ECE 417 Lecture 6: kNN and Linear Classifiers

Amit Das

09/14/2017

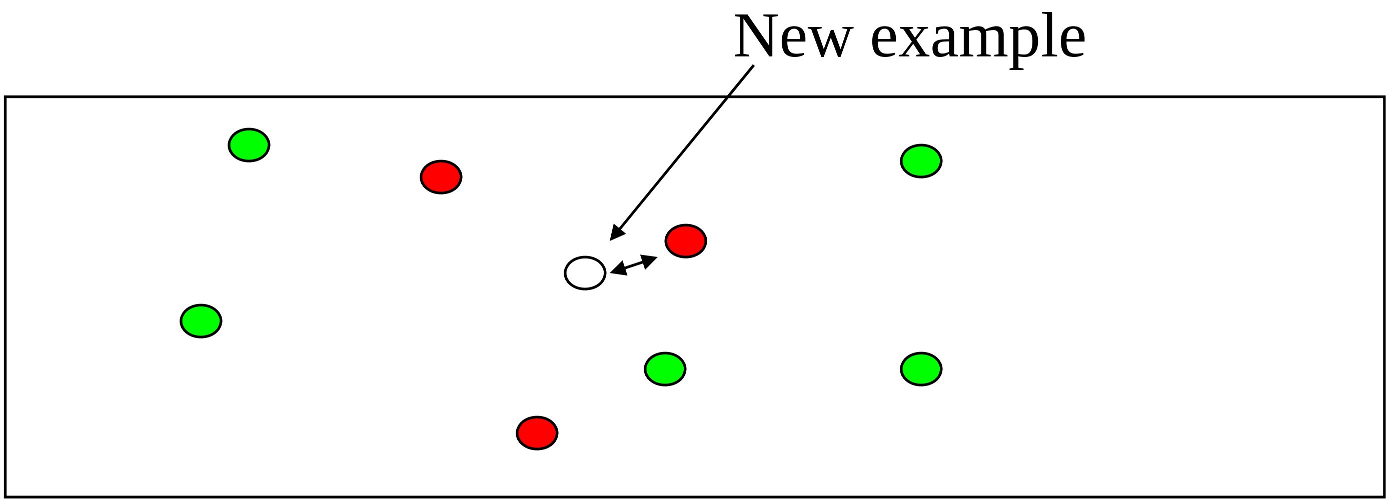
Acknowledgments

- http://research.cs.tamu.edu/prism/lectures/pr/pr_l8.pdf

- http://classes.engr.oregonstate.edu/eecs/spring2012/cs534/notes/knn.pdf

Nearest Neighbor Algorithm

- Remember all training examples
- Given a new example **x,** find the its closest training example <**x**^{i}, *y*^{i}> and predict y^i

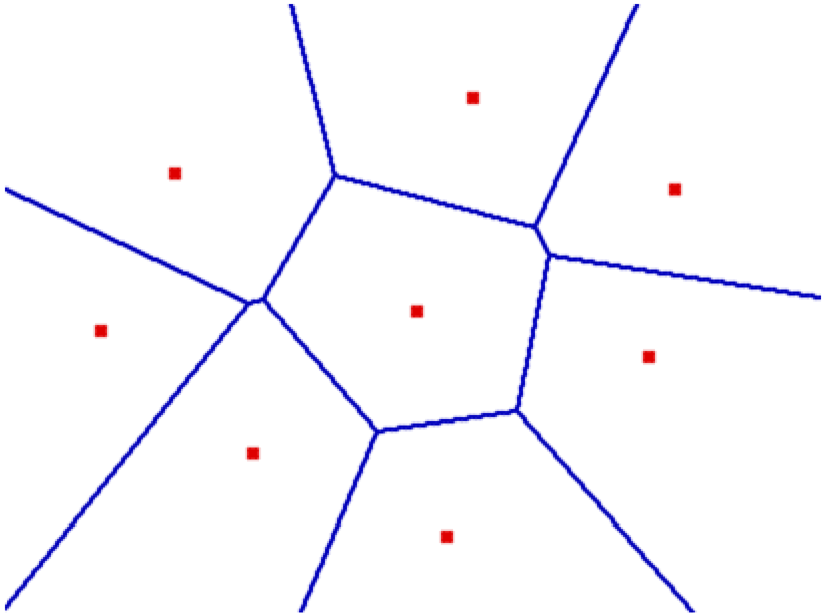


- How to measure distance – Euclidean (squared):

$$\left\|\mathbf{x}-\mathbf{x}^i\right\|^2=\sum_j (x_j - x_j^i)^2$$

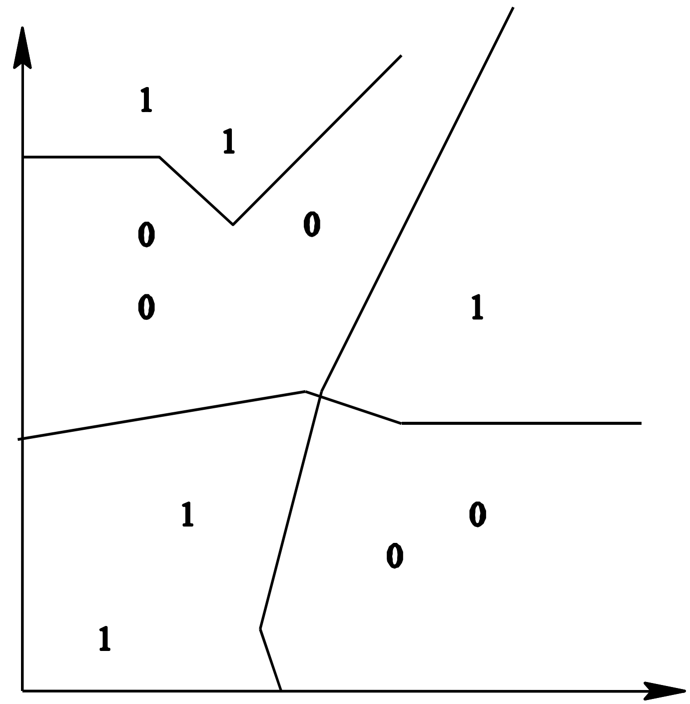
Decision Boundaries: The Voronoi Diagram

- Given a set of points, a **Voronoi diagram** describes the areas that are nearest to any given point.
- These areas can be viewed as zones of control.

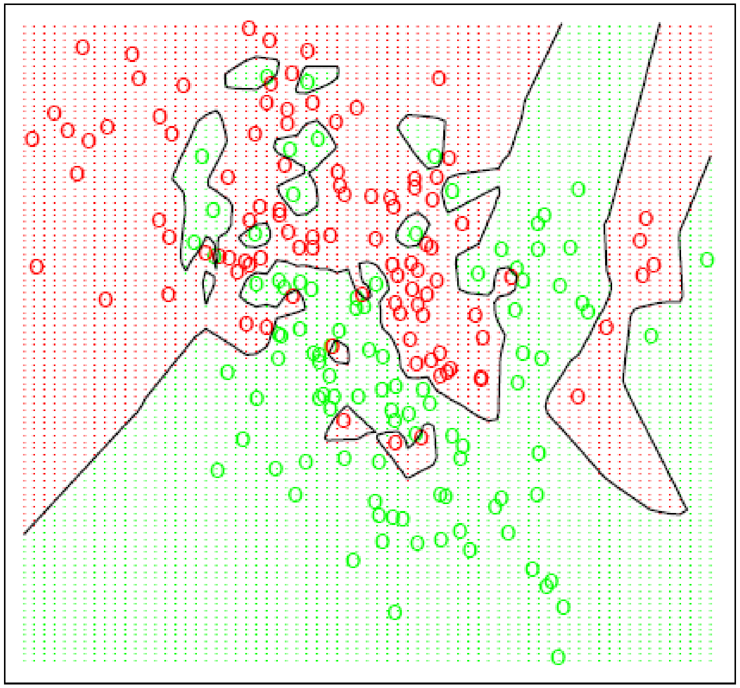


Decision Boundaries: The Voronoi Diagram

- Decision boundaries are formed by a **subset** of the Voronoi diagram of the training data
- Each line segment is equidistant between two points **of opposite class**.
- The more examples that are stored, the more fragmented and complex the decision boundaries can become.



Decision Boundaries

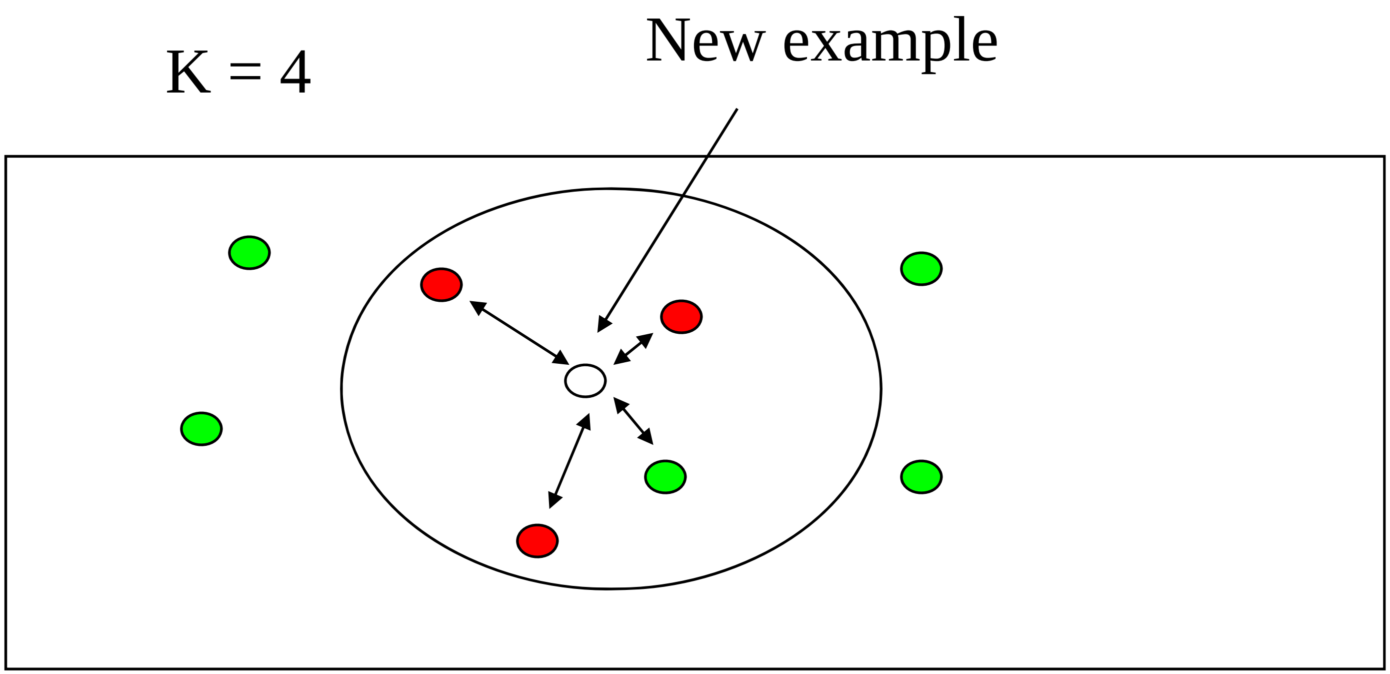


With large number of examples and possible noise in the labels, the decision boundary can become nasty!

We end up overfitting the data

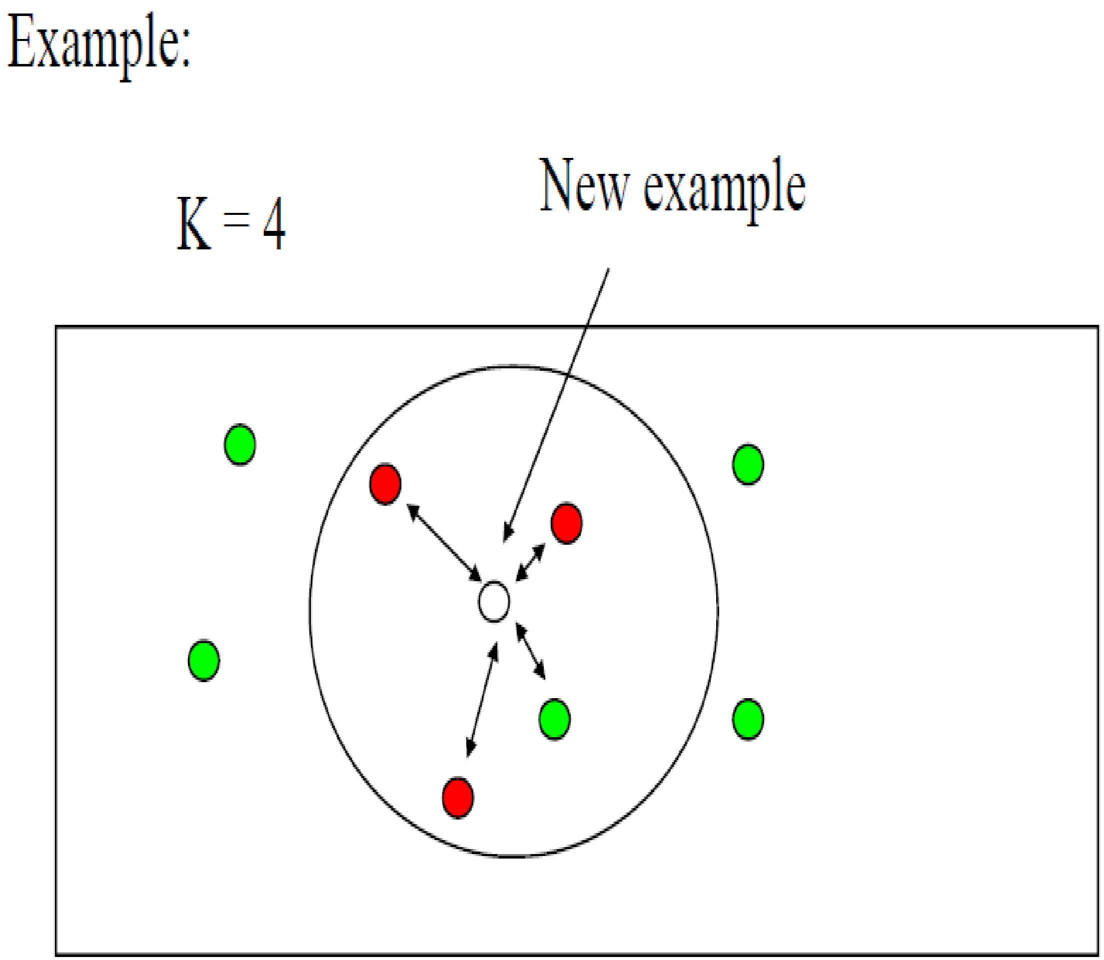
K-Nearest Neighbor

Example:



Find the ***k*** nearest neighbors and have them vote. Has a smoothing effect. This is especially good when there is noise in the class labels.

k-Nearest Neighbor: Probabilistic Interpretation



- Can interpret k-NN as Maximum Aposteriori (MAP) Classifier a.k.a. Baye's classifier.
- Posterior Probability = p(class|example)

$$p(y = \mathrm{red}|x) = \frac{3}{4}$$

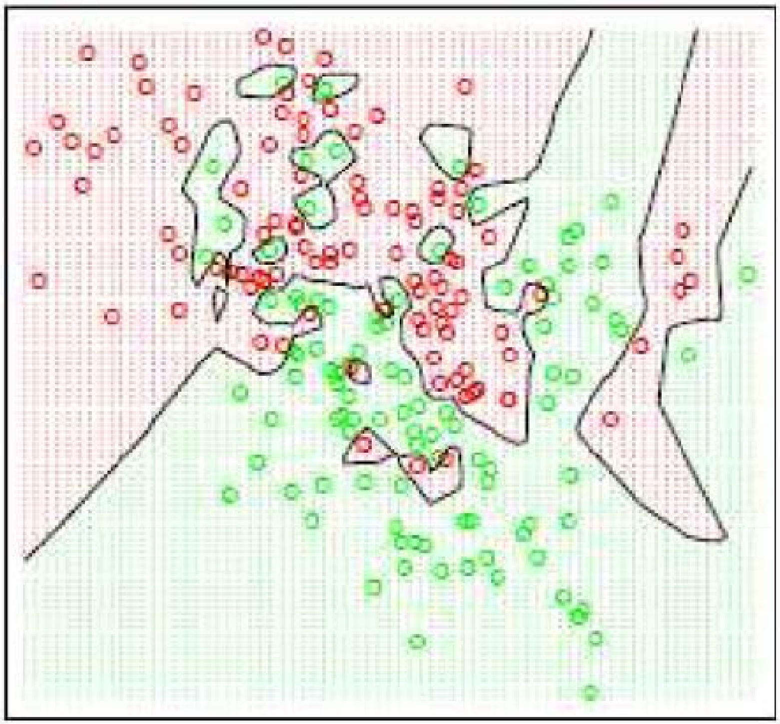
$$p(y = \mathrm{green}|x) = \frac{1}{4}$$

- k-NN uses MAP Rule:

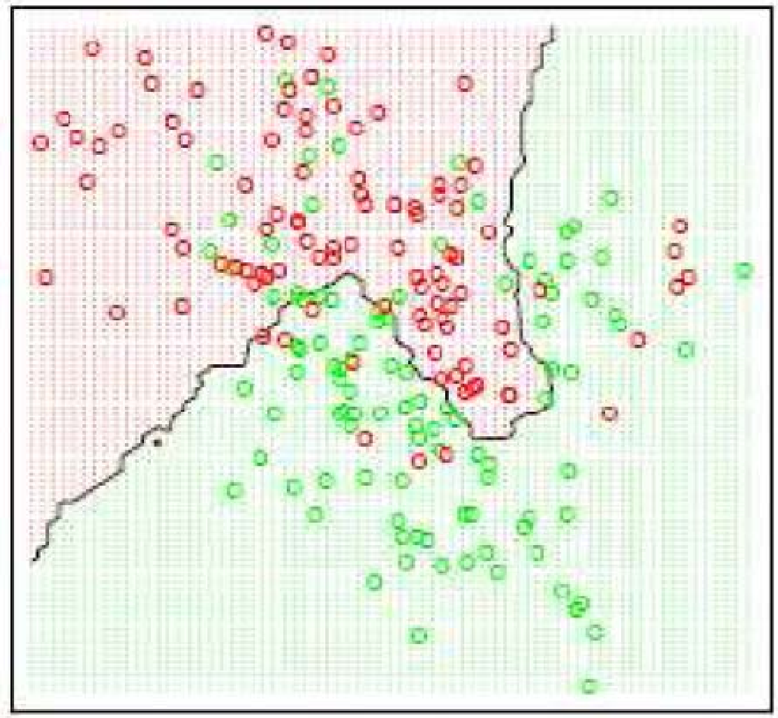
$$\begin{aligned} y^\star &= \arg\max p(y|x) \\ &= \mathrm{red} \end{aligned}$$

Effect of K

K=1



K=15



Figures from Hastie, Tibshirani and Friedman (Elements of Statistical Learning)

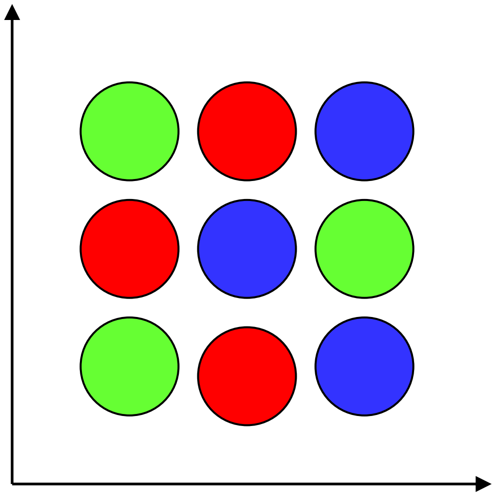
Larger k produces smoother boundary effect and can reduce the impact of class label noise.

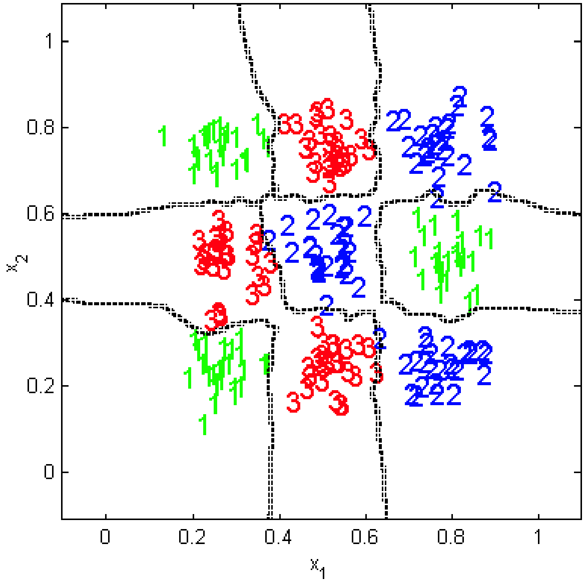
But when K = N, we always predict the majority class

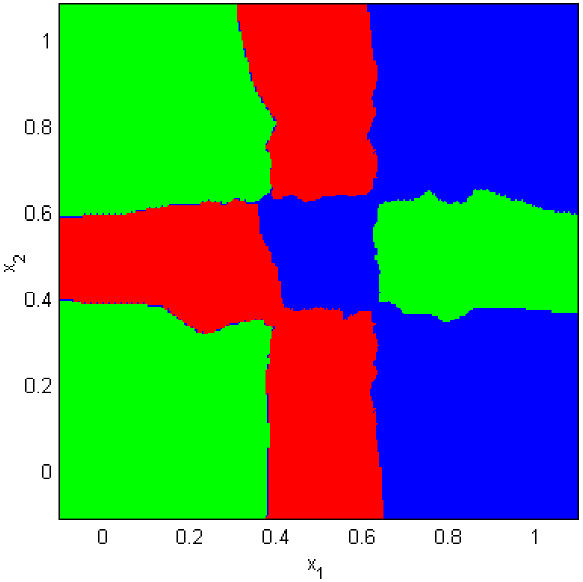
kNN in action

Example I

- Three-class 2D problem with non-linearly separable, multimodal likelihoods
- We use the kNN rule ($k = 5$) and the Euclidean distance
- The resulting decision boundaries and decision regions are shown below

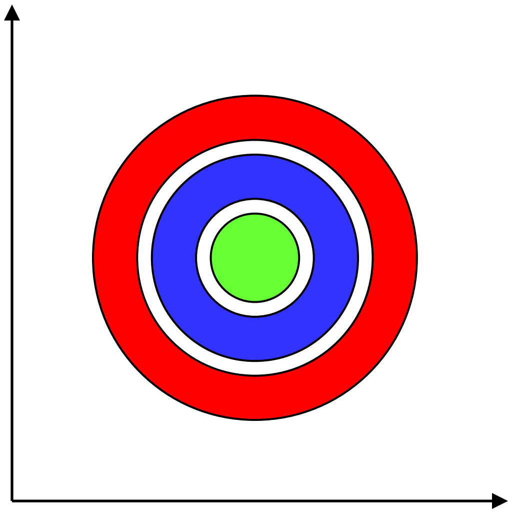


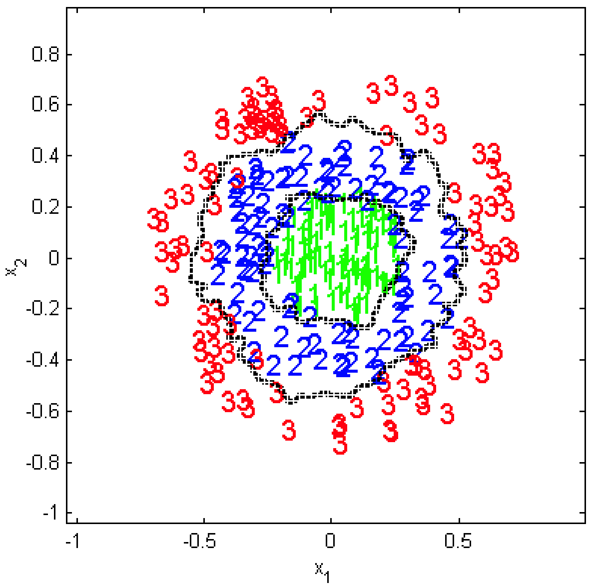


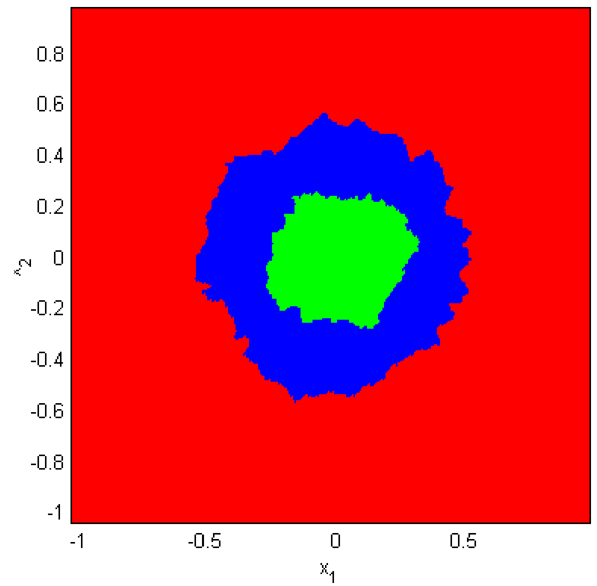


Example II

- Two-dim 3-class problem with unimodal likelihoods with a common mean; these classes are also not linearly separable
- We used the kNN rule ($k = 5$), and the Euclidean distance as a metric

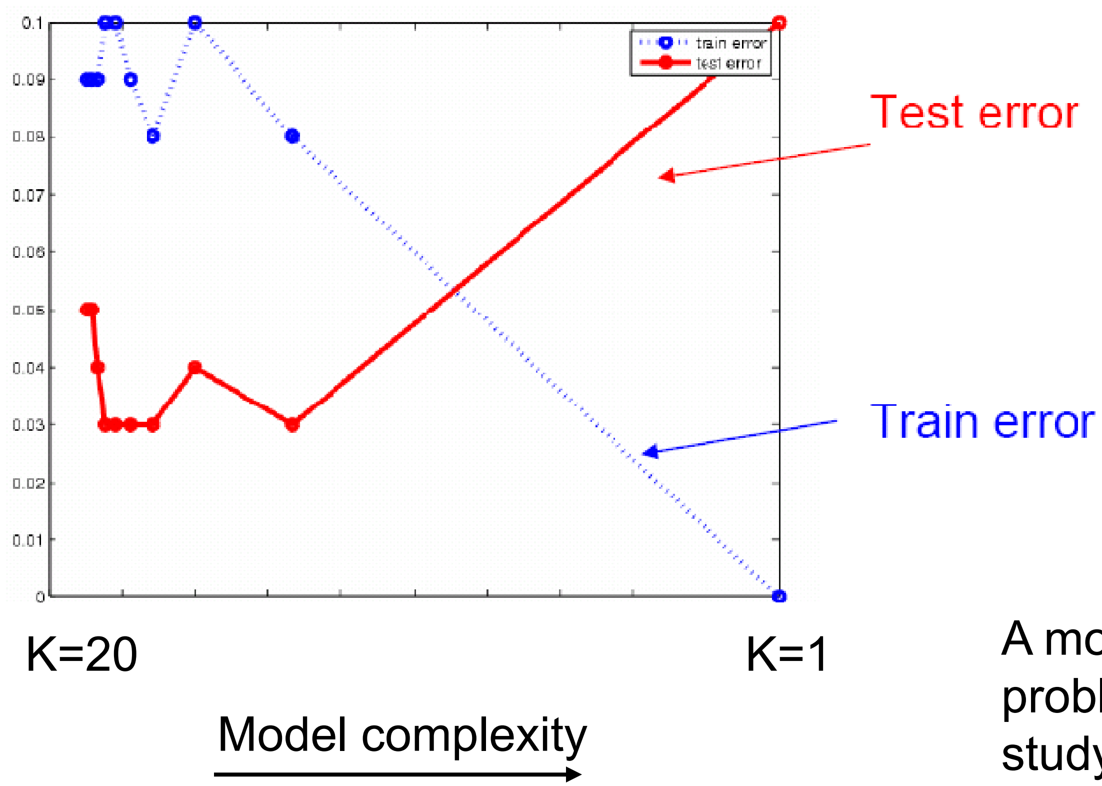






Question: how to choose k?

- Can we choose k to minimize the mistakes that we make on training examples (*training error*)?



A model selection problem that we will study later

Characteristics of the k-NN Classifier

- Advantages:
  - Simple interpretation of Baye's classifier
  - Can discover complex non-linear boundaries between classes
  - Easy to Implement (10 effective lines of Matlab code)
- Disadvantages:
  - Lazy learning algorithm: It defers the "learning" until a test example is given. Thus, it doesn't "learn" from the training data. It actually memorizes all the data.
  - Considers all training data before it can classify a test example → Large storage and computation requirements.
  - Susceptible to curse of dimensionality.

09/14/2017 (1)

ECE 417 (Machine Learning)

Baye's Classifier

Amit Das

Motivation: Design an automatic fruit detector by observing color. $\boxed{\text{"Peaches vs Apples"}}$

Classes: Class = { peaches, apples }

or

$$Y = \{0, 1\}$$

Sometimes called : C_0 , C_1 (machine learning)

H_0 , H_1 (hypothesis in communications/statistics)

Observations: $X \in \mathbb{R}^n$

For the fruit detector problem, $n = 1$ (assume)

since 'X' represents color.

Goal: What threshold (τ) will you choose so that the prob of error (P_e) is minimized?

Here, $\tau \in \mathcal{X}$ (the domain of X)

For the fruit detector, 'τ' is also some color.

Notations: X = observation (r.v.)

Y = label / class (r.v.)

$\hat{Y}$ = predicted label / class (r.v.)

($\hat{Y}$ is a fn. of X. Thus, $\hat{Y} \triangleq \hat{Y}(X)$)

Error: $\boxed{\hat{Y} \neq Y}$

Predicted class $\hat{Y}$ \ Y (True Class)	1	0
1	No error	False Alarm error
0	Miss error	No error

Terminology

Communications Theory	FA error	Miss error
Statistics	Type 1	Type II
Machine Learning	FPR (false +ve rate)	FNR (false -ve rate)

Prob. of error = P_e

$$= P(\hat{y} \neq y)$$

$$= \sum_{y=0}^{1} P(\hat{y} \neq y, y)$$

$$= \sum_{y=0}^{1} P(\hat{y} \neq y \mid y) P(y)$$

$$= P(\hat{y} \neq y \mid y=0) P(y=0) + P(\hat{y} \neq y \mid y=1) P(y=1)$$

$$= \underbrace{P(\hat{y}=1 \mid y=0)}_{P_{FA}} \underbrace{P(y=0)}_{\pi_0} + \underbrace{P(\hat{y}=0 \mid y=1)}_{P_M} \underbrace{P(y=1)}_{\pi_1}$$

$$= \pi_0 P_{FA} + \pi_1 P_M$$

$$\boxed{P_e = \pi_0 P_{FA} + \pi_1 P_M} \quad — (1)$$

$$\left(\begin{array}{l} (a)\ \pi_0 = 1 - \pi_1 \\ (b)\ P_{FA} \neq 1 - P_M \end{array} \right)$$

then our objective is:

Find τ s.t. P_e is minimized

i.e. $\tau^* = \arg\min_{\tau} P_e(\tau)$

Soln:

It can be shown that the Baye's classifier is the optimal classifier that minimizes P_e.

Baye's Classifier: $\hat{y} = \underset{y}{\operatorname{argmax}} \; P(y|x)$ ← posterior probability of class y given that you have observed x

$$= \underset{y}{\operatorname{argmax}} \left\{ P(y=1|x),\ P(y=0|x) \right\}$$

$$= \begin{cases} 1, & P(y=1|x) \geq P(y=0|x) \\ 0, & P(y=1|x) < P(y=0|x) \end{cases}$$

$$\Leftrightarrow \quad P(y=1|x) \underset{\hat{y}=0}{\overset{\hat{y}=1}{\gtrless}} P(y=0|x)$$

$$\Leftrightarrow \quad \boxed{\frac{P(y=1|x)}{P(y=0|x)} \underset{\hat{y}=0}{\overset{\hat{y}=1}{\gtrless}} 1} \quad -(2)$$

→ MAP rule (Max aposteriori rule for uniform loss)

$$\Leftrightarrow \quad \frac{P(x|y=1)}{P(x|y=0)} \underset{\hat{y}=0}{\overset{\hat{y}=1}{\gtrless}} \eta, \quad \eta = \frac{\pi_0}{\pi_1}$$

Note: When $\eta = 1$ (equal priors), MAP rule becomes Maximum Likelihood Rule.

$$\Leftrightarrow \quad \boxed{L(x) \underset{\hat{y}=0}{\overset{\hat{y}=1}{\gtrless}} \eta} \quad \text{--- (3)} \quad \leftarrow$$

MAP rule in terms of ratio of likelihoods $L(x) \triangleq \frac{p(x|y=1)}{p(x|y=0)}$

A.k.a:

1. Likelihood ratio test (LRT) (communications theory, stats)
2. Discriminant Analysis (Machine Learning)

In particular, when $\delta(x) = \log L(x) - \log \eta$ is a:

(a) linear fn. of x $\Leftrightarrow$ $\delta(x)$ = linear discriminant

(b) quadratic fn. of x $\Leftrightarrow$ $\delta(x)$ = quadratic discriminant

Consider

$y=0 : \; x \sim N(\mu_0, \sigma_0)$ and $\sigma_0 = \sigma_1$

$y=1 : \; x \sim N(\mu_1, \sigma_1)$ $\mu_1 > \mu_0$

Thus,

Likelihood of x for class 0 $= p(x|y=0) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_0)^2}{2\sigma^2}}$

Likelihood of x for class 1 $= p(x|y=1) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_1)^2}{2\sigma^2}}$

Thus $L(x) = \dfrac{p(x|y=1)}{p(x|y=0)}$

(c)

$$= \frac{\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_1)^2}{2\sigma^2}}}{\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_0)^2}{2\sigma^2}}} = e^{\frac{-1}{2\sigma^2}\left[(x-\mu_1)^2 - (x-\mu_0)^2\right]}$$

$$L(x) \gtrless \eta$$

$$\Leftrightarrow e^{-\frac{1}{2\sigma^2}\left[(x-\mu_1)^2 - (x-\mu_0)^2\right]} \underset{\hat{y}=0}{\overset{\hat{y}=1}{\gtrless}} \ln(\eta)$$

Take log(·) on both sides,

(since log(·) is monotonic) $\Leftrightarrow$

$$\boxed{x \underset{\hat{y}=0}{\overset{\hat{y}=1}{\gtrless}} \frac{\sigma^2}{(\mu_1-\mu_0)} \ln(\eta) + \left(\frac{\mu_1+\mu_0}{2}\right)} \quad \text{--- (4)}$$

Note that $\log L(x)$ is linear in x. Hence,

$$\delta(x) = x - \frac{\sigma^2}{(\mu_1-\mu_0)} \ln(\eta) - \left(\frac{\mu_1+\mu_0}{2}\right)$$ is a linear discriminant. (LDA)

Therefore, (4) can also be expressed as,

$$\delta(x) = x - \frac{\sigma^2}{(\mu_1-\mu_0)} \ln\eta - \left(\frac{\mu_1+\mu_0}{2}\right) \underset{\hat{y}=0}{\overset{\hat{y}=1}{\gtrless}} 0$$

$$= \frac{(\mu_1-\mu_0)x}{\sigma^2} - \ln(\eta) - \frac{(\mu_1-\mu_0)(\mu_1+\mu_0)}{2\sigma^2} \underset{\hat{y}=0}{\overset{\hat{y}=1}{\gtrless}} 0$$

$$\Leftrightarrow \hat{y}(x) = \begin{cases} 1, & \delta(x) \geq 0 \\ 0, & \delta(x) < 0 \end{cases}$$

This is just a sign test of $\delta(x)$

Extensions

(a) $\delta(x)$ linear in x $\Rightarrow$ LDA (lin. discriminant analysis)

quadratic in x $\Rightarrow$ QDA (quadratic discriminant analysis)

If we consider,

$$y=0 : \quad x \sim \mathcal{N}(\mu_0, \sigma_0) \qquad \boxed{\sigma_0 \neq \sigma_1}$$

$$y=1 : \quad x \sim \mathcal{N}(\mu_1, \sigma_1)$$

then, $\delta(x)$ will be a fn. of x^2

Then $\delta(x)$ is called a Quadratic Discriminant.

(b) For n-dim features, i.e., $x \in \mathbb{R}^n$ and assuming both $p(x|y=0)$, $p(x|y=1)$ Gaussian distributed,

$$y=0 : \quad x \sim \mathcal{N}(\mu_0, \Sigma_0) \qquad \mu_k \in \mathbb{R}^n$$

$$y=1 : \quad x \sim \mathcal{N}(\mu_1, \Sigma_1) \qquad \Sigma_k \succeq 0, \quad k=0,1$$

$$\mu_1 > \mu_0, \quad \Sigma_0 = \Sigma_1 = \Sigma$$

LDA: $$\delta(x) = x^T \Sigma^{-1}(\mu_1 - \mu_0) - \frac{1}{2}(\mu_1+\mu_0)^T \Sigma^{-1}(\mu_1 - \mu_0) + \log\frac{\hat{\pi}_1}{\hat{\pi}_0} \underset{\hat{y}=0}{\overset{\hat{y}=1}{\gtrless}} 0$$

This is a general case of the 1-dim LDA.

(c) If

$$y=0: \quad x \sim N(\mu_0, \Sigma_0)$$

$$y=1: \quad x \sim N(\mu_1, \Sigma_1)$$

$$\Sigma_1 \neq \Sigma_0, \quad \mu_1 > \mu_0$$

Then

$$\delta(x) = \frac{1}{2} x^T \left(\Sigma_0^{-1} - \Sigma_1^{-1}\right) x + \left(\mu_1^T \Sigma_1^{-1} - \mu_0^T \Sigma_0^{-1}\right) x$$

$$- \frac{1}{2} \left[\mu_1^T \Sigma_1^{-1} \mu_1 - \mu_0^T \Sigma_0^{-1} \mu_0 \right] + \frac{1}{2} \log \frac{|\Sigma_0|}{|\Sigma_1|} + \log \frac{\pi_1}{\pi_0} \underset{\hat{y}=0}{\overset{\hat{y}=1}{\gtrless}} 0$$

is a quadratic discriminant.

Terminology

$\delta(x)$	Machine Learning	Communication Theory
linear in x	LDA	Matched Filter Detector
quadratic in x	QDA	Quadratic Detector (e.g. phase detection of signal)

— End of extensions —

Error Evaluation

Again consider,

$$(\text{peaches})\ y=0 \ : \ x \sim N(\mu_0, \sigma) \qquad (\text{equal variance})$$

$$(\text{apples})\ y=1 \ : \ x \sim N(\mu_1, \sigma)$$

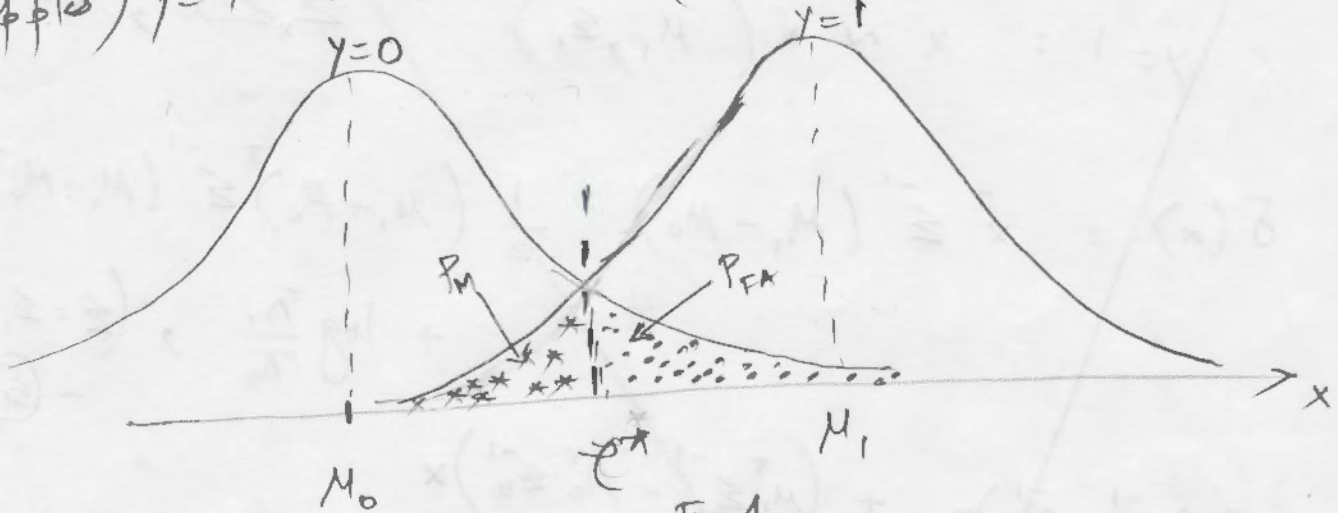


Fig 1

$$c^* = c_{optimal} = \frac{\sigma^2}{(\mu_1 - \mu_0)} \ln\left(\frac{\pi_0}{\pi_1}\right) + \left(\frac{\mu_1 + \mu_0}{2}\right) \quad (\text{from } 4)$$

The decision rule is

$$\hat{y}(x) = \begin{cases} 1, & x \geq c^* \\ 0, & x < c^* \end{cases} \quad - \ 7$$

$$P_{FA} = p\left(\hat{y} = 1 \mid y = 0\right) = \text{Dotted area in Fig 1}$$

$$= \int_{c^*}^{\infty} \underbrace{\frac{1}{\sqrt{2\pi\sigma^2}}\, e^{\frac{(x-\mu_0)^2}{2\sigma^2}}}_{p(x \mid y=0)}\, dx = Q\left(\frac{c^* - \mu_0}{\sigma}\right)$$

$$P_M = P(\hat{y}=0 \mid y=1) = \text{Area with "*" in Fig 1}$$

$$= \int_{-\infty}^{\tau^*} \underbrace{\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_1)^2}{2\sigma^2}}}_{P(x|y=1)} dx$$

$$= \Phi\left(\frac{\tau^* - \mu_1}{\sigma}\right) \qquad \left(\Phi = \text{cumulative distribution fn}\right)$$

$$= Q\left(\frac{\mu_1 - \tau^*}{\sigma}\right) \qquad \left(\text{since } \Phi(x) = Q(-x)\right)$$

$$\therefore P_e = \pi_0 P_{FA} + \pi_1 P_M \qquad (\text{rewriting } 1)$$

$$\therefore P_e(\tau^*) = \pi_0 Q\left(\frac{\tau^* - \mu_0}{\sigma}\right) + \pi_1 Q\left(\frac{\mu_1 - \tau^*}{\sigma}\right) \quad - (8)$$

Since $P_e(\tau^*)$ is the minimum prob. error,

$$P_e(\tau^*) < P_e(\tau), \quad \tau \neq \tau^*$$

<u>Another useful probability:</u>

$$P_D = P(\hat{y}=1 \mid y=1) = \int_{\tau^*}^{\infty} P(x \mid y=1)\, dx$$

(Detection)

$$= \int_{\tau^*}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_1)^2}{2\sigma^2}} = Q\left(\frac{\tau^* - \mu_1}{\sigma}\right)$$

Other Useful Metrics

- Precision

- Recall / Sensitivity

- Accuracy

- F1 score

Consider the peaches vs apples problem in Fig 1 again. This time we'll give the counts

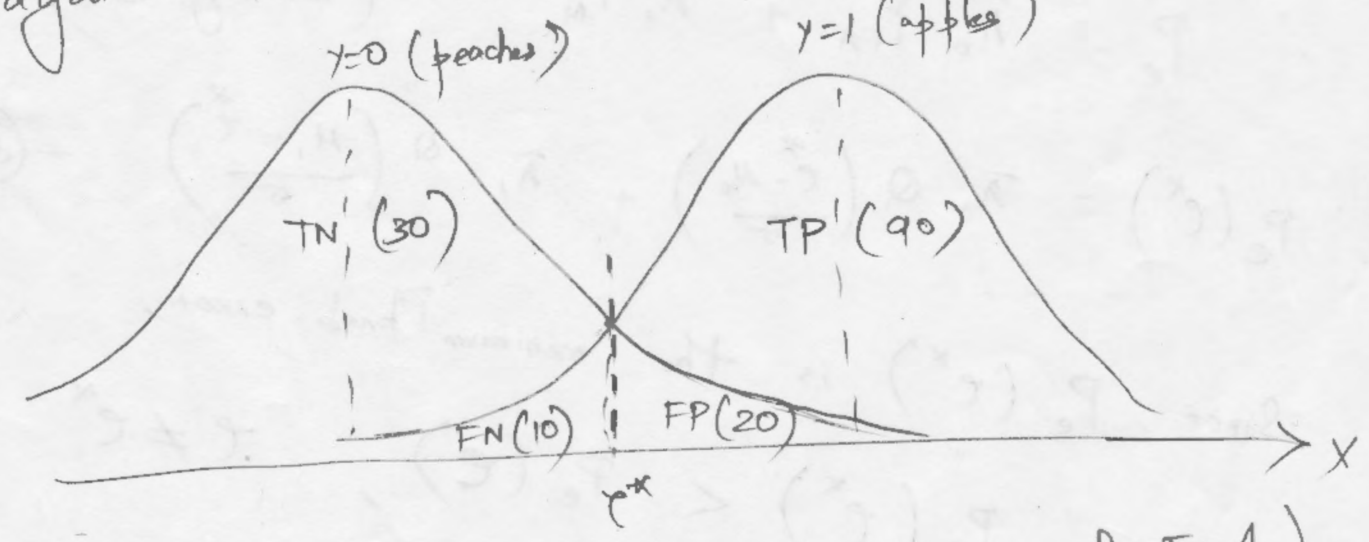


Fig 2 (Histogram version of Fig 1)

TP = true +ve

FP = false +ve

TN = true -ve

FN = false -ve

Total # of peaches = TN + FP = 50

Total # of apples = TP + FN = 100

Let's evaluate all the useful metrics

(a) $$P_{FA} = \frac{FP}{FP+TN} = \frac{20}{20+30} = \frac{2}{5} = 0.4$$

(P_{FA} = Type I error)

(b) $$P_M = \frac{FN}{FN+TP} = \frac{10}{10+90} = 0.1$$

(P_M = Type II error)

(c) $$P_D = \frac{TP}{TP+FN} = \frac{90}{90+10} = 0.9 = 1 - P_M$$

(P_D = Recall = sensitivity = TPR)

"Recall" means "What % of +ve examples (apples) did we recover?"

(d) $$\text{Precision} = \frac{TP}{TP+FP} = \frac{90}{90+20} = \frac{9}{11} = 0.82$$

"Precision" means "How precise was our classifier in finding the +ve examples (apples)?"

e) Accuracy $= \dfrac{TP + TN}{(TP + FN) + (TN + FP)}$

$$= \frac{90 + 30}{150} = \frac{4}{5} = 0.8$$

f) F_1 score $= \dfrac{2}{\dfrac{1}{\text{Recall}} + \dfrac{1}{\text{Precision}}} = \dfrac{2}{\dfrac{10}{9} + \dfrac{11}{9}}$

$$= 18/21 = 6/7$$

where, $\underset{\text{(worst)}}{0} \leq F_1 \leq \underset{\text{(best)}}{1}$

F_1 score penalizes low values of recall or precision. Thus, it's a score describing overall performance of the classifier

Problem 1 (20 points)

You want to classify zoo animals. Your zoo only has two species: elephants and giraffes. There are more elephants than giraffes: if Y is the species,

$$\begin{aligned} p_Y(\text{elephant}) &= \frac{e}{e+1} \\ p_Y(\text{giraffe}) &= \frac{1}{e+1} \end{aligned}$$

where $e = 2.718\ldots$ is the base of the natural logarithm. The height of giraffes is Gaussian, with mean $\mu_G = 5$ meters and variance $\sigma_G^2 = 1$. The height of elephants is also Gaussian, with mean $\mu_E = 3$ and variance $\sigma_E^2 = 1$. Under these circumstances, the minimum probability of error classifier is

$$\hat{y}(x) = \begin{cases} \text{giraffe} & x > \theta \\ \text{elephant} & x < \theta \end{cases}$$

Find the value of θ that minimizes the probability of error.

Problem :

$$Y = \begin{cases} \text{elephant} = 0 \\ \text{giraffe} = 1 \end{cases} \qquad X = \text{height}$$

$$P(Y = \text{elephant}) = \frac{e}{1+e} = \pi_0$$

$$P(Y = \text{giraffe}) = \frac{1}{1+e} = \pi_1$$

$$Y = 0 : \quad x \sim N(3, 1)$$

$$Y = 1 : \quad x \sim N(5, 1)$$

Classify b/w elephants & giraffes based on their heights s.t it yields min prob error. Find the min P_{error}.

Soln:

$$\mu_0 = 3, \quad \mu_1 = 5$$

$$\sigma_0 = \sigma_1 = \sigma = 1$$

Using 4,

$$x \underset{\hat{y}=0}{\overset{\hat{y}=1}{\gtrless}} \frac{\sigma^2}{(\mu_1 - \mu_0)} \ln\left(\frac{\pi_0}{\pi_1}\right) + \left(\frac{\mu_1 + \mu_0}{2}\right)$$

$$= \frac{1}{(5-3)} \ln(e) + \frac{5+3}{2}$$

$$= \frac{1}{2} + 4 = 4.5 = e^*$$

$$\therefore \hat{y} = \begin{cases} 1, & x \geq 4.5 \\ 0, & x < 4.5 \end{cases}$$

Using 8 to find P_e, we've

$$P_e = \pi_0 \, Q\left(\frac{e^* - \mu_0}{\sigma}\right) + \pi_1 \, Q\left(\frac{\mu_1 - e^*}{\sigma}\right)$$

$$= \frac{e}{1+e} \, Q\left(\frac{4.5-3}{1}\right) + \frac{1}{1+e} \, Q\left(\frac{5-4.5}{1}\right)$$

$$= \frac{e}{1+e} \, Q(1.5) + \frac{1}{1+e} \, Q(0.5)$$