Machine Learning

by ambedkar@IISc

- Introduction
- What is Data and Model?
- Machine Learning Workflow
- Distance based Classifiers
- Bayes Decision Theory

Note

- These notes is prepared for the Machine Learning course taught at
  IISc.
- These notes should be used in conjunction with the classroom or
  online lectures.
- Where ever some mathematical calculations are involved, I prefer using the blackboard on write on the screen in the online mode. But usually, slides are one-to-one correspondence with what I
  thought during the lectures.
- Time to time, I continuously try to improve these notes and correct the typos. If you see any typos, please mail me at
  ad@iisc.ac.in

About the course

On ML

- Why getting into or mastering ML need not be very easy?
  - If one is starting fresh, there is an ocean out there
  - If one already knows some concepts, one can be confused about what to learn next
- Machine learning can be viewed as list of models or methods
- Or...solutions to some practical problems based on few foundational principles, that involve probabilistic and statistical concepts.

The Best Strategy

- One should constantly....
  - strengthen the foundations
  - try to understand relations between different paradigms and methods
  - most importantly, always experiment...

People Involved

- Instructors
  - Ambedkar Dukkipati
  - Chiranjib Bhattacharyya
- TAs
  - Shubham Gupta
  - Nabanita Paul
  - Tony Gracious
  - Shaarad A R
  - Mariyamma Antony
  - Chaitanya Murti
- Web Support: Aakash Patel

Course Webpage (click below):
https://sml.csa.iisc.ac.in/Courses/Spring21/E0_270/Feb-2021.html

Agenda

About the course

Introduction

Data and Models

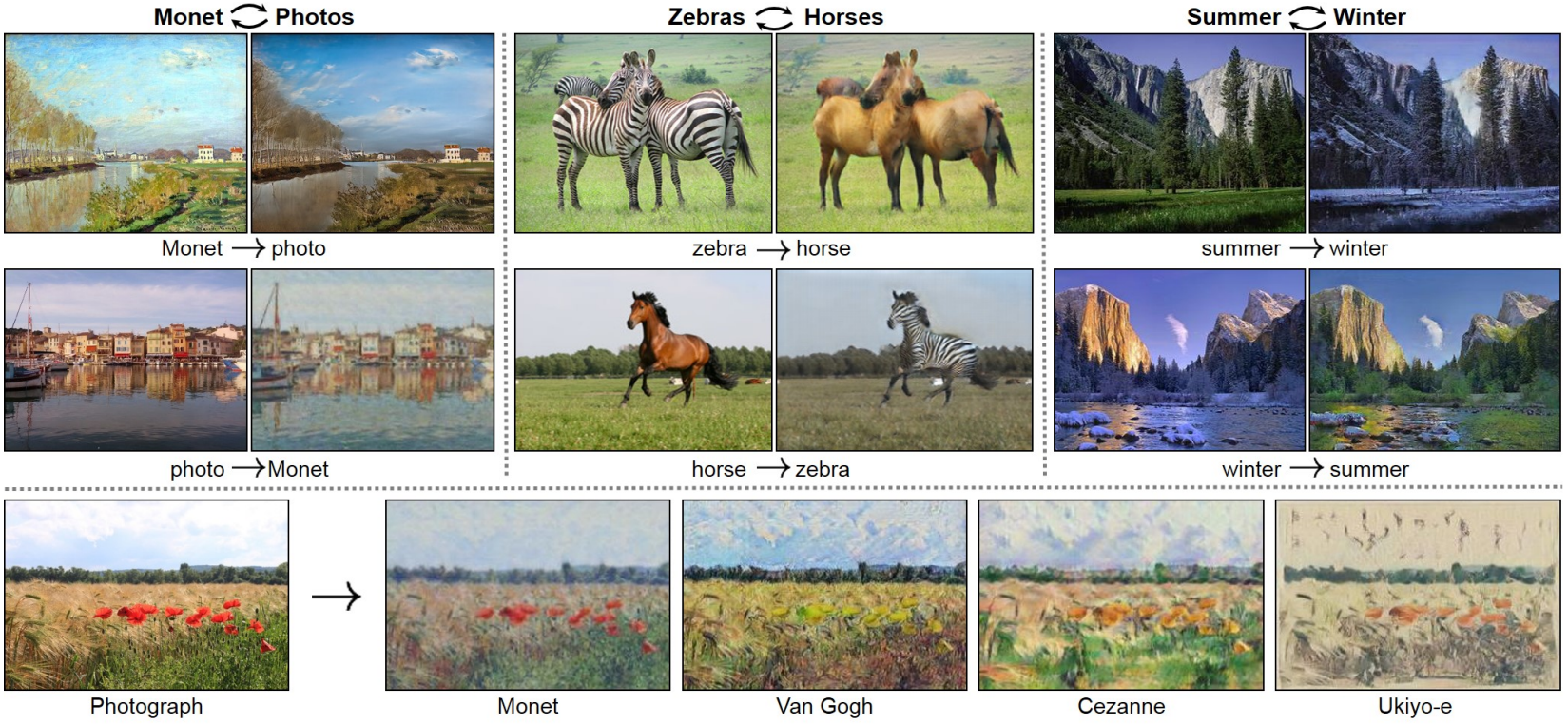
Machine Learning Workflow

Distance Based Classifiers

Bayesian Decision Theory

Introduction

Machine Learning 101 - Building the hype!



CycleGAN: Image to Image Translation[1]

- Using video games to train autonomous driving systems
- More realistic image filtering in smartphone cameras etc.

[1]Image Source: `https://junyanz.github.io/CycleGAN/`

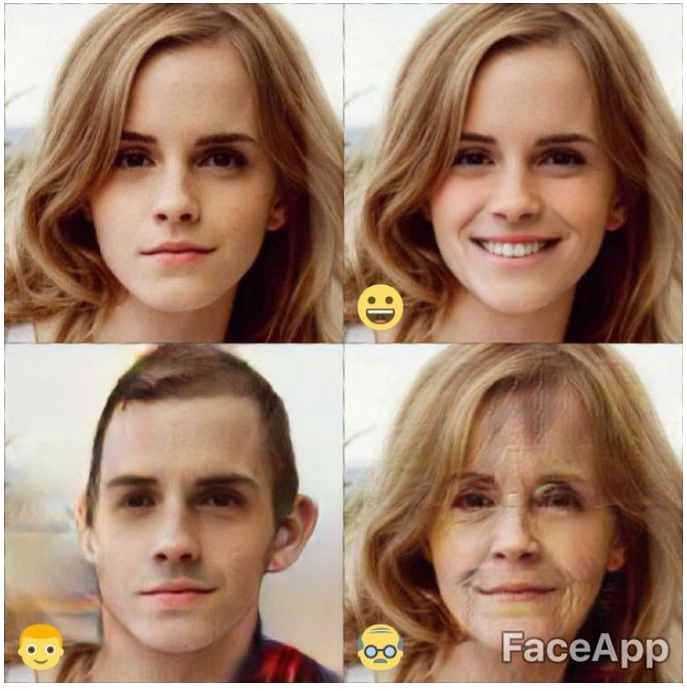
Machine Learning 101 - Building the hype!

Colorizing a Grayscale Image[2]

- Converting all old movies into their colored version
- Restoring old paintings etc.

[2]Image Source: `https://github.com/ImagingLab/Colorizing-with-GANs`

Machine Learning 101 - Building the hype!



Neural FaceApp[3]

- Victim identification during police investigations
- Smartphone filters etc.

[3]Image Source: Google

Machine Learning 101 - Building the hype!

Original Sentence	Flipped Sentiment
the film is strictly routine !	the film is full of imagination.
after watching this movie, I felt that disappointed.	after seeing this film, I'm a fan.
the acting is uniformly bad either.	the performances are uniformly good.
this is just awful.	this is pure genius.

Flipping sentiment of a sentence[4]

- De-radicalizing posts on Facebook
- Removing offensive sentences from movie captions

[4]Source: Toward Controlled Generation of Text

Machine Learning 101 - Building the hype!

Dialogue	**Dialogue**
SYSTEM: Hi. This is an Alexa Prize socialbot.	...
USER: Hi	USER: Let's talk about movies.
SYSTEM: Do you have pets?	SYSTEM: What's your favorite movie?
USER: Yes, I have two rabbits.	USER: star wars movie
Candidate Responses	**Candidate Responses**
BoWFactGenerator: Did you know that male rabbits are called bucks, females are does?	BoWFactGenerator: Here's an interesting fact. The international telephone dialing code for Antarctica is 672.
GRUQuestionGenerator: How about the American family of it?	GRUQuestionGenerator: how about your New York City?
Initiatorbot: Do you believe in love at first sight?	Initiatorbot: How was your day?
LSTMClassifierMSMarco: Mar 23, 2017 Two Parts: Learning What to Feed Your Rabbit Learning How to ...	LSTMClassifierMSMarco: A third Anthology film will be released in 2020.
DualEncoderRedditNews: bye doggie	DualEncoderRedditNews: Now whisper it to me , one more time bby.
Alicebot: Hurrah! Two is a good number of rabbits.	Alicebot: What did you like about the robots in that movie?
Elizabot: OK, but can you elaborate a bit?	Elizabot: How does that make you feel?
VHREDRedditMovies: Hmm. Nope, never heard of it.	**Evi: Star Wars movie a movie in the Star Wars series.**
VHREDSubtitles: Would you like to see them?	VHREDRedditMovies: Oh please. Please. Pleeeease. Let this happen.
BoWWashingtonPost: I do but I call people with dogs crazy also.	VHREDSubtitles: What?
	BoWWashingtonPost: A much more enjoyable feature than last year's old-timer's convention masquerading as a star wars movie.

Chatbots[5]

- In personal assistants like Siri, Google Assistant etc.
- Challenges include sustaining a long range conversation etc.

[5]Image Source: A Deep Reinforcement Learning Chatbot

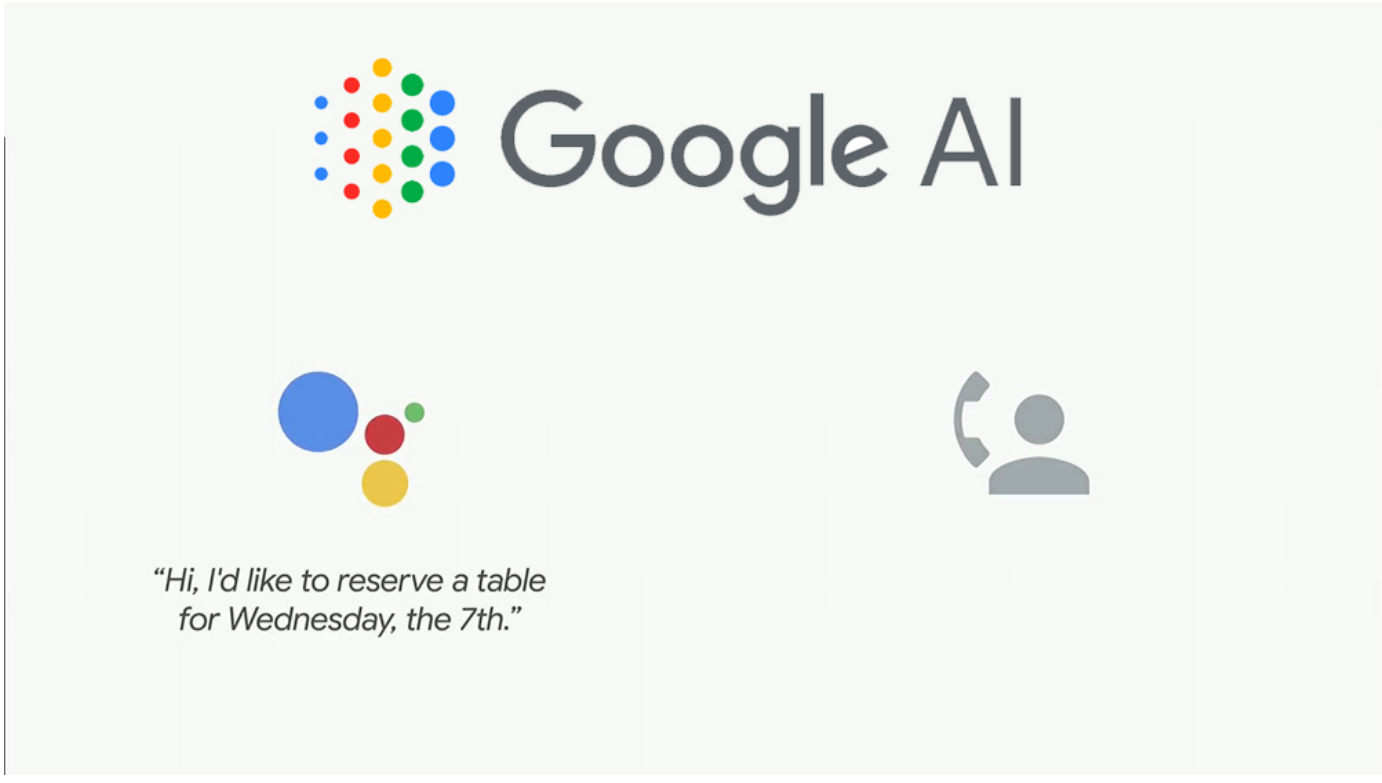
Machine Learning 101 - Building the hype!

Visual Question Answering[6]

- Transcribing videos to generate documentation of a procedure
- Helping blind people in sensing the world around them

[6]Image Source: Making V in VQA Matter

Machine Learning 101 - Building the hype!

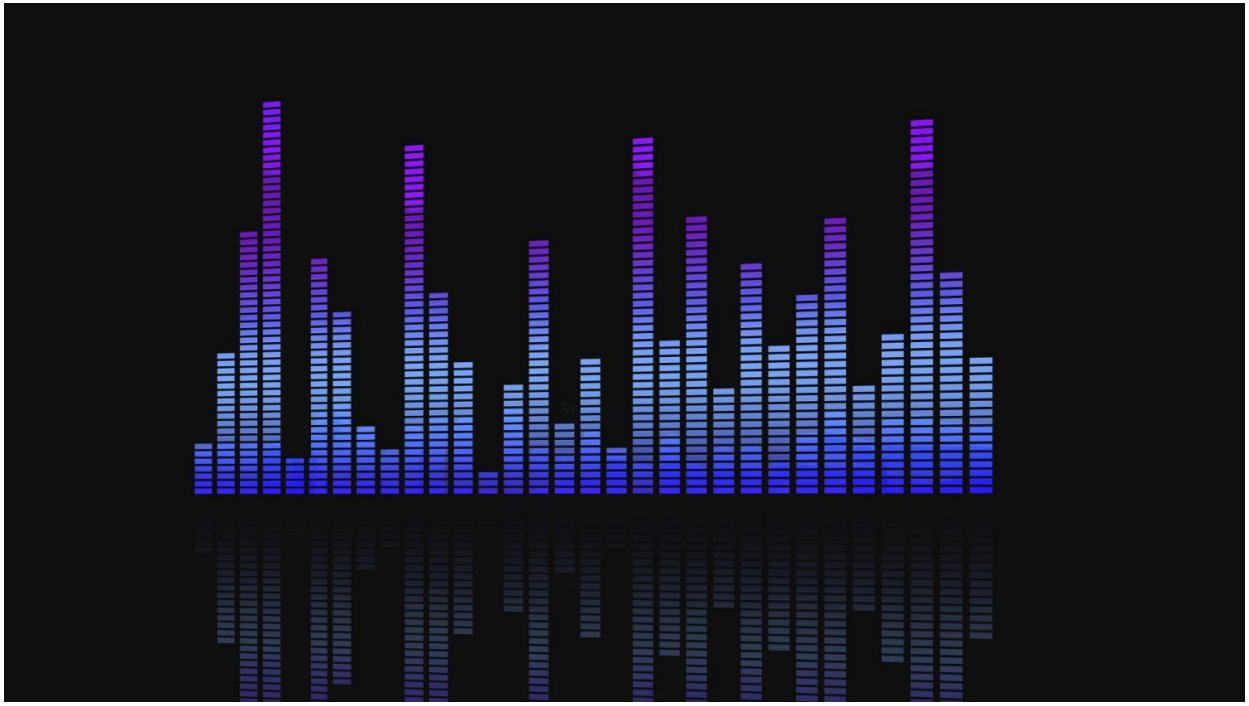


Speech Generation[7]

- Talking in a real world setting
- Personal assistants

[7] Image Source: Google

Machine Learning 101 - Building the hype!



Generating Music[8]

- Conditionally generating music
- Can we replace the monotonous music at customer cares and personalize it to users?

[8]Image Source: Google

Machine Learning 101 - Building the hype!

- Find topics from billions of documents in completely unsupervised way
- Used for improving search results, categorizing documents, finding trends in literature etc.
- The most commonly used algorithm (LDA) is efficient enough to run on a single laptop

Theme	Description	Top words
State bans	State level regulations on abortion	ban, state, govt, bill, ohio
Women's rights	Abortion as women's fundamental right	women's, rights, pills, reproductive, healthcare
Religious views	Church's stance on abortion	jesus, religion, bible, god, faith
Abortion is murder	Perceiving abortion as an act of killing	kill, murder, wrong, life, baby
Planned Parenthood	organization for reproductive health services	planned, parenthood, defund, pp, clinics

Topic Modeling[9]

Machine Learning 101 - Building the hype!

Countless other Applications:

- **Biology and Medicine:**
  - Protein interaction prediction
  - Automated drug discovery
  - Predicting diseases faster than human experts etc.
- **Security:**
  - Applications like face recognition
  - Detecting fraudulent transactions
  - Automated video surveillance etc.
- **Social Sciences**
  - Spreading ideas in a social network
  - Friend recommendations
  - Analyzing large scale surveys etc.

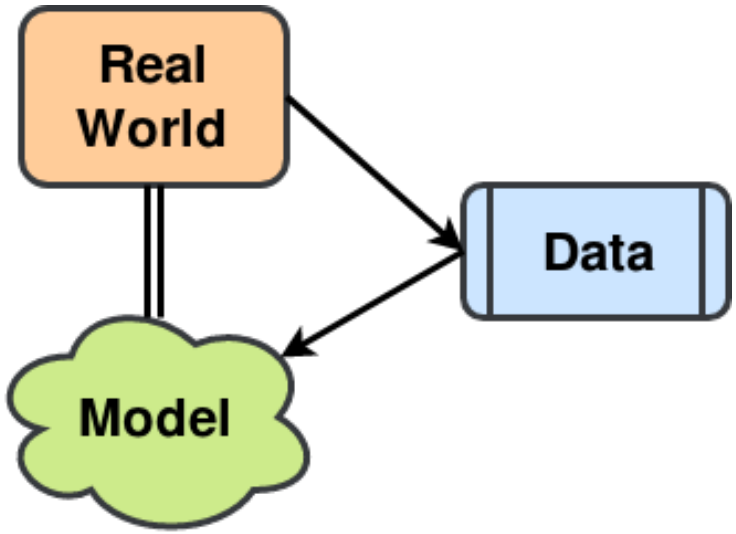
Machine Learning 101 - Building the hype!

- **Information Extraction**
  - Web search
  - Question answering
  - Knowledge graph mining etc.
- **Economics and Finance**
  - Algorithmic Trading
  - Analyzing purchase patterns and market analysis
  - e-commerce applications like product recommendations etc.
- **Others**
  - Automated theorem proving
  - Robotics
  - Advertising
  - And many more...

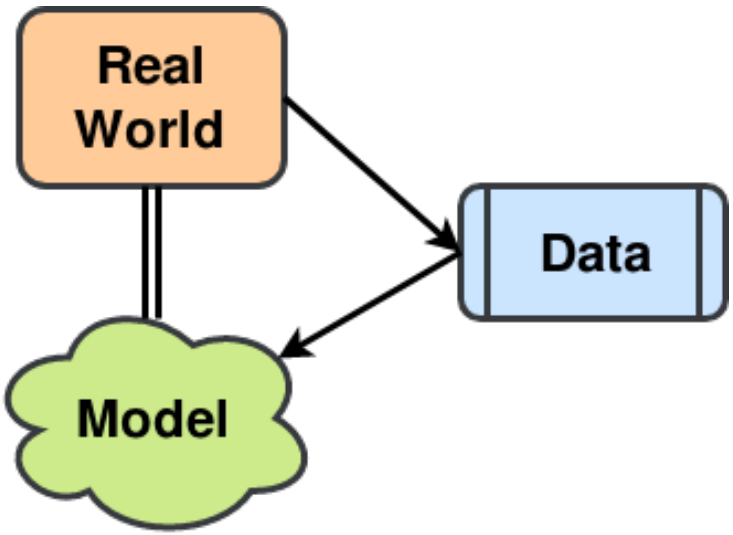
Data and Models

Basics: Data and Models

- Real world offers you data
- A model is a representation of real world
- Data obtained from real world is used for finding parameters of the model
- The model is then used for making predictions or gaining insights about the real world



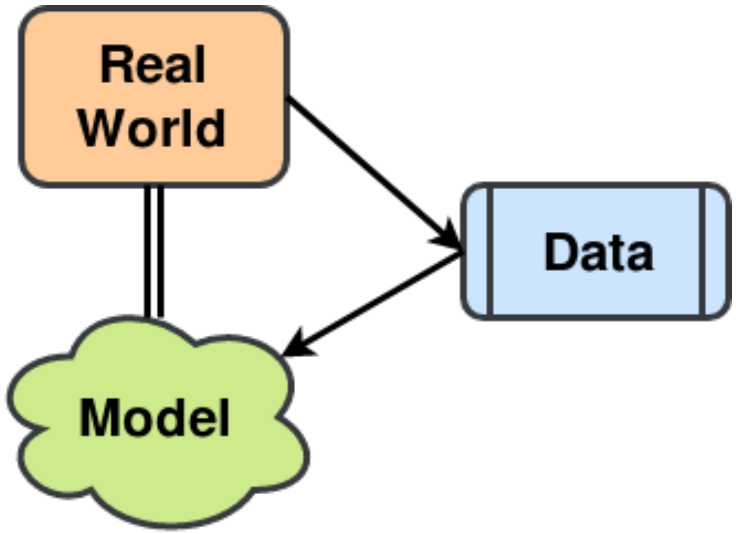
Basics: Data and Models - Example 1



- **Real World**: Sentences used by people in conversations about machine learning
- **Data**: Sentences uttered during this talk
- **Model**: A probability distribution over all possible sentences of length ≤ 50 with Markov assumption

Basics: Data and Models - Example 2

- **Real World**: Students and friendships among them
- **Data**: An observed friendship network involving students from grade one and grade two in a school
- **Model**: Assume people in same grade become friends with probability p and students across grades become friends with probability q



Representation of Data

Most often we represent data in the form of a vector in real space

- Feature vector corresponding to a speech signal
- Feature vector corresponding to a region to predict housing prices
- Feature vector corresponding to pixels of an image
- Feature vector corresponding to a word or a sentence in natural language text (Is this possible?)

Different types of data

- **Spatially Regular Data**
  - Images
- **Sequential Data**
  - Sentences
  - Time series data
- **Relational Data**
  - Tabular data collected during surveys
  - Graph structured data
- **Multimodal Data**
  - videos
  - medical records

Basics: Models

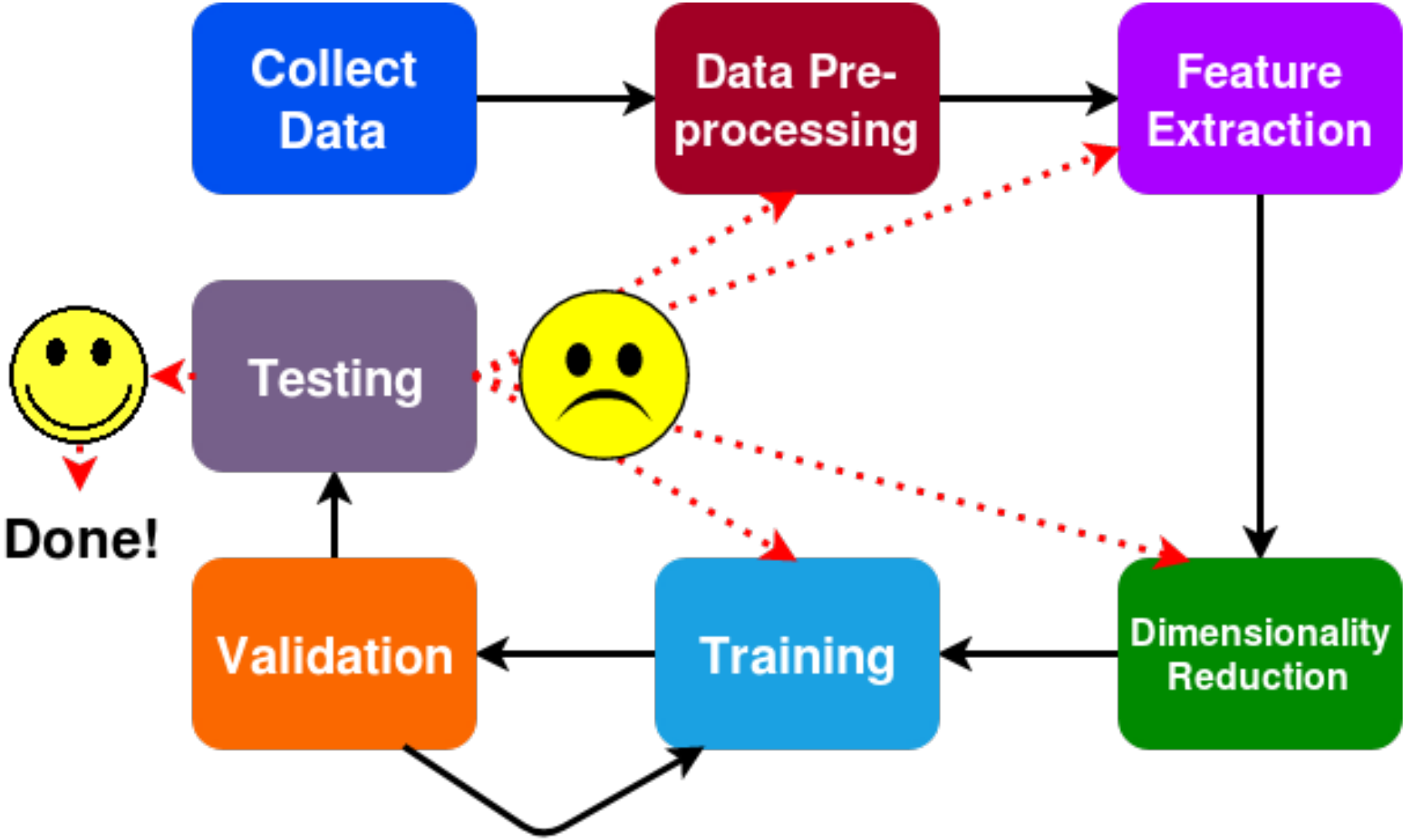
- A model is an abstraction of real world
- Model the aspects of real world that are to be studied
  - *e.g.*, assume an auto-regressive model on words in a sentence
- A very complicated model is usually of no use
  - Should be flexible enough to represent phenomenon of interest
  - Should be tractable
- *e.g.*, assuming that the target variable is a linear function of features in linear Gaussian models for regression

Basics: Models - Examples

- Linear Gaussian model - Regression
- Naïve Bayes model - Classification
- Gaussian mixture model - Clustering
- Hidden Markov model - Discrete valued time series
- Linear dynamical system - Continuous valued time series
- Restricted Boltzmann machines - Data with latent variables
- Stochastic Blockmodels - Networks
- etc.

Machine Learning Workflow

Machine Learning Workflow



Machine Learning Workflow

Machine Learning Workflow - Data Pre-processing

- **Data Cleaning**
  - Removing outliers
  - Filling in missing values
  - Denoising the data
- **Normalization**
  - Making data zero mean
  - Scaling the values
- **Integration**
  - Combine data from different sources

Machine Learning Workflow - Feature Extraction

- **Manually Finding Features**
  - Using domain expertise
  - Finding relevant information
  - *e.g.*, using Mel-Frequency Cepstrum (MFC) to represent sound signals
- **Automatically Discovering Features**
  - Features themselves are learnable
  - These feature are usually not interpretable
  - *e.g.*, Multi Layer Perceptrons (MLP)

Machine Learning Workflow - Dimensionality Reduction

- Finding a compressed representation of data that contains approximately the same information
- Discard features that are not relevant or highly correlated
- Reduces the number of parameters needed in the model
- Leads to better generalization performance
- Use methods like Principle Component Analysis (PCA)

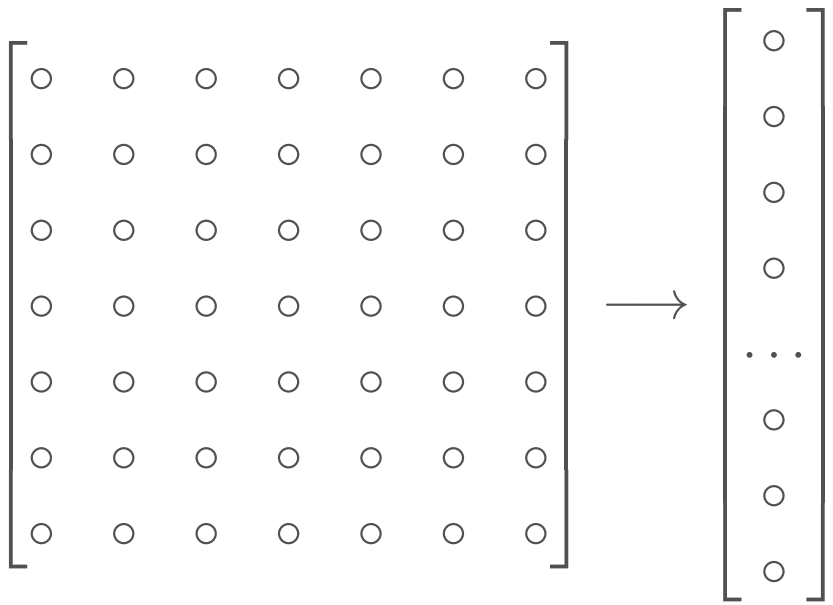
Machine Learning Workflow - Other Components

- **Training**
  - Choose a model
  - Use observed data to learn parameters of the model
  - *e.g.*, learning weights of a neural network
- **Validation**
  - Use validation strategies to fine tune model hyperparameters
  - Perform model selection
  - *e.g.*, using K-fold cross validation to select a value of regularization parameter
- **Testing**
  - Compute the performance on unseen data
  - Diagnose the problems
  - Deploy the model

Distance Based Classifiers

Data Representation

- Learning modules are trained with "data"
  - Supervised: Data comes as a set of input-output pairs $\{(x_n, y_n)\}_{n=1}^{N}$
  - Unsupervised: Data as inputs $\{x_n\}_{n=1}^{N}$
- Each input x_n is usually a D dimensional feature vector
  - Say x_n is usually a 7×7 image. It can be represented using a vector of size 49 of pixel intensities



Data Representation(contd...)

- Note: In certain applications input x_n need not be a fixed length of vector. For example protein sequences, etc.
- Output y_n can be
  - real values (eg. regression)
  - categorical (eg. classification)
  - structured object (eg. structured output learning)
- The learning task becomes tougher and tougher when the dimensionality of data is very high.

Data: In what form?

- Data is always "raw".
- Most machine learning models works only when the "nice" and "appropriate" and "useful" features are fed to them.
- So feature can be learned or extracted
  - Learned: The model/algorithms automatically learn the useful features
  - Extracted: Hand-crafted features defined by a domain expert.

Data in vector space representation

- Each feature vector $x_n \in \mathbb{R}^{D\times 1}$ is a point in the D dimensional vector space $\mathbb{R}^D$
- By putting data in a vector space we can incorporate all tools that is provided by Linear Algebra in our problem solving
- More importantly matrix computations play an important role in machine learning

Data in vector space representation(contd...)

- Vector space provides us with distance and similarity measures

- Euclidean distance between $x_n, x_m \in \mathbb{R}^D$

$$d(x_n, x_m) = \|x_n - x_m\|_2 = \sqrt{(x_n - x_m)^T (x_n - x_m)}$$

$$= \sqrt{\sum_{d=1}^{D} (x_{n_d} - x_{m_d})^2}$$

Data in vector space representation(contd...)

- Vector space provides us with distance and similarity measures
- Inner product similarity between $x_n, x_m \in \mathbb{R}^D$ (cosine similarity)

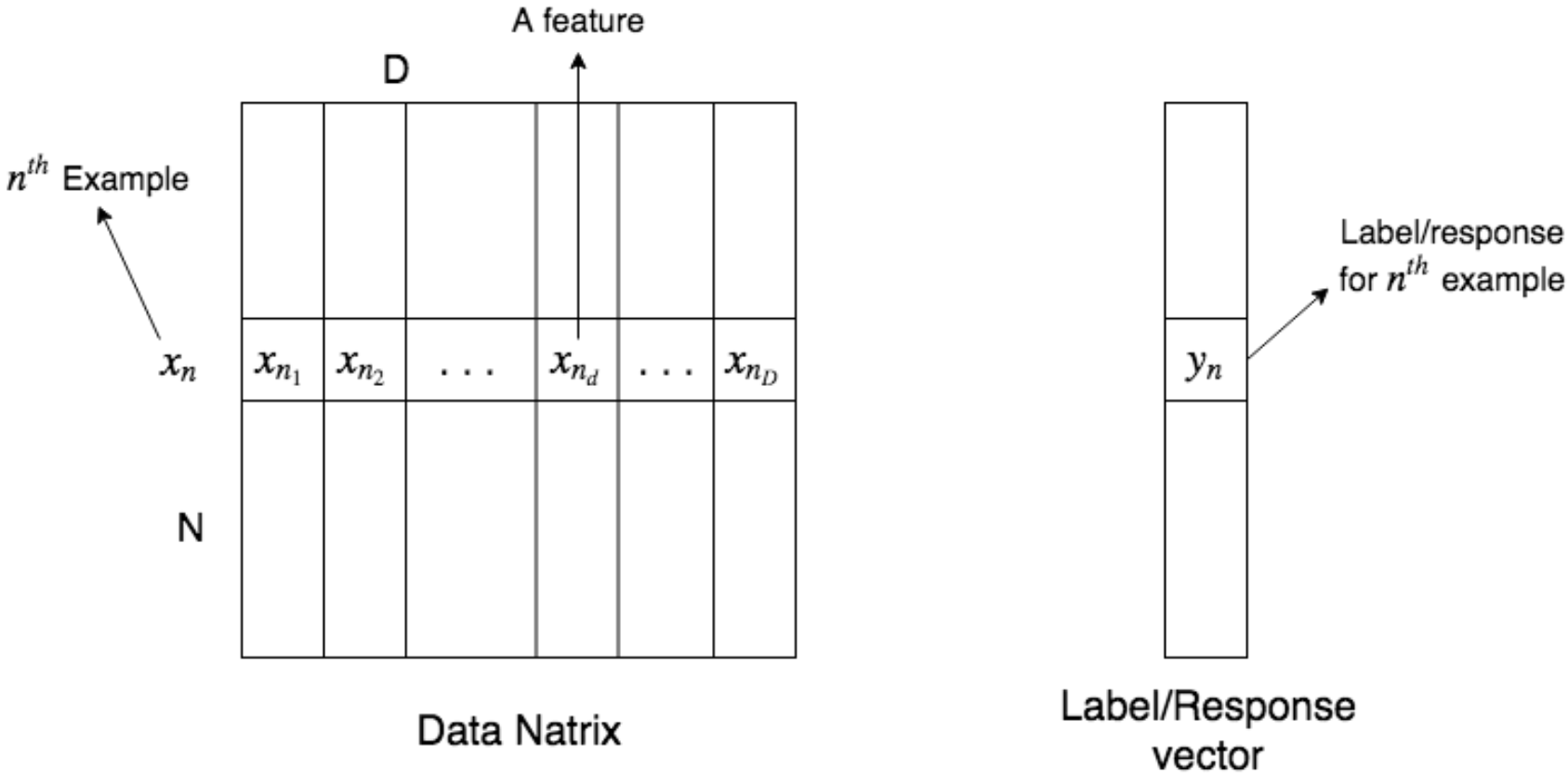
$$\langle x_n, x_m \rangle = x_n^T x_m = \sum_{d=1}^{D} x_{n_d} x_{m_d}$$

- ℓ_1 distance between $x_n, x_m \in \mathbb{R}^D$

$$\ell_1(x_n, x_m) = \|x_n - x_m\|_1 = \sum_{d=1}^{D} ||x_{n_d} - x_{m_d}||$$

Data Matrix

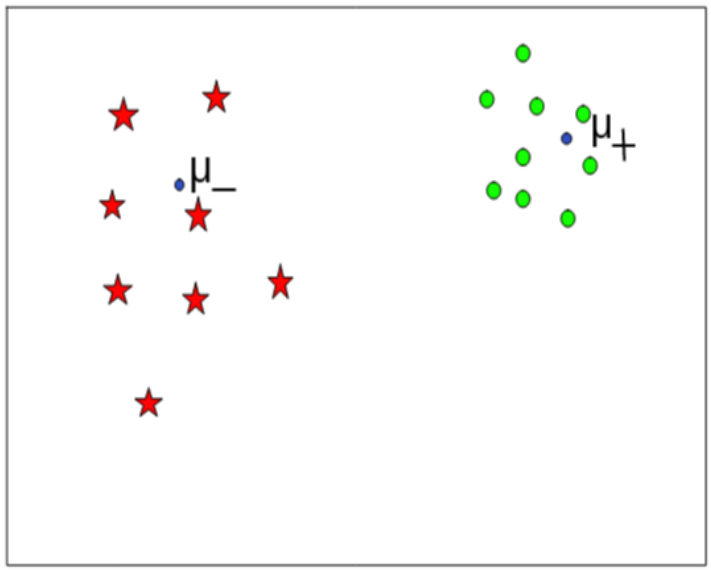
- $x = \{x_1, \ldots, x_n\}$ denotes data in form of $N \times D$ feature matrix



- $y = \{y_1, \ldots, y_N\}$ denotes labels/responses in the form of an $N \times 1$ label/response vector.

Setting

- Given N labelled training examples $\{x_n, y_n\}_{n=1}^{N}$ from two classes (+ve and -ve)
  - Assume positive is green and negative is Red.
  - Assume we have N_+ examples from +ve class and N_- examples from negative class.
- **Aim:** Learn a model to predict label y for a new test sample.



A Simple Decision Rule based on Means

Rule: Assign test sample to classes with closer mean.

- The mean of each class is given by

$$\mu_- = \frac{1}{N_-} \sum_{y_n=-1} x_n$$

$$\mu_+ = \frac{1}{N_+} \sum_{y_n=+1} x_n$$

- Note:- Can we just store the two means and throw away data.

A Simple Decision Rule based on Means (contd. . .)

- Distances from each mean are given by

$$\|\mu_- - x\|^2 = \|\mu_-\|^2 + \|x\|^2 - 2\langle \mu_-, x\rangle$$

$$\|\mu_+ - x\|^2 = \|\mu_+\|^2 + \|x\|^2 - 2\langle \mu_+, x\rangle$$

- Here
  - $\|a - b\|^2$ denotes squared Euclidean distance between a and b.
  - $\langle a, b\rangle$ denotes inner product of two vector a and b.
  - $\|a\|^2 = \langle a, a\rangle$ denotes squared l_2 norm of a.

The Decision Rule

- Denote the decision rule by $f : \chi \longrightarrow \{+1, -1\}$

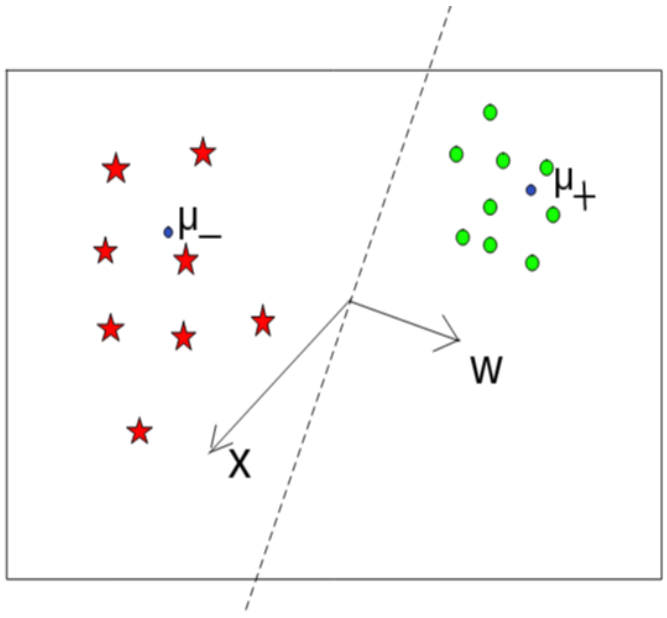
$$\begin{aligned} f(x) &= \|\mu_- - x\|^2 - \|\mu_+ - x\|^2 \\ &= 2\langle \mu_+ - \mu_-, x \rangle + \|\mu_-\|^2 - \|\mu_+\|^2 \end{aligned}$$

- Decision Rule: if $f(x) > 0$ then x in $+1$
  otherwise x in -1

  i.e. y = sign$[f(x)]$

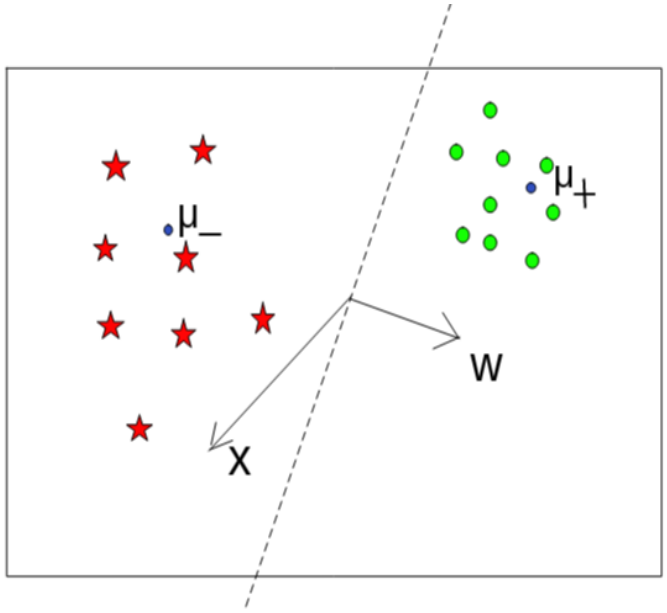
The Decision Rule

- Note: $f(x)$ denotes a hyperplane based classification rule, where $w = \mu_+ - \mu_-$ represents the direction rule to the hyperplane.



- This specific form of decision rule appears in many supervised algorithms.
- Inner product can be replaced by more general similarity measures.

Decision Rule Based on Means: Some Comments



- It can be implemented easily.
- Would require plenty of traning data for each class.
  - Because to estimate mean reliably
  - Note: if we have class imabalanced data, this will not work.

Decision Rule Based on Means: Some Comments (contd ..)

- It can only learn linear decision boundaries.
  - We need to replace Euclidean distance by nonlinear distance function. Kernels?
- Data: We assume that there is an underlying probability distribution.
  - Mean can be thought of as one characteristic of a distribution.
  - How about modelling each class by a class conditional probability distribution.
  - Then compute distances from these distributions.
  - Linear Discriminant Analysis

Bayesian Decision Theory

Bayesian Decision Making in Real Life

Let us help a fisherman trying to classifying his catch. For simplicity, let us consider that he has to classify between Sea bass (ω_1) and Salmon (ω_2).

- It is a two class calssification problem
- We will study this in various scinarios

Decision Rule: Based on Prior Knowledge

Fishermen will have some domain or prior knowledge. Suppose, except for this we do not have any other knowledge.

- Suppose, in a particular season there is a more probability of catching sea bass or in a particular area probability of getting Salmon is more.
- Suppose the prior probabilities are $P(\omega_1)$ and $P(\omega_2)$. $(P(\omega_1) + P(\omega_2) = 1 \ \& \ P(\omega_1), P(\omega_2) \geq 0)$
- Rule (or common sense) says

$$\text{Decide } \omega_1 \text{ if } P(\omega_1) > P(\omega_2)$$

$$\omega_2 \text{ otherwise}$$

Decision Rule: Based on Prior Knowledge (contd...)

How good is this?

- It looks fine but for every catch the class label is going to be the same.

- Can we feed the image of of the fish to our model so that it can consider its features before deciding on the label?

Decision Rule: Based on class conditional probabilities

Aim here is to get features of the fish and feed it to our model.

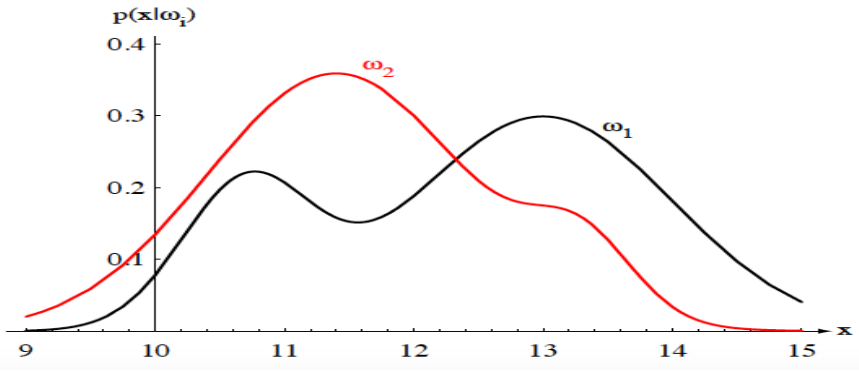
- Suppose we can get features of the fish like measurement of weight (x).
- We will consider the class conditional densities $P(x|\omega_i),\ i = 1, 2\ )$, which are also called likelihood.
- $P(x|\omega_i)$ denotes probability of observing a particular feature(s) x provided it has a class lablel ω_i.

Decision Rule: Based on class conditional probabilities (Contd...)

Now the decision Rule:

Decide ω_1 if $P(x|\omega_1) > P(x|\omega_2)$

ω_2 otherwise



Likelihoods

Bayesian way....

Bayesian formulation helps in combining prior knowledge and class conditional probabilities into a single rule by finding posterior distribution $P(\omega_i|x)$

Decision Rule: Using posterior distribution

Using bayes rule

$$P(\omega_i|x) = \frac{P(x|\omega_i)P(\omega_i)}{P(x)} \; i = 1, 2$$

where

$$P(x) = \sum_{i=1,2} P(x|\omega_i)P(\omega_i)$$

$P(x)$ is called evidence

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}$$

Rules says

$\omega_1 \; if \; P(\omega_1|x) > P(\omega_2|x)$

$\omega_2 \; otherwise$

Note:
Prior and likelihood are the main factors determining the posterior probability the evidence can be considered as scaling.

Error Analysis

The probability of error is

$$\begin{aligned} P(error|x) &= P(\omega_1|x) \; if \; we \; decide \; \omega_2 \\ &= P(\omega_2|x) \; if \; we \; decide \; \omega_1 \end{aligned}$$

The overall probability of error is

$$P(error) = \int_{-\infty}^{+\infty} P(error, x)dx \quad = \int_{-\infty}^{+\infty} P(error|x)P(x)dx$$

The bayes decision rule says

$$\begin{aligned} &\omega_1 \; if \; P(\omega_1|x) \; > \; P(\omega_2|x) \\ &\omega_2 \; otherwise \end{aligned}$$

So, it minimizes $P(error|x)$. Hence $P(error)$ is also minimized

Bayesian Decision Theory: A General Setting

$\{\omega_1, \omega_2, \ldots, \omega_c\}$	:	a finite set of classes
$\{\alpha_1, \alpha_2, \ldots, \alpha_a\}$	:	a finite set of actions
$\lambda(\alpha_i \mid \omega_j)$, $i = 1, 2, \ldots, a$ and $j = 1, 2, \ldots, c$	:	denotes a loss function that describes loss for taking action α_i when the of the x value is ω_i
$x \in \mathbb{R}^D$	:	is a feature vector which is an instance of random vectors
$P(x \mid \omega_j)$, $j = 1, 2, \ldots, c$	:	class conditional probability density function or likelihood
$P(\omega_j)$, $j = 1, 2, \ldots, c$	:	prior probabilities

- Posterior probabilities $P(\omega_i|x)$ $j = 1, 2, \ldots, c$ can be calculated using the bayes formula $P(\omega_i|x) = \frac{P(x|\omega_i)P(\omega_i)}{P(x)}$
- where the evidence $P(x) = \sum_j^c P(x|\omega_j)P(\omega_j)$

Bayesian Decision Rule as Risk Minimization

Suppose given $x \in \mathbb{R}^D$, we take action α_i, then the expected loss associated with taking action α_i is

$$R(\alpha_i|x) = \sum_{j=1}^{c} \lambda(\alpha_i|\omega_j)P(\omega_j|x)$$

This is called the conditional risk. In continuous form overall risk is

$$R = \int_{x\in\mathbb{R}^D} R(\alpha(x)|x)P(x)\mathrm{d}x$$

Bayesian Decision Rule as Risk Minimization

Aim: Find the decision rule that minimizes the overall risk R.

- The minimum risk is called the Bayes risk
- Suppose $\alpha^*(x) = \underset{\alpha(x)=\{\alpha_1,\alpha_2,\ldots,\alpha_a\}}{\arg\min} R(\alpha_i|x)$
- Then

$$R^* = \int_{x\in\mathbb{R}^D} R(\alpha^*(x)|x)P(x)\mathrm{d}x$$

  is the minimum risk.

Two Class Classification and Likelihood Ratio

- Let action α_i denotes deciding that true class label is ω_1, α_2 denotes deciding that true class is ω_2
- Let $\lambda_{ij} = \lambda(\alpha_i|\omega_j)$ for $i = 1, 2$ and $j = 1, 2$, denotes the loss incurred when thte decision is α_i but true class is ω_j
- The conditional risk for any observation $x \in \mathbb{R}^d$ is

$$R(\alpha_1|x) = \lambda_{11}P(\omega_1|x) + \lambda_{12}P(\omega_2|x)$$
$$R(\alpha_2|x) = \lambda_{21}P(\omega_1|x) + \lambda_{22}P(\omega_2|x)$$

- Decision rule is

$$\omega_1 \text{ if } R(\alpha_1|x) \; < \; R(\alpha_2|x)$$
$$\omega_2 \text{ otherwise}$$

- Here we are taking decision based on the risk not by minimum posterior probabilities.

Two Class Classification and Likelihood Ratio (contd...)

$$\begin{aligned} R(\alpha_1|x) &< R(\alpha_2|x) \\ \lambda_{11}P(\omega_1|x) + \lambda_{12}P(\omega_2|x) &< \lambda_{21}P(\omega_1|x) + \lambda_{22}P(\omega_2|x) \end{aligned}$$

- We have $\lambda_{21} = \lambda(\alpha_2|\omega_1)$ loss occurred for being wrong
- We have $\lambda_{11} = \lambda(\alpha_1|\omega_1)$ loss occurred for being right
- Similarly λ_{12} and λ_{22}
- It is sensible to assume $\lambda_{21} > \lambda_{11}$ and $\lambda_{12} > \lambda_{22}$ as risk in being wrong is greater than for being right.
- So, $\lambda_{21} - \lambda_{11} > 0$ and $\lambda_{12} - \lambda_{22} > 0$
- Now by minimum risk strategy we decide ω_1 if $(\lambda_{21} - \lambda_{11})P(\omega_1|x) > (\lambda_{12} - \lambda_{22})P(\omega_2|x)$ else ω_2.

Two Class Classification and Likelihood Ratio (contd...)

Now using bayes theorem we write the previous strategy in terms of prior and likelihood as given below.

$$(\lambda_{21} - \lambda_{11})P(\omega_1)P(x|\omega_1) > (\lambda_{12} - \lambda_{22})P(\omega_2)P(x|\omega_2)$$

$$\implies \frac{P(x|\omega_1)}{P(x|\omega_2)} > \frac{\lambda_{12} - \lambda_{22}}{\lambda_{21} - \lambda_{11}} \frac{P(\omega_2)}{P(\omega_1)}$$

$$\implies \text{likelihood ratio} > \text{quantity independent of x}$$

$$\implies \psi(x) > c, \text{ where } \psi(x) = \frac{P(x|\omega_1)}{P(x|\omega_2)}$$

Two Class Classification and Likelihood Ratio: Summary

- Bayes rule can be interpreted as deciding ω_1 if the likelihood ratio exceeds a threshold value that is independent of x.

- Assumption is that we know the class conditional densities.

- In practical setting we learn likelihood from the training dataset. That is the threshold c act as prior and $\psi(x)$ act as classifier whose parameters are to be learned from the data.

Classification with 0-1 loss

- $\{\omega_1, \omega_2, ..., \omega_c\}$ a finite set of classes
- $\{\alpha_1, \alpha_2, ..., \alpha_c\}$ a finite set of actions corresponding to $\{\omega_1, \omega_2, ..., \omega_c\}$
- 0-1 loss is define as

$$\begin{aligned} \lambda(\alpha_i|\omega_j) &= 0 \ if \ i = j \\ &= 1 \ if \ i \neq j \\ i, j &= 1, 2, ..c \end{aligned}$$

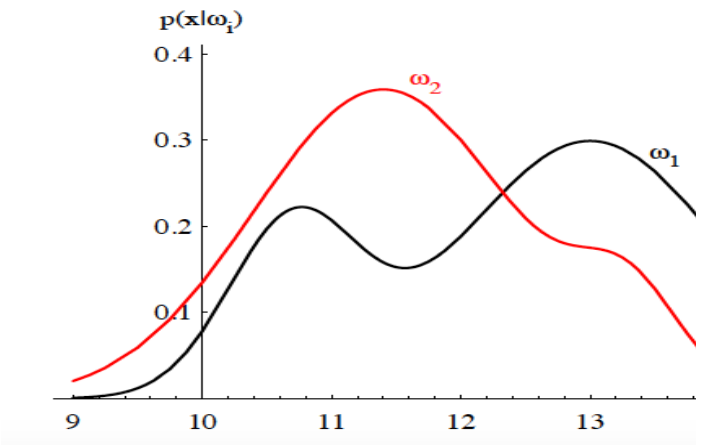
This assigns no loss to a correct decision and assigns unit loss to wrong decision. Now conditional risk

$$R(\alpha_i|x) = \sum_{j}^{c} \lambda(\alpha) i|\omega_j) P(\omega_j|x) = \sum_{j \neq i} P(\omega_j|x) = 1 - P(\omega_i|x)$$

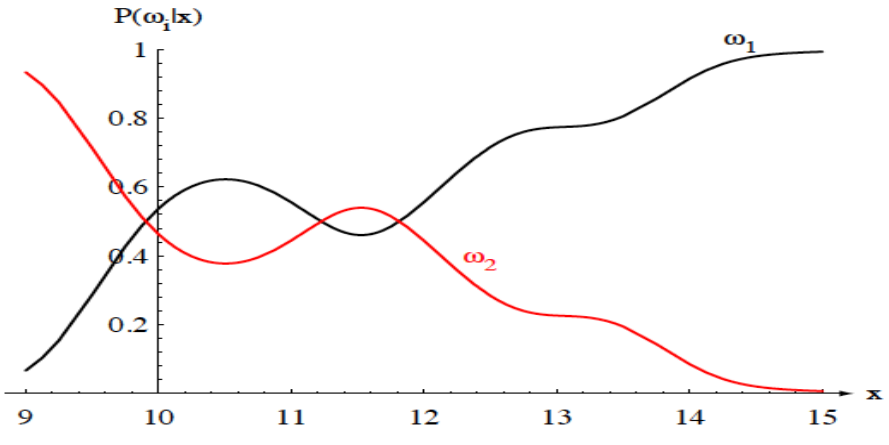
$\implies$ If we decide on ω_i if $P(\omega_j|x)$ is maximum

$\implies R(\alpha_i|x)$ is minimum $\implies$ $R(x)$ is minimum

Bayes rule in action

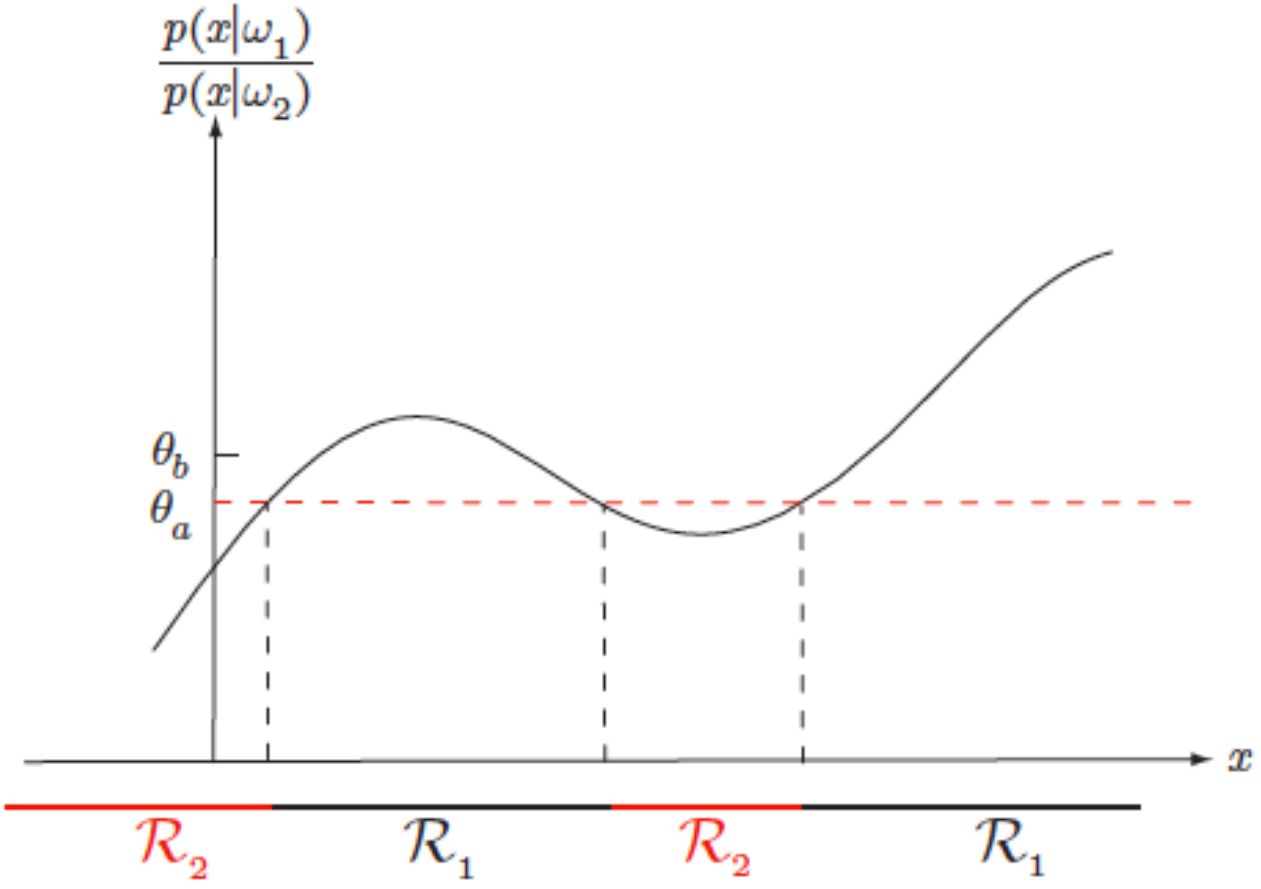


(A) *Likelihood*



(B) *Posterior*

Bayes rule in action



Likelihood Ratio and threshold for decision boundary

Minimax Criterion (Two class case)

Aim: To design the classifier such a way that it performs well over a range of prior probabilities.

Example: We would like to design our classifier such a way that it can be used in a difficult place where we do not know the prior probabilities

Implies: Design the classifier so that the worst over all risk for any value of prior is as small as possible. That is

Minimize the Maximum Possible Risk

Minimax Criterion (cont...)

Let $\mathcal{R}_1$ be the region where we decide ω_1.
Let $\mathcal{R}_2$ be the region where we decide ω_2
We have

$$
\begin{aligned}
R &= \int_{\mathbb{R}^D} R(\alpha(x)|x)P(x)\mathrm{d}x \\
&= \int_{\mathcal{R}_1} R(\alpha(x)|x)P(x)\mathrm{d}x \; + \; \int_{\mathcal{R}_2} R(\alpha(x)|x)P(x)\mathrm{d}x \\
&= \int_{\mathcal{R}_1} [\lambda_{11}P(\omega_1|x) + \lambda_{12}P(\omega_2|x)] + \int_{\mathcal{R}_2} [\lambda_{21}P(\omega_1|x) + \lambda_{22}P(\omega_2|x)]
\end{aligned}
$$

Minimax Criterion (contd. . .)

Using Bayes theorem

$$R = \int_{\mathcal{R}_1} [\lambda_{11} P(\omega_1) P(x|\omega_1) + \lambda_{12} P(\omega_2) P(x|\omega_2)] \mathrm{d}x + \int_{\mathcal{R}_2} [\lambda_{21} P(\omega_1) P(x|\omega_1) + \lambda_{22} P(\omega_2) P(x|\omega_2)] \mathrm{d}x$$

We have

- $P(\omega_2) = 1 - P(\omega_1)$ and
- $\int_{\mathcal{R}_1} P(x|\omega_1)\mathrm{d}x = 1 - \int_{\mathcal{R}_2} P(x|\omega_1)\mathrm{d}x$

Now we can write it R as

$$R(P(\omega_1)) = \lambda_{22} + (\lambda_{12} - \lambda_{22}) \int_{\mathcal{R}_1} P(x|\omega_1)\mathrm{d}x + P(\omega_1) \Big[(\lambda_{11} - \lambda_{22}) + (\lambda_{21} - \lambda_{11}) \int_{\mathcal{R}_2} P(x|\omega_1)\mathrm{d}x - (\lambda_{12} - \lambda_{22}) \int_{\mathcal{R}_1} P(x|\omega_2)\mathrm{d}x \Big]$$

Minimax Classification

- Once decision boundary is set (i.e $\mathcal{R}_1$ and $\mathcal{R}_2$) the overall risk is linear in $P(\omega_1)$
- If we can find boundary such that constant of proportionality is zero then corresponding decision boundary gives the minimax solution.
- $$(\lambda_{11}-\lambda_{22})+(\lambda_{21}-\lambda_{11})\int_{\mathcal{R}_2} P(x|\omega_1)\mathrm{d}x-(\lambda_{12}-\lambda_{22})\int_{\mathcal{R}_1} P(x|\omega_2)\mathrm{d}x$$

  is zero for minimax solution
- $\lambda_{22}+(\lambda_{12}-\lambda_{22})\int_{R_1} P(x|\omega_1)\mathrm{d}x$ is the minimax risk.
- This minimax formulation has applications in game theory. Think that an adversary is providing with a wrong prior.

Discriminant Functions

Now we try to describe classifiers as discriminant functions. Let $\omega_1, \omega_2, \ldots, \omega_c$ are class labels and features are D-dimentional vectors.

AIM: Now the aim is to learn a function

$$g : \mathbb{R}^D \rightarrow \{\omega_1, \ldots, \omega_c\}$$

We realize the function g by using $g_1, ..., g_c : \mathbb{R}^D \rightarrow \mathbb{R}$ as follows. and with decision rule

$$g(x) = \omega_i \quad \text{if}$$
$$g_i(x) > g_j(x) \; \forall \; j = 1, 2, \ldots, c \; \& \; j \neq i$$

Bayes Classifier as Discriminant Functions

The bayes classifier can be represented using discriminant functions. In that case

$$g_i(x) = -R(\alpha_i|x) \; i = 1, 2, .., c$$

$\because$ Maximum discriminant function corresponds to minimum conditional risk. For $0-1$ loss function (minimum error rate) we have

$$g_i(x) = P(\omega_i|x) \; i = 1, 2, ..., c$$

$\therefore$ Maximum discriminant function corresponds to maximum posterior probability

Now,

$$g_i(x) = P(\omega_i|x) = \frac{P(x|\omega_i)P(\omega_i)}{\sum_{i=1}^{c} P(x|\omega_i)P(\omega_i)}$$

Bayes Classifier as Discriminant Functions (contd..)

Now,

$$g_i^{(1)}(x) = P(\omega_i|x) = \frac{P(x|\omega_i)P(\omega_i)}{\sum_{i=1}^{c} P(x|\omega_i)P(\omega_i)}$$

$$g_i^{(2)}(x) = P(x|\omega_i)P(\omega_i)$$

$$g_i^{(3)}(x) = \ln P(x|\omega_i) + \ln P(\omega_i)$$

All the discriminant functions $g^{(1)}, g^{(2)}, \& \ g^{(3)}$ are equivalent.

For two category case, suppose g_1 and g_2 are discriminant functions corresponding to two classes.
Let $g(x) = g_1(x) - g_2(x)$ and use the following decision rule

$$Decide\ \omega_1\ if\ g(x) > 0$$

$$\omega_2\ otherwise$$

Bayes Classifier as Discriminant Functions two class case

For two category case,
suppose g_1 and g_2 are discriminant functions corresponding to two classes.
Let $g(x) = g_1(x) - g_2(x)$ and use the following decision rule

$$Decide\ \omega_1\ if\ g(x) > 0$$
$$\omega_2\ otherwise$$

Now,

$$\begin{aligned} g(x) &= P(\omega_1|x) - P(\omega_2|x) \\ &= \ln \frac{P(x|\omega_1)}{P(x|\omega_2)} + \ln \frac{P(\omega_1)}{P(\omega_2)} \end{aligned}$$

Discriminant Functions for Normal densities

- Normal distribution

$$P(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]$$

- Multivariate normal distribution

$$P(x) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right]$$

Discriminant Functions for Normal densities

For a c class classfification problem we have discriminant functions $g_1, \ldots, g_c$ which are functions from $\mathbb{R}^D$ to $\mathbb{R}$.

In the case of Bayesian classifier we have, for $i = 1, \ldots, c$

$g_i(x) = \ln P(x|\omega_i) + \ln P(\omega_i)$

Suppose $P(x|\omega_i) = N(\mu_i, \Sigma_i)$, then

$$g_i(x) = -\frac{1}{2}(x - \mu_i)^T \Sigma^{-1}(x - \mu_i) - \frac{d}{2}\ln(2\pi) - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$

Discriminant Functions for Normal densities $\Sigma_i = \sigma^2 I$

Since features are statistically independent and each feature has same covariance σ. Geometrically, samples fall in equal size hyperspherical clustered centered at μ_i.
We have $|\Sigma_i| = \sigma^{2d}$ and $\Sigma_i^{-1} = \frac{1}{\sigma^2} I$ then

$$g_i(x) = -\frac{||x - \mu_i||^2}{2\sigma^2} + \ln P(\omega_i)$$

$$\text{where } ||x - \mu_i||^2 = (x - \mu_i)^T (x - \mu_i)$$

$$g_i(x) = -\frac{1}{2\sigma^2}[x^T x - 2\mu_i^T x + \mu_i^T \mu_i] + \ln P(\omega_i)$$

By ignoring $x^T x$ (Since it is same for all the discriminant functions). We get

$$g_i(x) = \omega_i^T x + \omega_{i0} \text{ which is linear}$$

$$\text{where } \omega_i = \frac{1}{\sigma^2}\mu_i,\ \omega_{i0} = -\frac{1}{2\sigma^2}\mu_i^T \mu_i + \ln P(\omega_i)$$

Discriminant Functions for Normal densities $\Sigma_i = \Sigma$

In this case we get

$$\begin{aligned} g_i(x) &= -\frac{1}{2}(x-\mu_i)^T\Sigma^{-1}(x-\mu_i) + \ln P(\omega_i) \\ &= \omega_i^T x + \omega_{i0} \end{aligned}$$

Where $\omega_i = \Sigma^{-1}\mu_i$, $\omega_{i0} = -\frac{1}{2}\mu_i^T\Sigma^{-1}\mu_i + \ln P(\omega_i)$

Discriminant Functions for Normal densities Σ_i = arbitrary

In this case we get

$$g_i(x) = -\frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i) - \frac{d}{2}\ln(2\pi) - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$
$$= x^T\Omega_i x + \omega_i^T x + \omega_{i0}$$

Where,

$$\Omega_i = -\frac{1}{2}\Sigma_i^{-1},$$
$$\omega_i = \Sigma^{-1}\mu_i$$
$$\omega_{i0} = -\frac{1}{2}\mu_i^T\Sigma_i^{-1}\mu_i - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$

Summary

- Yes! Machine learning is very exiting field and it has many applications
- Data and Models
- Machine learning workflow
- Distance based classifiers
- Bayes decision theory

References:

- Chapter 2, Pattern Classification by Duda, Hart and Stork

Acknowledgements

These notes have been prepared over a period of time and I have referred to many sources. Where ever it is possible, I acknowledged the sources; if I have missed any, my apologies and acknowledgments can be implicitly assumed. Several of my students also contributed to drawing figures etc., and I am very thankful to them