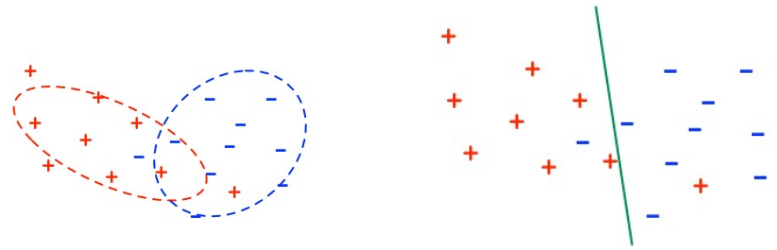
Classification with generative models

CSE 250B

Classification with parametrized models

Classifiers with a fixed number of parameters can represent a limited set of functions. Learning a model is about picking a good approximation.

Typically the x's are points in p-dimensional Euclidean space, $\mathbb{R}^p$.



Two ways to classify:

- **Generative**: model the individual classes.
- **Discriminative**: model the decision boundary between the classes.

Quick review of conditional probability

Formula for conditional probability: for any events A, B,

$$\Pr(A|B) = \frac{\Pr(A \cap B)}{\Pr(B)}.$$

Applied twice, this yields Bayes' rule:

$$\Pr(H|E) \;=\; \frac{\Pr(E|H)}{\Pr(E)}\Pr(H).$$

Example: Toss ten coins. What is the probability that the first is heads, given that nine of them are heads?

$$H = \text{first coin is heads}$$
$$E = \text{nine of the ten coins are heads}$$

$$\Pr(H|E) = \frac{\Pr(E|H)}{\Pr(E)} \cdot \Pr(H) = \frac{\binom{9}{8}\frac{1}{2^9}}{\binom{10}{9}\frac{1}{2^{10}}} \cdot \frac{1}{2} = \frac{9}{10}$$

Summation rule

Suppose events $A_1, \ldots, A_k$ are disjoint events, one of which must occur. Then for any other event E,

$$\begin{aligned}\Pr(E) &= \Pr(E, A_1) + \Pr(E, A_2) + \cdots + \Pr(E, A_k) \\ &= \Pr(E|A_1)\Pr(A_1) + \Pr(E|A_2)\Pr(A_2) + \cdots + \Pr(E|A_k)\Pr(A_k)\end{aligned}$$

Example: Sex bias in graduate admissions. In 1969, there were 12673 applicants for graduate study at Berkeley. 44% of the male applicants were accepted, and 35% of the female applicants.

Over the sample space of applicants, define:

$$M = \text{male}$$
$$F = \text{female}$$
$$A = \text{admitted}$$

So: $\Pr(A|M) = 0.44$ and $\Pr(A|F) = 0.35$.

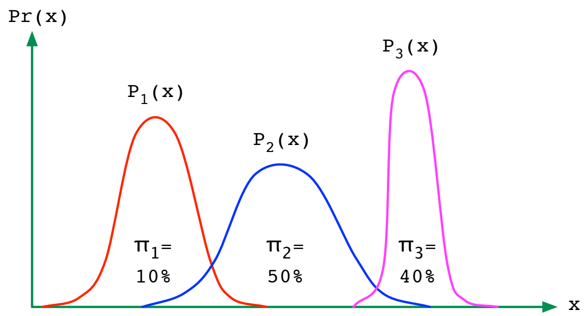
In every department, the accept rate for female applicants was at least as high as the accept rate for male applicants. How could this be?

Generative models

An unknown underlying distribution D over $\mathcal{X} \times \mathcal{Y}$.
Generating a point (x, y) in two steps:

1. Last week: first choose x, then choose y given x.
2. Now: first choose y, then choose x given y.

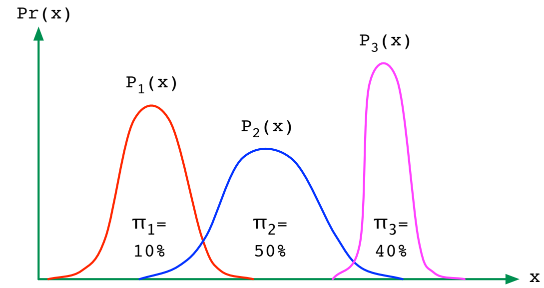
Example:
$\mathcal{X} = \mathbb{R}$
$\mathcal{Y} = \{1, 2, 3\}$



The overall density is a mixture of the individual densities,

$$\Pr(x) = \pi_1 P_1(x) + \cdots + \pi_k P_k(x).$$

The Bayes-optimal prediction



Labels $\mathcal{Y} = \{1, 2, \ldots, k\}$, density $\Pr(x) = \pi_1 P_1(x) + \cdots + \pi_k P_k(x)$.

For any $x \in \mathcal{X}$ and any label j,

$$\Pr(y = j|x) = \frac{\Pr(y = j)\Pr(x|y = j)}{\Pr(x)} = \frac{\pi_j P_j(x)}{\sum_{i=1}^k \pi_i P_i(x)}$$

Bayes-optimal prediction: $h^*(x) = \arg\max_j \pi_j P_j(x)$.

Estimating the π_j is easy. Estimating the P_j is hard.

Estimating class-conditional distributions

Estimating an arbitrary distribution in $\mathbb{R}^p$:

- Can be done, e.g. with kernel density estimation.
- But number of samples needed is exponential in p.

Instead: approximate each P_j with a simple, parametric distribution.

Some options:

- Product distributions.
  Assume coordinates are independent: naive Bayes.
- Multivariate Gaussians.
  Linear and quadratic discriminant analysis.
- More general graphical models.

Naive Bayes

Labels $\mathcal{Y} = \{1, 2, \ldots, k\}$, density $\Pr(x) = \pi_1 P_1(x) + \cdots + \pi_k P_k(x)$.

Binarized MNIST:

- $k = 10$ classes
- $\mathcal{X} = \{0, 1\}^{784}$

Assume that **within each class**, the individual pixel values are independent:

$$P_j(x) = P_{j1}(x_1) \cdot P_{j2}(x_2) \cdots P_{j,784}(x_{784}).$$

Each P_{ji} is a coin flip: trivial to estimate!

Smoothed estimate of coin bias

Pick a class j and a pixel i. We need to estimate

$$p_{ji} = \Pr(x_i = 1 | y = j).$$

Out of a training set of size n,

$$\begin{aligned} n_j &= \#\text{ of instances of class } j \\ n_{ji} &= \#\text{ of instances of class } j \text{ with } x_i = 1 \end{aligned}$$

Then the maximum-likelihood estimate of p_{ji} is

$$\widehat{p}_{ji} = n_{ji}/n_j.$$

This causes problems if $n_{ji} = 0$. Instead, use "Laplace smoothing":

$$\widehat{p}_{ji} = \frac{n_{ji}+1}{n_j+2}.$$

Form of the classifier

Data space $\mathcal{X} = \{0,1\}^p$, label space $\mathcal{Y} = \{1,\ldots,k\}$. Estimate:

- $\{\pi_j : 1 \le j \le k\}$
- $\{p_{ji} : 1 \le j \le k, 1 \le i \le p\}$

Then classify point x as

$$\arg\max_j \quad \pi_j \prod_{i=1}^p p_{ji}^{x_i}(1-p_{ji})^{1-x_i}.$$

To avoid underflow: take the log:

$$\arg\max_j \quad \underbrace{\log \pi_j + \sum_{i=1}^p \left(x_i \log p_{ji} + (1-x_i)\log(1-p_{ji})\right)}_{\text{of the form } w \cdot x + b}$$

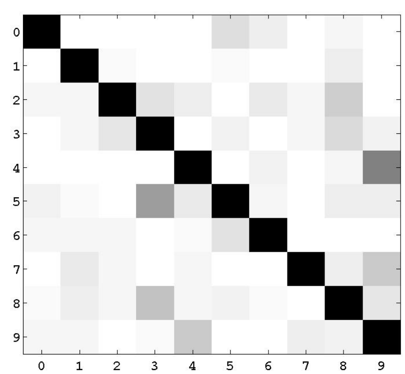
A linear classifier!

Example: MNIST

Result of training: mean vectors for each class.

Test error rate: 15.54%.

Visualization of the "confusion matrix" $\longrightarrow$



Other types of data

How would you handle data:

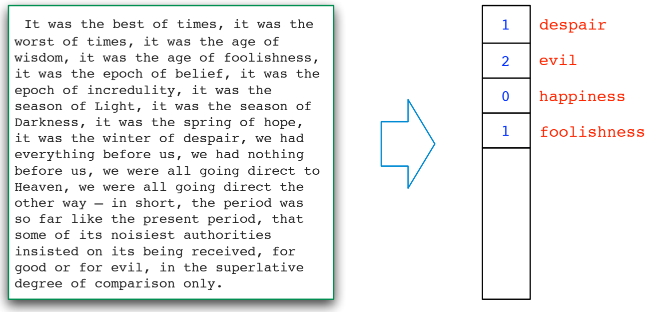
- Whose features take on more than two discrete values (such as ten possible colors)?
- Whose features are real-valued?
- Whose features are positive integers?
- Whose features are mixed: some real, some Boolean, etc?

How would you handle "missing data": situations in which data points occasionally (or regularly) have missing entries?

- At train time: **???**
- At test time: **???**

Handling text data

Bag-of-words: vectorial representation of text documents.



- Fix V = some vocabulary.
- Treat each document as a vector of length $|V|$:

$$x = (x_1, x_2, \ldots, x_{|V|}),$$

where x_i = # of times the ith word appears in the document.

A standard distribution over such document-vectors x: the **multinomial**.

Multinomial naive Bayes

Multinomial distribution over a vocabulary V:

$$p = (p_1, \ldots, p_{|V|}), \quad \text{such that } \ p_i \geq 0 \text{ and } \sum_i p_i = 1$$

Document $x = (x_1, \ldots, x_{|V|})$ has probability $p_1^{x_1} p_2^{x_2} \cdots p_{|V|}^{x_{|V|}}$.

For naive Bayes: one multinomial distribution per class.

- Class probabilities $\pi_1, \ldots, \pi_k$
- Multinomials $p^{(1)} = (p_{11}, \ldots, p_{1|V|}), \ldots, p^{(k)} = (p_{k1}, \ldots, p_{k|V|})$

Classify document x as

$$\arg\max_j \quad \pi_j \prod_{i=1}^{|V|} p_{ji}^{x_i}.$$

(As always, take log to avoid underflow: linear classifier.)

Improving performance of multinomial naive Bayes

A variety of heuristics that are standard in text retrieval, such as:

1. Compensating for burstiness.
   Problem: Once a word has appeared in a document, it has a much higher chance of appearing again.

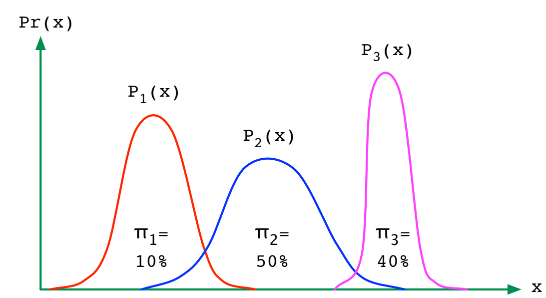
   Solution: Instead of the number of occurrences f of a word, use $\log(1+f)$.

2. Downweighting common words.
   Problem: Common words can have a unduly large influence on classification.

   Solution: Weight each word w by **inverse document frequency**:

$$\log \frac{\# \text{ docs}}{\#(\text{docs containing } w)}$$

Recall: generative model framework



Labels $\mathcal{Y} = \{1, 2, \ldots, k\}$, density $\Pr(x) = \pi_1 P_1(x) + \cdots + \pi_k P_k(x)$.

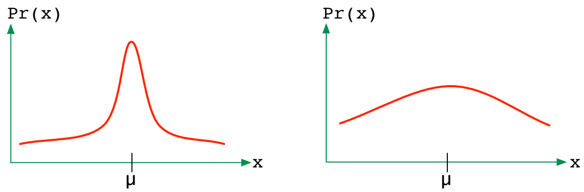
Approximate each P_j with a simple, parametric distribution:

- Product distributions.
  Assume coordinates are independent: naive Bayes.
- Multivariate Gaussians.
  Linear and quadratic discriminant analysis.
- More general graphical models.

Variance

If you had to summarize the entire distribution of a r.v. X by a single number, you would use the mean (or median). Call it μ.

But these don't capture the *spread* of X:



What would be a good measure of spread? How about the average distance away from the mean: $\mathbb{E}(|X-\mu|)$?

For convenience, take the square instead of the absolute value.

Variance: $\quad \text{var}(X) = \mathbb{E}(X-\mu)^2 = \mathbb{E}(X^2) - \mu^2,$

where $\mu = \mathbb{E}(X)$. The variance is always ≥ 0.

Variance: example

Recall: $\text{var}(X) = \mathbb{E}(X-\mu)^2 = \mathbb{E}(X^2) - \mu^2$, where $\mu = \mathbb{E}(X)$.

Toss a coin of bias p. Let $X \in \{0,1\}$ be the outcome.

$$\begin{aligned} \mathbb{E}(X) &= p \\ \mathbb{E}(X^2) &= p \\ \mathbb{E}(X-\mu)^2 &= p^2 \cdot (1-p) + (1-p)^2 \cdot p = p(1-p) \\ \mathbb{E}(X^2) - \mu^2 &= p - p^2 = p(1-p) \end{aligned}$$

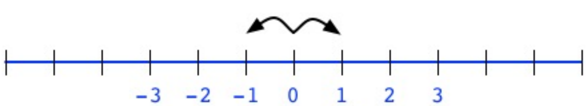
This variance is highest when $p = 1/2$ (fair coin).

The standard deviation of X is $\text{std}(X) = \sqrt{\text{var}(X)}$.
It is the average amount by which X differs from its mean.

Variance of a sum

$\text{var}(X_1 + \cdots + X_k) = \text{var}(X_1) + \cdots + \text{var}(X_k)$ if the X_i are independent.

Symmetric random walk. A drunken man sets out from a bar. At each time step, he either moves one step to the right or one step to the left, with equal probabilities. Roughly where is he after n steps?



Let $X_i \in \{-1, 1\}$ be his ith step. Then $\mathbb{E}(X_i) = ?0$ and $\text{var}(X_i) = ?1$.

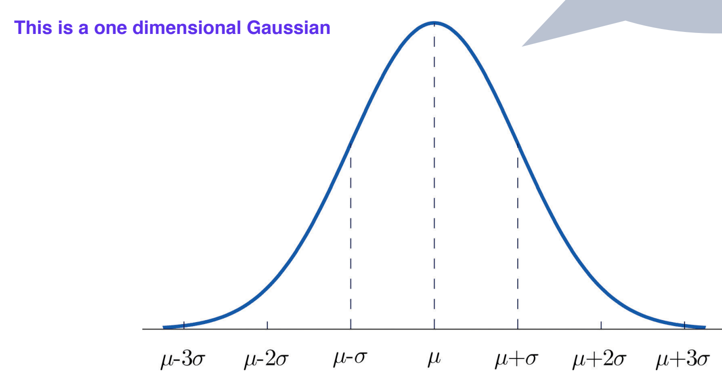
His position after n steps is $X = X_1 + \cdots + X_n$.

$$\begin{aligned} \mathbb{E}(X) &= 0 \\ \text{var}(X) &= n \\ \text{stddev}(X) &= \sqrt{n} \end{aligned}$$

He is likely to be pretty close to where he started!

The univariate Gaussian

When you want to use the correlation between X, use the Gaussian



The Gaussian $N(\mu, \sigma^2)$ has mean μ, variance σ^2, and density function

$$p(x) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

But what if we have **two** variables?

How many standard deviations from the mean are you?

Bivariate distributions

When you are multi-dimensional variables...

Simplest option: treat each variable as independent.

Example: For a large collection of people, measure the two variables

$$H = \text{height}$$
$$W = \text{weight}$$

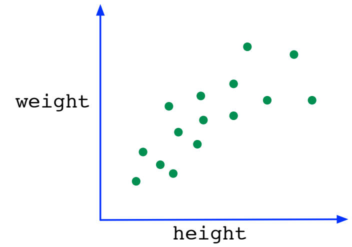
Independence would mean

$$\Pr(H = h, W = w) = \Pr(H = h)\Pr(W = w),$$

which would also imply $\mathbb{E}(HW) = \mathbb{E}(H)\mathbb{E}(W)$.

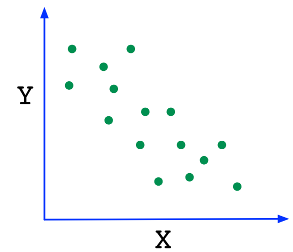
Is this an accurate approximation?
No: we'd expect height and weight to be **positively correlated**.

Types of correlation

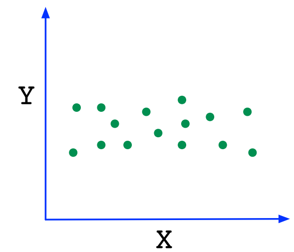


H, W positively correlated.
This also implies

$$\mathbb{E}(HW) > \mathbb{E}(H)\mathbb{E}(W).$$

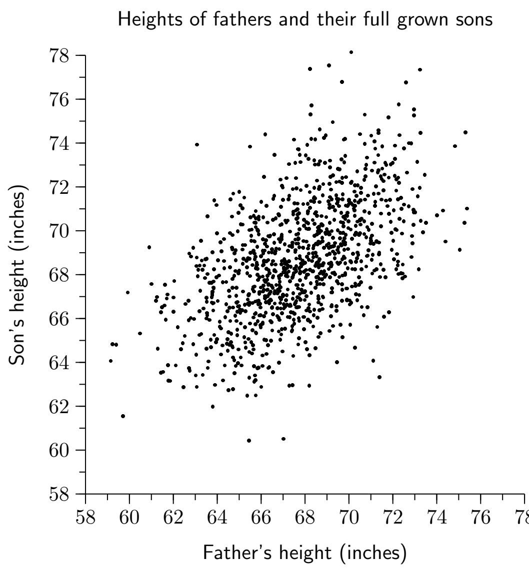


X, Y negatively correlated



X, Y uncorrelated

Pearson (1903): fathers and sons

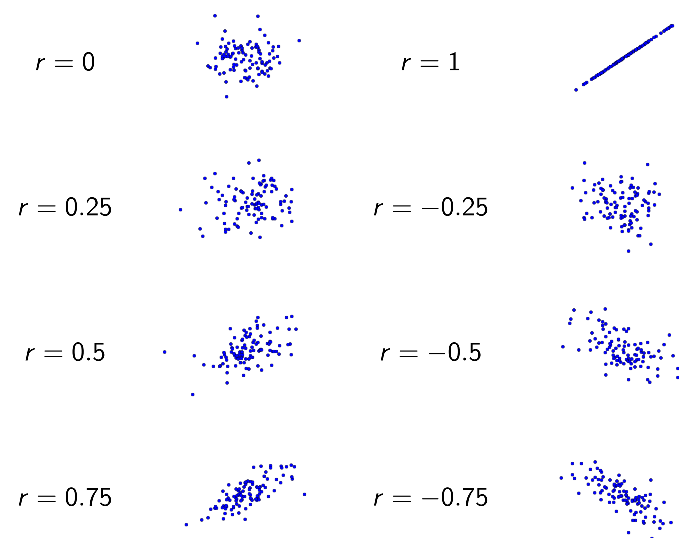


How to quantify the degree of correlation?

This is called the correlation coefficient

This is called the correlation coefficient
This is called the correlation coefficient

Correlation pictures



Covariance and correlation

Suppose X has mean μ_X and Y has mean μ_Y.

- Covariance

$$\text{cov}(X, Y) = \mathbb{E}[(X - \mu_X)(Y - \mu_Y)] = \mathbb{E}[XY] - \mu_X \mu_Y$$

  Maximized when $X = Y$, in which case it is $\text{var}(X)$.
  In general, it is at most $\text{std}(X)\text{std}(Y)$.

- Correlation

$$\text{corr}(X, Y) = \frac{\text{cov}(X, Y)}{\text{std}(X)\text{std}(Y)}$$

  This is always in the range $[-1, 1]$.

The correlation says that there is some type of dependence between X&Y

The co-variance is the maximum magnitude of the measure of the correlation.

Covariance and correlation: example 1

$$\text{cov}(X, Y) = \mathbb{E}[(X - \mu_X)(Y - \mu_Y)] \;=\; \mathbb{E}[XY] - \mu_X \mu_Y$$

$$\text{corr}(X, Y) = \frac{\text{cov}(X, Y)}{\text{std}(X)\text{std}(Y)}$$

x	y	$\Pr(x, y)$
-1	-1	$1/3$
-1	1	$1/6$
1	-1	$1/3$
1	1	$1/6$

$$\mu_X = 0$$
$$\mu_Y = -1/3$$
$$\text{var}(X) = 1$$
$$\text{var}(Y) = 8/9$$
$$\text{cov}(X, Y) = 0$$
$$\text{corr}(X, Y) = 0$$

In this case, X, Y are independent. Independent variables always have zero covariance and correlation.

Covariance and correlation: example 2

$$\text{cov}(X, Y) = \mathbb{E}[(X - \mu_X)(Y - \mu_Y)] \;=\; \mathbb{E}[XY] - \mu_X \mu_Y$$

$$\text{corr}(X, Y) = \frac{\text{cov}(X, Y)}{\text{std}(X)\text{std}(Y)}$$

x	y	$\Pr(x, y)$
-1	-10	$1/6$
-1	10	$1/3$
1	-10	$1/3$
1	10	$1/6$

$$\mu_X = 0$$
$$\mu_Y = 0$$
$$\text{var}(X) = 1$$
$$\text{var}(Y) = 100$$
$$\text{cov}(X, Y) = -10/3$$
$$\text{corr}(X, Y) = -1/3$$

In this case, X and Y are negatively correlated.

The bivariate (2-d) Gaussian

A distribution over $(x, y) \in \mathbb{R}^2$, parametrized by:

- **Mean** $(\mu_x, \mu_y) \in \mathbb{R}^2$
- **Covariance matrix**

Variance of X

Variance of Y

CoVariance

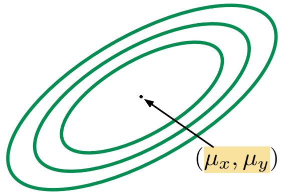
$$\Sigma = \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix}$$

  where $\Sigma_{xx} = \text{var}(X)$, $\Sigma_{yy} = \text{var}(Y)$, $\Sigma_{xy} = \Sigma_{yx} = \text{cov}(X, Y)$

Density $$p(x, y) = \frac{1}{2\pi|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}\begin{bmatrix} x - \mu_x \\ y - \mu_y \end{bmatrix}^T \Sigma^{-1} \begin{bmatrix} x - \mu_x \\ y - \mu_y \end{bmatrix}\right)$$

CoVariance Matrix

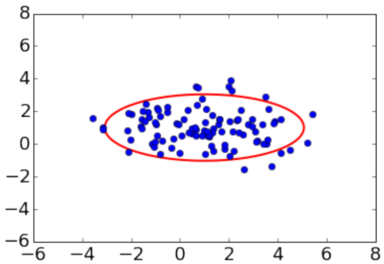
The density is highest at the mean, and falls off in ellipsoidal contours.



Bivariate Gaussian: examples

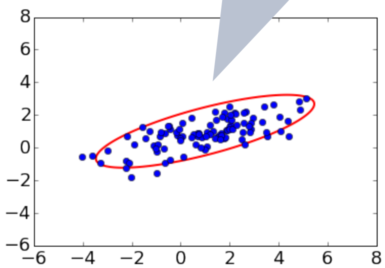
Question?
If I shift this axis, then I could make the correlation 0…conversely, I can make the correlation as "normal" to the mean as possible….

In either case, the mean is $(1, 1)$.



$$\Sigma = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}$$

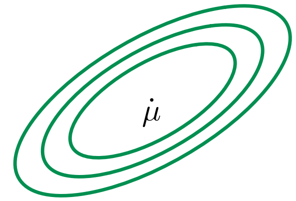
We can see this is uncorrelated
4 is the variance of X
1 is the variance of Y
The covariance is 0



$$\Sigma = \begin{bmatrix} 4 & 1.5 \\ 1.5 & 1 \end{bmatrix}$$

The covariance is .75, which is the std deviation
of x* the standard deviation of y
1.5*1.5

The multivariate Gaussian



$N(\mu, \Sigma)$: Gaussian in $\mathbb{R}^p$

- mean: $\mu \in \mathbb{R}^p$
- covariance: $p \times p$ matrix Σ

Density $$p(x) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$$

Let $X = (X_1, X_2, \ldots, X_p)$ be a random draw from $N(\mu, \Sigma)$.

- μ is the vector of coordinate-wise means:
$$\mu_1 = \mathbb{E}X_1,\ \mu_2 = \mathbb{E}X_2, \ldots,\ \mu_p = \mathbb{E}X_p.$$
- Σ is a matrix containing all pairwise covariances:
$$\Sigma_{ij} = \Sigma_{ji} = \text{cov}(X_i, X_j) \quad \text{if } i \neq j$$
$$\Sigma_{ii} = \text{var}(X_i)$$
- In matrix/vector form: $\mu = \mathbb{E}X$ and $\Sigma = \mathbb{E}(X-\mu)(X-\mu)^T$.

The highest density is at the mean…as you move out from the mean, the density forms an ellipsoid…bigger and bigger as density goes down

Special case: spherical Gaussian

The X_i are independent and all have the same variance σ^2. Thus

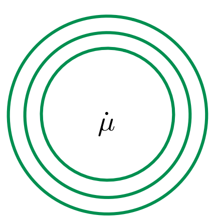
$$\Sigma = \sigma^2 I_p = \text{diag}(\sigma^2, \sigma^2, \ldots, \sigma^2)$$

(off-diagonal elements zero, diagonal elements σ^2).

Each X_i is an independent univariate Gaussian $N(\mu_i, \sigma^2)$:

$$\Pr(x) = \prod_{i=1}^{p}\left(\frac{1}{\sigma\sqrt{2\pi}}e^{-(x_i-\mu_i)^2/2\sigma^2}\right) = \frac{1}{(2\pi)^{p/2}\sigma^p}\exp\left(-\frac{\|x-\mu\|^2}{2\sigma^2}\right)$$

Density at a point depends only on its distance from μ:



Special case: diagonal Gaussian

The X_i are independent, with variances σ_i^2. Thus

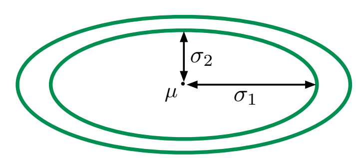
$$\Sigma = \text{diag}(\sigma_1^2, \ldots, \sigma_p^2)$$

(all off-diagonal elements zero).

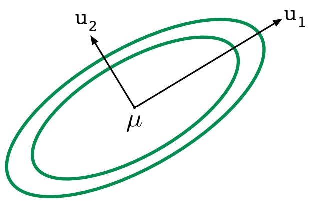
Each X_i is an independent univariate Gaussian $N(\mu_i, \sigma_i^2)$:

$$p(x) = \frac{1}{(2\pi)^{p/2}\sigma_1\cdots\sigma_p}\exp\left(-\sum_{i=1}^{p}\frac{(x_i-\mu_i)^2}{2\sigma_i^2}\right)$$

Contours of equal density are axis-aligned ellipsoids centered at μ:



The general Gaussian $N(\mu, \Sigma)$ in $\mathbb{R}^p$



Eigendecomposition of Σ yields:

- **Eigenvalues**
  $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$
- Corresponding **eigenvectors**
  $u_1, \ldots, u_p$

Recall density: $$p(x) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}\underbrace{(x-\mu)^T\Sigma^{-1}(x-\mu)}_{\text{What is this?}}\right)$$

This is a quadratic function

If we write $S = \Sigma^{-1}$ then S is a $p \times p$ matrix and

$$(x-\mu)^T\Sigma^{-1}(x-\mu) \;=\; \sum_{i,j} S_{ij}(x_i - \mu_i)(x_j - \mu_j),$$

a **quadratic function** of x.

Binary classification with Gaussian generative model

Estimate class probabilities π_1, π_2 and fit a Gaussian to each class:

$$P_1 = N(\mu_1, \Sigma_1), P_2 = N(\mu_2, \Sigma_2)$$

E.g. If data points $x^{(1)}, \ldots, x^{(m)} \in \mathbb{R}^p$ are class 1:

$$\mu_1 \;=\; \frac{1}{m}\left(x^{(1)} + \cdots + x^{(m)}\right) \quad \text{and} \quad \Sigma_1 \;=\; \frac{1}{m}\sum_{i=1}^{m}(x^{(i)} - \mu_1)(x^{(i)} - \mu_1)^T$$

Given a new point x, predict class 1 iff:

$$\pi_1 P_1(x) > \pi_2 P_2(x) \quad \Leftrightarrow \quad x^T M x + 2w^T x \geq \theta,$$

Checking the boundary of a gaussian is a quadratic

where:

This is the gaussian density

$$M = \frac{1}{2}(\Sigma_2^{-1} - \Sigma_1^{-1})$$
$$w = \Sigma_1^{-1}\mu_1 - \Sigma_2^{-1}\mu_2$$

and θ is a constant depending on the various parameters.

$\Sigma_1 = \Sigma_2$: linear decision boundary. Otherwise, quadratic boundary.

Linear decision boundary

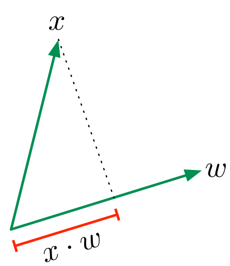
When $\Sigma_1 = \Sigma_2 = \Sigma$: choose class 1 iff

$$x \;\cdot\; \underbrace{\Sigma^{-1}(\mu_1 - \mu_2)}_{w} \;\geq\; \theta.$$

What does $x \cdot w$ (or equivalently $x^T w$, or $w^T x$) mean?

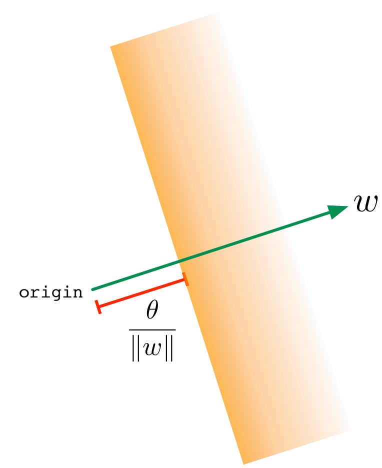
Algebraically: $x \cdot w = w \cdot x = x^T w = w^T x = \sum_{i=1}^{p} x_i w_i$

Geometrically: Suppose w is a unit vector (that is, $\|w\| = 1$). Then $x \cdot w$ is the projection of vector x onto direction w.



Linear decision boundary

Let w be any vector in $\mathbb{R}^p$. What is meant by decision rule $w \cdot x \geq \theta$?

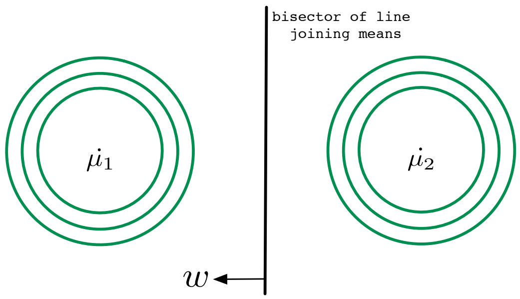


Common covariance: $\Sigma_1 = \Sigma_2 = \Sigma$

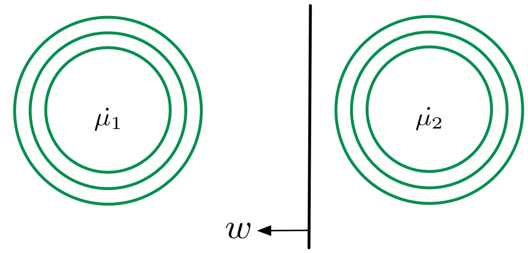
Linear decision boundary: choose class 1 iff

$$x \cdot \underbrace{\Sigma^{-1}(\mu_1 - \mu_2)}_{w} \geq \theta.$$

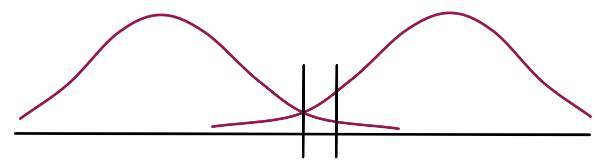
Example 1: Spherical Gaussians with $\Sigma = I_p$ and $\pi_1 = \pi_2$.



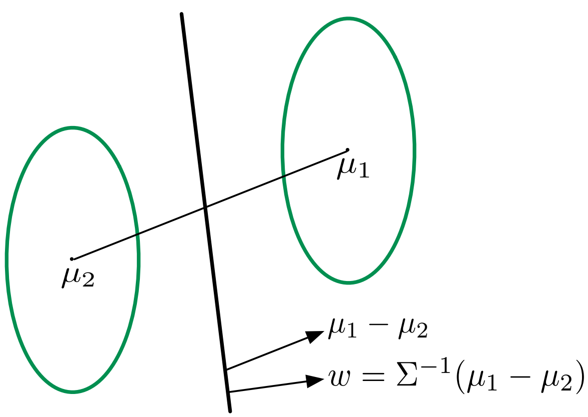
Example 2: Again spherical, but now $\pi_1 > \pi_2$.



One-d projection onto w:



Example 3: Non-spherical.



Rule: $w \cdot x \geq \theta$

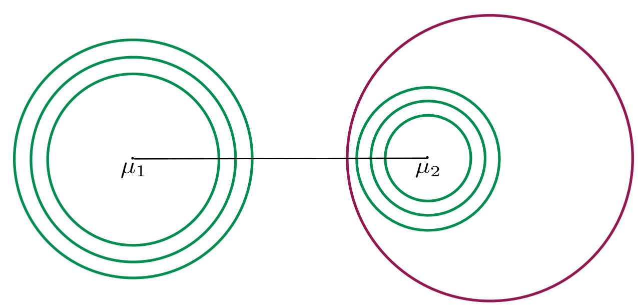
- w, θ dictated by probability model, assuming it is a perfect fit
- Common practice: choose w as above, but fit θ to minimize training/validation error

Different covariances: $\Sigma_1 \neq \Sigma_2$

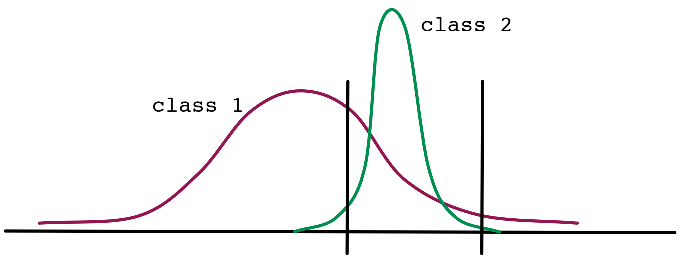
Quadratic boundary: choose class 1 iff $x^T M x + 2w^T x \geq \theta$, where:

$$M = \frac{1}{2}(\Sigma_2^{-1} - \Sigma_1^{-1})$$
$$w = \Sigma_1^{-1}\mu_1 - \Sigma_2^{-1}\mu_2$$

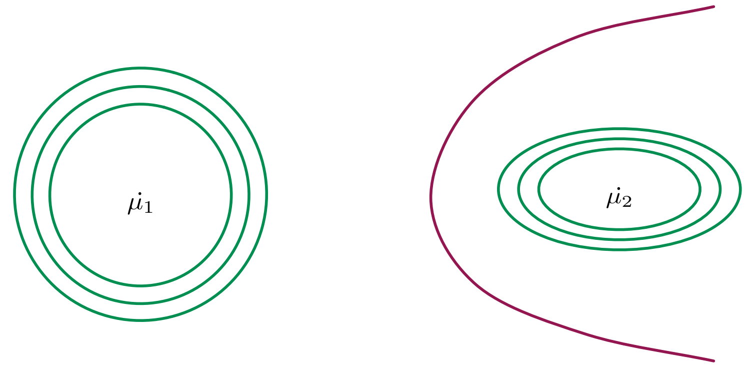
Example 1: $\Sigma_1 = \sigma_1^2 I_p$ and $\Sigma_2 = \sigma_2^2 I_p$ with $\sigma_1 > \sigma_2$



Example 2: Same thing in 1-d. $\mathcal{X} = \mathbb{R}$.



Example 3: A parabolic boundary.



Many other possibilities!

Multiclass discriminant analysis

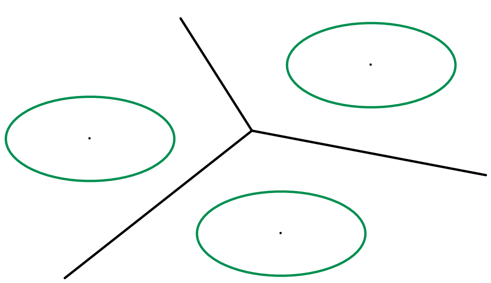
k classes: weights π_j, class-conditional distributions $P_j = N(\mu_j, \Sigma_j)$.

Each class has an associated **quadratic** function

$$f_j(x) = \log\left(\pi_j P_j(x)\right)$$

To class a point x, pick $\arg\max_j f_j(x)$.

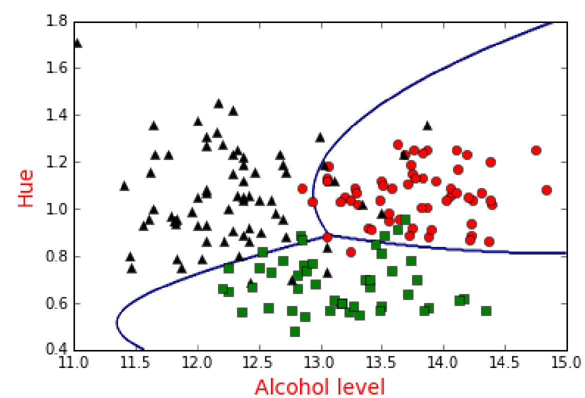
If $\Sigma_1 = \cdots = \Sigma_k$, the boundaries are **linear**.



Example: "wine" data set

Data from three wineries from the same region of Italy

- 13 attributes: hue, color intensity, flavanoids, ash content, ...
- 178 instances in all: split into 118 train, 60 test



Test error using multiclass discriminant analysis: 1/60

Example: MNIST

To each digit, fit:

- class probability π_j
- mean $\mu_j \in \mathbb{R}^{784}$
- covariance matrix $\Sigma_j \in \mathbb{R}^{784 \times 784}$

Problem: formula for normal density uses Σ_j^{-1}, which is singular.

- Need to regularize: $\Sigma_j \to \Sigma_j + \sigma^2 I$
- This is a good idea even without the singularity issue

Error rate with regularization: **???**

Fisher's linear discriminant

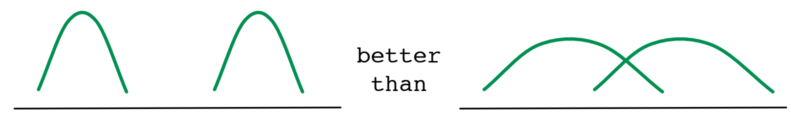
A framework for linear classification without Gaussian assumptions.

Use only first- and second-order statistics of the classes.

Class 1	Class 2
mean μ_1	mean μ_2
cov Σ_1	cov Σ_2
# pts n_1	# pts n_2

A linear classifier projects all data onto a direction w. Choose w so that:

- Projected means are well-separated, i.e. $(w \cdot \mu_1 - w \cdot \mu_2)^2$ is large.
- Projected within-class variance is small.



Fisher LDA (linear discriminant analysis)

Two classes: means μ_1, μ_2; covariances Σ_1, Σ_2; sample sizes n_1, n_2.

Project data onto direction (unit vector) w.

- Projected means: $w \cdot \mu_1$ and $w \cdot \mu_2$
- Projected variances: $w^T \Sigma_1 w$ and $w^T \Sigma_2 w$
- Average projected variance:

$$\frac{n_1(w^T \Sigma_1 w) + n_2(w^T \Sigma_2 w)}{n_1 + n_2} = w^T \Sigma w,$$

where $\Sigma = (n_1 \Sigma_1 + n_2 \Sigma_2)/(n_1 + n_2)$.

Find w to maximize $J(w) = \dfrac{(w \cdot \mu_1 - w \cdot \mu_2)^2}{w^T \Sigma w}$

Solution: $w \propto \Sigma^{-1}(\mu_1 - \mu_2)$. Look familiar?

Fisher LDA: proof

Goal: find w to maximize $J(w) = \dfrac{(w \cdot \mu_1 - w \cdot \mu_2)^2}{w^T \Sigma w}$

1. Assume Σ_1, Σ_2 are full rank; else project.
2. Since Σ_1 and Σ_2 are p.d., so is their weighted average, Σ.
3. Write $u = \Sigma^{1/2} w$. Then

$$\max_w \frac{(w^T(\mu_1 - \mu_2))^2}{w^T \Sigma w} = \max_u \frac{(u^T \Sigma^{-1/2}(\mu_1 - \mu_2))^2}{u^T u} = \max_{u : \|u\| = 1} (u \cdot (\Sigma^{-1/2}(\mu_1 - \mu_2)))^2$$

4. Solution: u is the unit vector in direction $\Sigma^{-1/2}(\mu_1 - \mu_2)$.
5. Therefore: $w = \Sigma^{-1/2} u \propto \Sigma^{-1}(\mu_1 - \mu_2)$.