2.2 STATISTICAL SIGNAL PROCESSING

Dr TIAN Jing

tianjing@nus.edu.sg

Module objective

Module: Statistical signal processing

Knowledge and understanding

- Understand the fundamentals of statistical signal processing, learning distribution from signal, and statistical classification

Key skills

- Design, build, implement a statistical classification approach that could use features extracted from the wavelet transofrmation

Topics

- Statistical signal processing modelling
- Statistical signal classification

Statistical signal processing

- Learning: For a given set of observations $\boldsymbol{X} = [\boldsymbol{X}_1, \boldsymbol{X}_{2,}, \cdots, \boldsymbol{X}_N]$ from some random variables, learn a model $f(\boldsymbol{X}|\boldsymbol{\theta})$ (defined by the parameter $\boldsymbol{\theta}$) to best describe the data.
- Estimation: Given a noisy observation $\boldsymbol{X}_n = \boldsymbol{Z}_n + \boldsymbol{w}_n$, where $\boldsymbol{Z}_n$ is the unknown clean signal and $\boldsymbol{w}_n$ is noise, and the learned model $f(\boldsymbol{X}|\boldsymbol{\theta})$, estimate the unknown clean signal.
- Prediction: Assume that we have two classes of objects as $\omega_1 \text{ and } \omega_2$, and we already have learned models $f(\boldsymbol{X}|\boldsymbol{\theta}_1)$ and $f(\boldsymbol{X}|\boldsymbol{\theta}_2)$ for $\omega_1 \text{ and } \omega_2$, respectively. For a new observation $\boldsymbol{X}_n$, decide which class it belongs to, that is $\boldsymbol{f}(\omega_1, \omega_2|\boldsymbol{X}_n, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$.

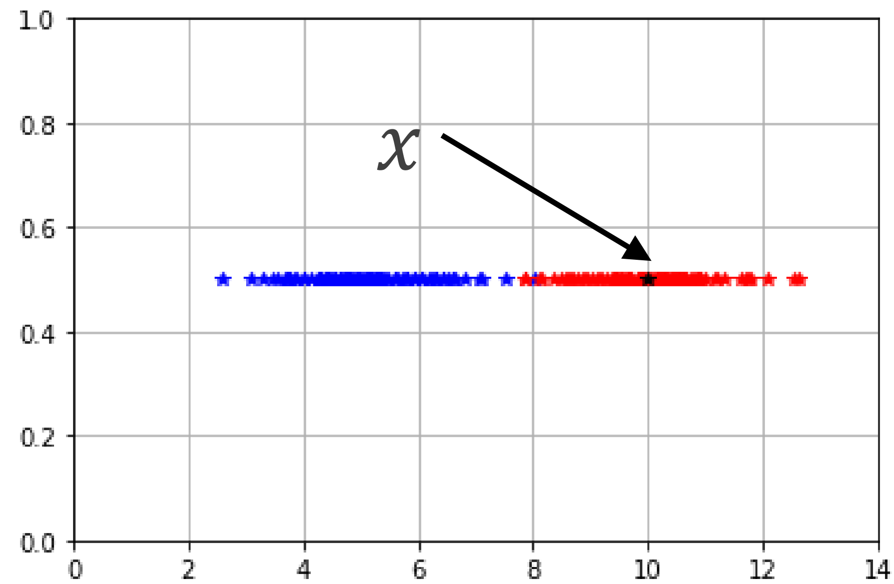
Signal classification

- Given a signal x, two categories of data, say ω_1, ω_2, does the signal x belong to ω_1 or ω_2?
- That means, we need to evaluate $P(\omega_1|x)$ and $P(\omega_2|x)$
- The class posterior probability is

Likelihood Priors

$$P(\omega_i|x) = \frac{P(x|\omega_i)P(\omega_i)}{P(x)}$$

Evidence



Signal classification

- Class priors $P(\omega_i)$
  - How much of each class? $P(\omega_i) \approx N_i/N$
- Class likelihood $P(x|\omega_i)$
  - Requires that we have a model for each ω_i, for example, ω_i can be modelled as Gaussian distribution
- Evidence
  - $P(x) = P(x|\omega_1)P(\omega_1) + P(x|\omega_2)P(\omega_2)$
- Classification rule
  - If $P(\omega_1|x) \geq P(\omega_2|x)$, then x is classified as ω_1
  - If $P(\omega_1|x) < P(\omega_2|x)$, then x is classified as ω_2

Example: Naive Bayes classifier

Given: $\boldsymbol{x}$=(Outlook=*Sunny*, Temperature=*Cool*, Humidity=*High*, Wind=*Strong*)
Predict: PlayTennis Yes or No?

PlayTennis: training examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Example

Given: x=(Outlook=*Sunny*, Temperature=*Cool*, Humidity=*High*, Wind=*Strong*)
Predict: PlayTennis Yes or No?

Bayesian Rule	
P(*Yes*|**x**)	0.0053
[P(*Sunny*|Yes)P(*Cool*| *Yes*)P(*High*|Yes)P(*Strong*| *Yes*)]P(Play= *Yes*)	
P(*No*|**x**)	0.0206
[P(*Sunny*|No) P(*Cool*|No)P(*High*|*No*)P(*Strong*|*No*)]P(Play=*No*)	

Decision: Given the fact $P(Yes|\boldsymbol{x}) < P(No|\boldsymbol{x})$, we decide $\boldsymbol{x}$ to be *No*.

P(Outlook=*Sunny*|Play=*Yes*) = 2/9

P(Temperature=*Cool*|Play=*Yes*) = 3/9

P(Huminity=*High*|Play=*Yes*) = 3/9

P(Wind=*Strong*|Play=*Yes*) = 3/9

P(Play=*Yes*) = 9/14

P(Outlook=S*unny*|Play=*No*) = 3/5

P(Temperature=*Cool*|Play==*No*) = 1/5

P(Huminity=*High*|Play=*No*) = 4/5

P(Wind=*Strong*|Play=*No*) = 3/5

P(Play=*No*) = 5/14

Example

- Probabilistic model

Outlook	Play=*Yes*	Play=*No*
Sunny	2/9	3/5
Overcast	4/9	0/5
Rain	3/9	2/5

Temperature	Play=*Yes*	Play=*No*
Hot	2/9	2/5
Mild	4/9	2/5
Cool	3/9	1/5

Humidity	Play=*Yes*	Play=No
High	3/9	4/5
Normal	6/9	1/5

Wind	Play=*Yes*	Play=*No*
Strong	3/9	3/5
Weak	6/9	2/5

Total: $P(Play = Yes) = 9/14, P(Play = No) = 5/14$

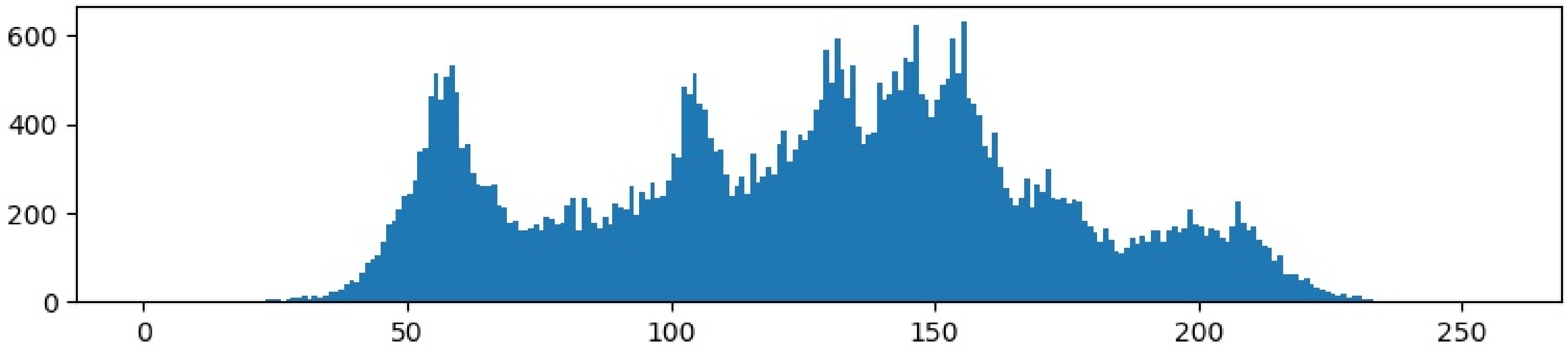
Learning distributions for data

- Problem: Given a collection of examples from some data, estimate its distribution

- Solution:
  - Assign a model to the distribution
  - Learn parameters of model from data

Non-parametric modelling

- Non-parametric models do not specify any priori set of parameters to model the distribution.
- Example: Histogram



Parametric modelling

- Collection of probability distributions which are described by a finite dimensional parameter set, such as
- Bernoulli distribution

$$p(\mathbf{x}|\boldsymbol{\mu}) = \prod_i \mu_i^{x_i}(1-\mu_i)^{x_i}$$

- aussian distribution

$$p(\mathbf{x}|\boldsymbol{\mu},\Sigma) = \frac{1}{\sqrt{(2\pi)^N|\Sigma|}} exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

Reference: http://mathworld.wolfram.com/BernoulliDistribution.html

Learn model parameters: A toy task

Example: **6 3 1 5 4 1 2 4 …**

- A person shoots a loaded dice repeatedly
- You observe the series of outcomes
- You can form a good idea of how the dice is loaded, by finding what the probabilities of the various numbers are for dice
- $\text{Probability}(number) = \frac{\text{Count}(number)}{\text{Sum}(rools)}$

This is a *maximum likelihood* estimate. Estimate that makes the observed sequence of numbers most probable.

Maximum likelihood estimation

- Idea: Assign a form to the distribution, e.g., a Gaussian distribution. Find the *distribution* that best fits the histogram of the data.
- The data are generated by draws from the distribution. Select the distribution that has the *highest* probability of generating the data.
- Given a collection of observations $(X_1, X_2, \ldots)$, estimate mean μ and covariance Θ

$$P(X_1, X_2, \cdots) = \prod_i \frac{1}{\sqrt{(2\pi)^d |\Theta|}} \exp\left(-\frac{1}{2}(X_i - \mu)^T \Theta^{-1} (X_i - \mu)\right)$$

- Maximizing w.r.t μ and Θ gives us

$$\mu = \frac{1}{N}\sum_i X_i \qquad \Theta = \frac{1}{N}\sum_i (X_i - \mu)(X_i - \mu)^T$$

A difficult task

- Additional rule
  - **Two persons** shoot loaded dice (say, red dice and blue dice) repeatedly. The dice are differently loaded for the two of them.
  - We observe the series of outcomes for both persons.
  - Instead there is a “caller” who randomly calls out the outcomes. 40% of the time he calls out the number from the left shooter, and 60% of the time, the one from the right (and you know this)
  - At any time, you do not know which of the two he is calling out
- <u>How do you determine the probability distributions for the two dice? </u>If you do not even know what fraction of time the blue numbers are called, and what fraction are red.

Key idea: Introduce a hidden label

- The caller will call out a number X <u>IF</u>
  - He selects “RED”, and the Red die rolls the number X
  - <u>OR</u>
  - He selects “BLUE” and the Blue die rolls the number X

- P(X) = P(Red)P(X|Red) + P(Blue)P(X|Blue)
  - E.g. P(6) = P(Red)P(6|Red) + P(Blue)P(6|Blue)

- A distribution that *combines* (or *mixes*) multiple multinomials is a *mixture* multinomial

$$P(X) = \sum_{Z} P(Z)P(X|Z)$$

Mixture weights

Component multinomials

Expectation maximization

- Iterative solution

- Get some initial estimates for all parameters
  - Dice shooter example: This includes probability distributions for dice AND the probability with which the caller selects the dice

- Two steps that are iterated:
  - Expectation Step: Estimate statistically, the values of unseen variables
  - Maximization Step: Using the estimated values of the unseen variables as truth, estimates of the model parameters

Expectation maximization

Step 1: Initialization

- We (guess) obtain an initial estimate for the probability distribution of the two sets of dice:

X	1	2	3	4	5	6
$P(X \vert \mathrm{Blue})$	0.3	0.3	0.1	0.1	0.1	0.1
$P(X \vert \mathrm{Red})$	0.4	0.05	0.05	0.05	0.05	0.4

- We (guess) obtain an initial estimate for the probability with which the caller calls out the two shooters

$P(Z = \mathrm{Blue})$	0.5
$P(Z = \mathrm{Red})$	0.5

Expectation maximization

Step 2: Estimate hidden labels

- Every observed roll of the dice contributes to both “Red” and “Blue”

6 4 5 1 2 3 4 5 2 2 1 4 3 4 6 2 1 6

6 (0.8), 4 (0.33), 5 (0.33), 1 (0.57), 2 (0.14), 3 (0.33), 4 (0.33), 5 (0.33), 2 (0.14), 2 (0.14), 1 (0.57), 4 (0.33), 3 (0.33), 4 (0.33), 6 (0.8), 2 (0.14), 1 (0.57), 6 (0.8)

6 (0.2), 4 (0.67), 5 (0.67), 1 (0.43), 2 (0.86), 3 (0.67), 4 (0.67), 5 (0.67), 2 (0.86), 2 (0.86), 1 (0.43), 4 (0.67), 3 (0.67), 4 (0.67), 6 (0.2), 2 (0.86), 1 (0.43), 6 (0.2)

Observed dice value

Probability to be red

Probability to be blue

Recall that we randomly guess

$$P(Z = \text{blue}) = P(Z = \text{red}) = 0.5$$
$$P(6|\text{blue}) = 0.1, P(6|\text{red}) = 0.4$$

For dice 6, we have

$$P(\text{red}|X = 6) = P(X = 6|Z = \text{red})P(Z = \text{red}) = 0.2$$
$$P(\text{blue}|X = 6) = P(X = 6|Z = \text{blue})P(Z = \text{blue}) = 0.05$$

After normalization,

$$P(\text{red}|X = 6) = 0.8, P(\text{blue}|X = 6) = 0.2$$

Expectation maximization

Step 2: Estimate hidden labels

- Every observed roll of the dice contributes to both “Red” and “Blue”
- Total count for “Red” is the sum of all the posterior probabilities in the red column
  - 7.31
- Total count for “Blue” is the sum of all the posterior probabilities in the blue column
  - 10.69

Called	P(red|X)	P(blue|X)
6	.8	.2
4	.33	.67
5	.33	.67
1	.57	.43
2	.14	.86
3	.33	.67
4	.33	.67
5	.33	.67
2	.14	.86
2	.14	.86
1	.57	.43
4	.33	.67
3	.33	.67
4	.33	.67
6	.8	.2
2	.14	.86
1	.57	.43
6	.8	.2
	7.31	10.69

Expectation maximization

Step 3: Update $P(X|\text{Red})$

- Total count for "Red" : 7.31
- Red:
  - Total count for 1: 1.71
  - Total count for 2: 0.56
  - Total count for 3: 0.66
  - Total count for 4: 1.32
  - Total count for 5: 0.66
  - Total count for 6: 2.4
- Updated probability of Red dice:
  - P(1 | Red) = 1.71/7.31 = 0.234
  - P(2 | Red) = 0.56/7.31 = 0.077
  - P(3 | Red) = 0.66/7.31 = 0.090
  - P(4 | Red) = 1.32/7.31 = 0.181
  - P(5 | Red) = 0.66/7.31 = 0.090
  - P(6 | Red) = 2.40/7.31 = 0.328

Called	P(red|X)	P(blue|X)
6	.8	.2
4	.33	.67
5	.33	.67
1	.57	.43
2	.14	.86
3	.33	.67
4	.33	.67
5	.33	.67
2	.14	.86
2	.14	.86
1	.57	.43
4	.33	.67
3	.33	.67
4	.33	.67
6	.8	.2
2	.14	.86
1	.57	.43
6	.8	.2
	7.31	10.69

Expectation maximization

Step 3: Update $P(X|\mathrm{Blue})$

- Total count for "Blue" : 10.69
- Blue:
  - Total count for 1: 1.29
  - Total count for 2: 3.44
  - Total count for 3: 1.34
  - Total count for 4: 2.68
  - Total count for 5: 1.34
  - Total count for 6: 0.6

- Updated probability of Blue dice:
  - P(1 | Blue) = 1.29/11.69 = 0.122
  - P(2 | Blue) = 0.56/11.69 = 0.322
  - P(3 | Blue) = 0.66/11.69 = 0.125
  - P(4 | Blue) = 1.32/11.69 = 0.250
  - P(5 | Blue) = 0.66/11.69 = 0.125
  - P(6 | Blue) = 2.40/11.69 = 0.056

Called	P(red|X)	P(blue|X)
6	.8	.2
4	.33	.67
5	.33	.67
1	.57	.43
2	.14	.86
3	.33	.67
4	.33	.67
5	.33	.67
2	.14	.86
2	.14	.86
1	.57	.43
4	.33	.67
3	.33	.67
4	.33	.67
6	.8	.2
2	.14	.86
1	.57	.43
6	.8	.2
	7.31	10.69

Expectation maximization

Step 4: Update $P(Z = \text{Blue})$ and $P(Z = \text{Red})$

- Total count for “Red” : 7.31
- Total count for “Blue” : 10.69
- Total instances = 18
  - Note 7.31+10.69 = 18
- We also revise our estimate for the probability that the caller calls out Red or Blue
  - i.e the fraction of times that he calls Red and the fraction of times he calls Blue
- $P(Z = \text{Red})$ = 7.31/18 = 0.41
- $P(Z = \text{Blue})$ = 10.69/18 = 0.59

Called	P(red|X)	P(blue|X)
6	.8	.2
4	.33	.67
5	.33	.67
1	.57	.43
2	.14	.86
3	.33	.67
4	.33	.67
5	.33	.67
2	.14	.86
2	.14	.86
1	.57	.43
4	.33	.67
3	.33	.67
4	.33	.67
6	.8	.2
2	.14	.86
1	.57	.43
6	.8	.2
	7.31	10.69

Summary

X	1	2	3	4	5	6
Previously, we randomly guess						
$P(X\|\text{Blue})$	0.3	0.3	0.1	0.1	0.1	0.1
$P(X\|\text{Red})$	0.4	0.05	0.05	0.05	0.05	0.4
Now, we have updated results						
$P(X\|\text{Blue})$	0.122	0.322	0.125	0.250	0.125	0.056
$P(X\|\text{Red})$	0.234	0.077	0.090	0.181	0.090	0.328

	Previous random guess	Updated results
$P(Z = \text{Blue})$	0.5	0.59
$P(Z = \text{Red})$	0.5	0.41

The EM algorithm will continue until the stopping criterion (e.g., # of iterations, change of variables).

Summary

1. Initialize $P(Z), P(X|Z)$
2. Estimate $P(Z|X)$ for each Z, for each called out number. Associate X with each value of Z, with weight $P(Z|X)$
3. Re-estimate $P(X|Z)$ for every value of X and Z, $P(Z|X) = \frac{P(X|Z)P(Z)}{\sum_{Z'} P(Z')P(X|Z')}$
4. Re-estimate $P(Z)$ as $P(Z) = \frac{\sum_X N_X P(Z|X)}{\sum_{Z'} \sum_X N_X P(Z|X)}$, where $P(X|Z) = \frac{\sum_{O \text{ such that } O==X} P(Z|X)}{\sum_O P(Z|O)} = \frac{N_X P(Z|X)}{\sum_X N_X P(Z|X)}$
5. If not converged, return to Step 2

Application: Background modeling in video



Input image   Background image   Foreground image

- Estimate background image (scene modeling)
- Find the color distribution models (foreground and background) at each pixel location
- Subtract the estimated background from the input image

Summary

- Major tasks of statistical signal processing
- Statistical signal modelling including parameter estimation and Bayesian signal classification

Thank you!

Dr TIAN Jing
Email: tianjing@nus.edu.sg