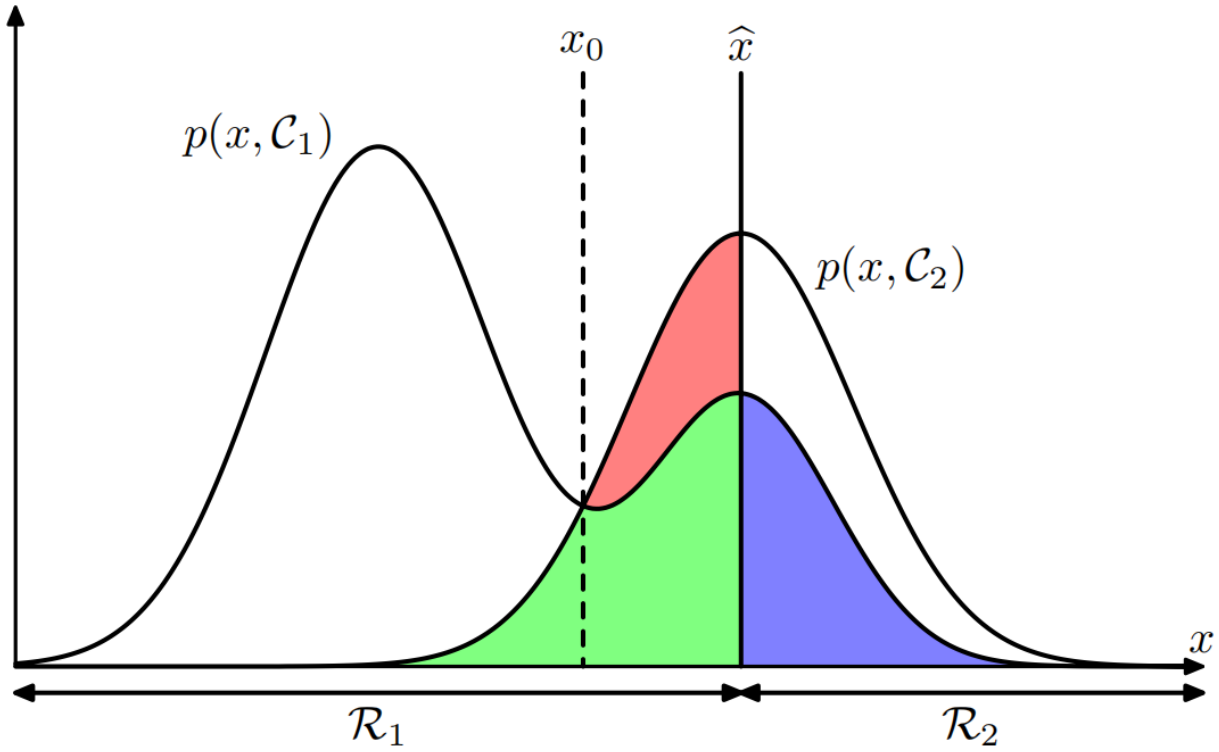
Decoupled PLDA

Dong Wang

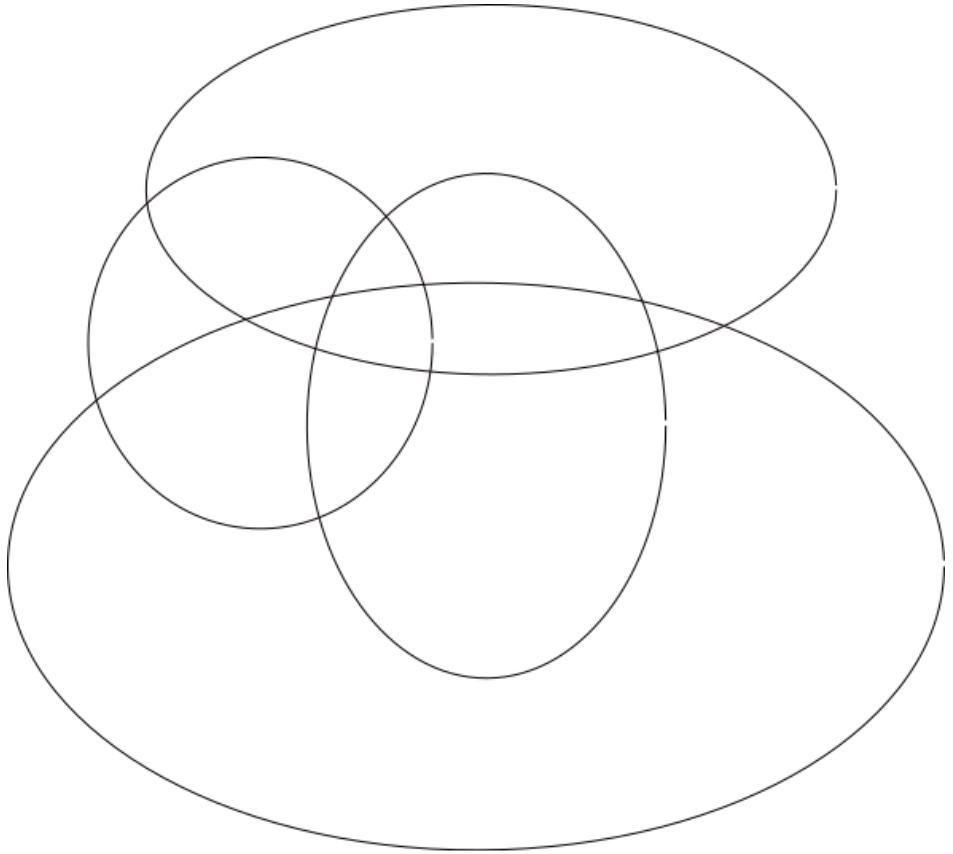
2020/08/17

Decision theory with two classes



- Bishop, PRML, 2006, pg. 40.

Decision with more classes



$$loss_j = \sum_{i \neq j} \ell_{ij} p(c_i|x) = \ell(1 - p(c_j|x))$$

- If the loss for each incorrect decision is equal, then MAP is optimal.

Look into the MAP

$$p(c|x) = \frac{p(x|c)p(c)}{p(x)} = \frac{p(x|c)p(c)}{\sum_c p(x|c)p(c)}$$

Bayes rule

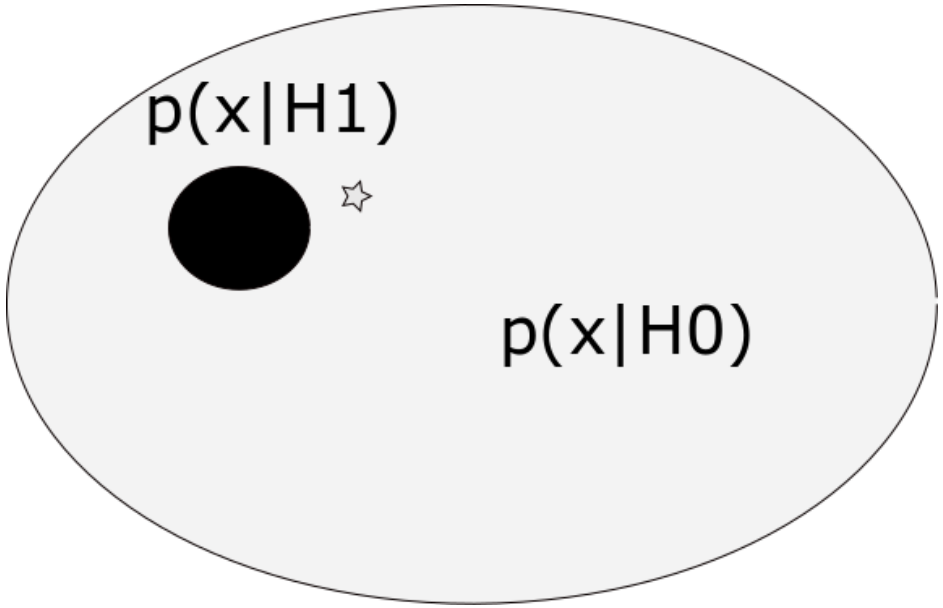
Marginalization rule

Remarks

- The derivation is correct in THEORY. That means:
  - It is nothing to do of the model used.
  - In the case the prior and condition are not accurate, the derivation for the posterior is correct, but the derived posterior is not correct, and the decision is therefore incorrect.

Now move to verification

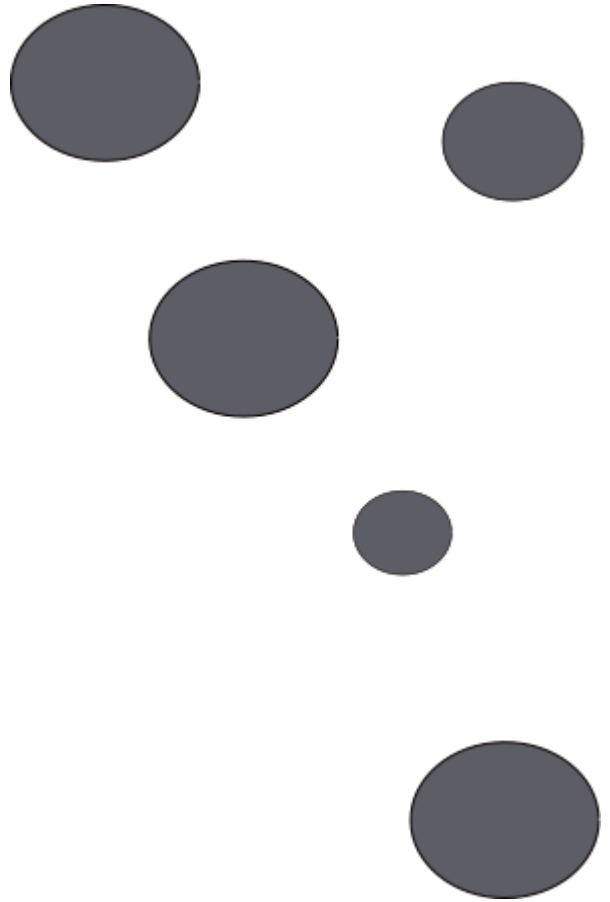
- Verification
  - Given a test sample, and an argued class, test if the argument is correct
  - An open set problem, where we need to consider all potential classes.
  - By comparing likelihood of two hypothesis, we know where we are.



Take a more task-oriented view

- Our goal: obtain the best decision on the test set!
- How do we choose the score?
- Pay attention p(x): it is used to give chance to alternatives!

$$p(c|x) = \frac{p(x|c)p(c)}{\sum_c p(x|c)p(c)}$$



Open-set challenge: how to set the model parameters?

- How to can determine p(x|c) for unseen c?
  - By assuming training and test is the same
- How to define normalization when the test set is not allowed to know?
  - By assuming a global generation model

$$p(c|x) = \frac{p(x|c)p(c)}{\sum_c p(x|c)p(c)} = \frac{p(x|c)p(c)}{\int p(x|c)p(c)\mathrm{dc}}$$

Performance is determined by…

- Request 1: How accurate p(x|c) trained on training data describes test within-class variance.
- Request 2: How accurate p(x) matches $\sum_c$ p(x|c)p(c) on all the test data samples.
- When the two conditions are all perfect, the decision will MBR.
- Note that p(x) approaches to the true marginal does not necessarily improve performance.

$$p(c|x) = \frac{p(x|c)p(c)}{\sum_c p(x|c)p(c)} = \frac{p(x|c)p(c)}{\int p(x|c)p(c)\mathrm{dc}}$$

Must be the posterior conditional coupled?

$$p(c|x) = \frac{p(x|c)p(c)}{\sum_c p(x|c)p(c)} = \frac{p(x|c)p(c)}{\int p(x|c)p(c)\mathrm{dc}}$$

- Theoretically, NO. The performance is not relevant if the conditional is coupled.
- In fact, the only request for p(x) is they correlates to $\sum_c$ p(x|c)p(c) on the test data; under this condition, it could be anything.

Decouple the conditional

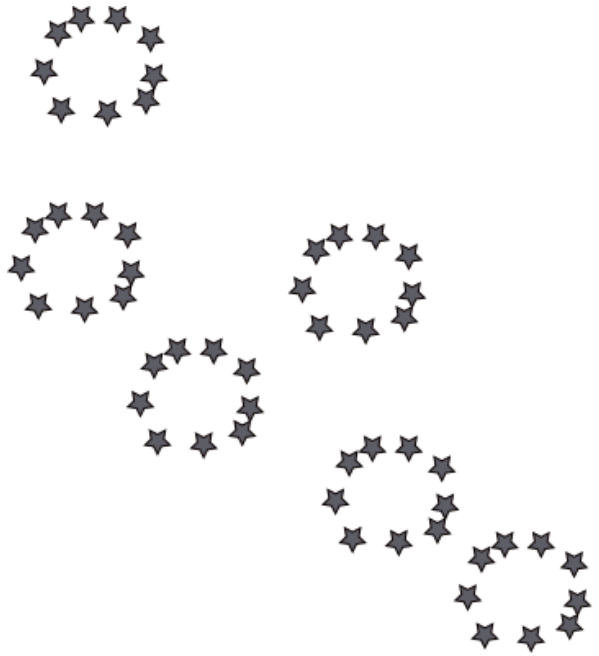
$$p(c|x) = \frac{p(x|c)p(c)}{\sum_c p(x|c)p(c)} = \frac{\int p_l(x|c)p_g(c|x_1,...,x_n)\mathrm{dc}}{\int p_g(x|c)\mathrm{dc}}$$

- We need balance the Request 1 and Request 2, where Request 1 focus on accuracy of the likelihood of the target class, where Request 2 focus on the accuracy when matching the entire test set.
- We meet Request 1 by a local model p_l, and Request 2 by a global model p_g.
- In the enrollment-test framework, p(c) for the target class need to consider the competitive classes, hence should be based on the global model (or, possible a new model?)
- If p(x|c) and p(c) are accurate, of course the best p(x) should use the correct p(x|c) and it falls back to the coupled case. However, the case is rare.

Two perspectives

- Break the dilemma between accuracy and generalization.
  - If we choose a complex p(x|c), even if the conditional is generalizable to test data, p(x) and p(c|x), if computed from p(x|c), will be less generalizable, since they need consider between-class covariance.

- Represent data from two perspectives
  - Use a simple p(x|c) to describe the high-level view
  - Use a complex p(x|c) to describe a low-level view



Employ to PLDA

$$p(c|x) = \frac{p(x|c)p(c)}{\sum_c p(x|c)p(c)} = \frac{\int p_l(x|c)p_g(c|x_1,...,x_n)\mathrm{dc}}{\int p_g(x|c)\mathrm{dc}}$$

- If $p_l(x|c) = p_g(x|c)$ and p(c) are all Gaussian, then we arrive PLDA.
- Now change $p_l(x|c) \neq p_g(x|c)$, we obtain a decoupled PLDA.

Decoupled PLDA

$$p_l(\boldsymbol{x}|\boldsymbol{\mu}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \tilde{\boldsymbol{\Sigma}}).$$

$$p_g(\boldsymbol{\mu}) = \mathcal{N}(\boldsymbol{\mu}; \mathbf{0}, \mathbf{I}\boldsymbol{\epsilon})$$

$$p_g(\boldsymbol{x}|\boldsymbol{\mu}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \sigma\mathbf{I}),$$

$$\frac{\int p_l(x|c)p_g(c|x_1, ..., x_n)\mathrm{dc}}{\int p_g(x|c)\mathrm{dc}}$$

Decoupled PLDA with transform trick

- Using p_g to represent p_l, by employing a simple transform on the data.

$$p_l(\boldsymbol{x}|\boldsymbol{\mu}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \tilde{\boldsymbol{\Sigma}}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \sigma\mathbf{M}^T\mathbf{M}) = p_g(\mathbf{M}\boldsymbol{x}|\boldsymbol{\mu}).$$

Model training

- Global model training as conventional PLDA
- Local model training by optimizing the transform M, so that the transformed data obtain the max likelihood when evaluated by the global model.
- SGD can be used for optimization.

$$\begin{aligned}\mathcal{L}(\mathbf{M}) &= \sum_{k}^{K}\sum_{i=1}^{n_k}\int p_g(\mathbf{M}\boldsymbol{x}_i^k|\boldsymbol{\mu},\sigma\mathbf{I})p_g(\boldsymbol{\mu}|\boldsymbol{x}_1^k,...,\boldsymbol{x}_{n_k}^k)\mathrm{d}\boldsymbol{\mu} \\ &= \sum_{k}^{K}\sum_{i=1}^{n_k}\mathcal{N}(\mathbf{M}\boldsymbol{x}_i^k;\frac{n_k\epsilon}{n_k\boldsymbol{\epsilon}+\sigma}\bar{\boldsymbol{x}}_k,\mathbf{I}(\sigma+\frac{\boldsymbol{\epsilon}\sigma}{n_k\boldsymbol{\epsilon}+\sigma})).\end{aligned}$$

Some results (from Lantian)

TABLE I
PERFORMANCE EER(%) ON HI-MIA DATESET.

	PLDA	CAT	SD/LT
1m-1m	0.620	0.465	0.388
3m-3m	0.891	0.810	0.648
5m-5m	1.135	1.135	0.892

Some results (from Lantian)

TABLE II
PERFORMANCE EER(%) ON AISHELL-1 DATASET

	PLDA	CAT	SD/LT
AN-AN	0.797	0.764	0.373
Mic-Mic	0.778	0.769	0.523
iOS-iOS	0.920	0.845	0.425

Some results (from Lantian)

TABLE III
PERFORMANCE EER(%) ON CSLT-CHRONOS DATASET.

	PLDA	CAT	SD/LT
1st-1st	4.799	4.840	3.907

Some results (from Lantian)

TABLE IV
PERFORMANCE EER(%) ON SITW.EVAL.CORE.

	PLDA	CAT	SD/LT
Eval.Core	6.397	6.151	5.467

Conclusions

- A decoupled scoring is possible. It can be demonstrated that if p(x|c) and p(c) are not accurate, then p(x) and p(c|x) will be not optimal even if coupled p(x|c) is used. In that case, decoupling may result in flexible and better performance.
- By simply allowing an independent Gaussian for the likelihood computation, decoupled PLDA achieved very significant performance improvement.