ECE 1390/2390 Image Processing and Computer Vision – Fall 2021

Generative Classification

Ahmed Dallal

Classification: Generative models

Supervised classification

$f(\boxed{I}) \rightarrow$ label

Given a collection of labeled examples, come up with a function that will predict the labels of new examples.

Training examples

"four"

"nine"

Novel input

?

Kristen Grauman

Supervised classification

How good is the function we come up with to do the classification? (What does "good" mean?)

Depends on:

Need to measure quality of function

- What mistakes does it make
- Cost associated with the mistakes

Kristen Grauman

Supervised classification

Since we know the desired labels of training data, we want to *minimize the expected misclassification*

Supervised classification

Two general strategies

- Use the training data to build representative probability model; separately model class-conditional densities and priors (***Generative***) Bayesian classifier
- Directly construct a good decision boundary, model the posterior (***Discriminative***) SVM, neural nets, logistic regression, etc.

Supervised classification: Generative

Given labeled training examples, predict labels for new examples

- Notation: $(4 \rightarrow 9)$ - object is a '4' but you call it a '9' (Mistake)
- We'll assume the cost of $X \rightarrow X$ is zero. Correct: no penalty

  $4 \rightarrow 4$: No mistake

Kristen Grauman

Supervised classification: Generative

Consider the two-class (binary) decision problem:

- $L(4 \rightarrow 9)$: Loss of classifying a 4 as a 9
- $L(9 \rightarrow 4)$: Loss of classifying a 9 as a 4

Kristen Grauman

Supervised classification: Generative

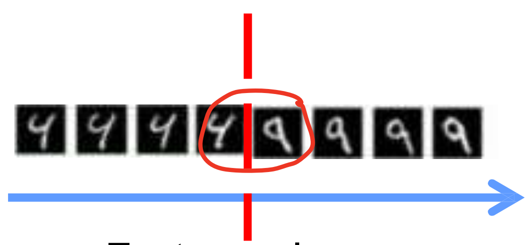
Risk of a classifier strategy ***S*** is expected loss:

$$R(S) = \Pr\left(4 \to 9 \mid \text{using } S\right) L\left(4 \to 9\right) + \Pr\left(9 \to 4 \mid \text{using } S\right) L\left(9 \to 4\right)$$

We want to choose a classifier so as to minimize this total risk

Kristen Grauman

Supervised classification: minimal risk



Feature value x

At best decision boundary, either choice of label yields same expected loss.

50|50

If we choose class "four" at boundary, expected loss is:

$$= P(\text{class is } 9|\mathbf{x})\ L(9 \to 4) + P(\text{class is } 4\ |\mathbf{x}) L(4 \to 4)$$

$$= P(\text{class is } 9|\mathbf{x})\ L(9 \to 4)$$

If we choose class "nine" at boundary, expected loss is:

$$= P(\text{class is } 4|\mathbf{x})\ L(4 \to 9)$$

Kristen Grauman

Supervised classification: minimal risk

- Feature value x

- At best decision boundary, either choice of label yields same expected loss.

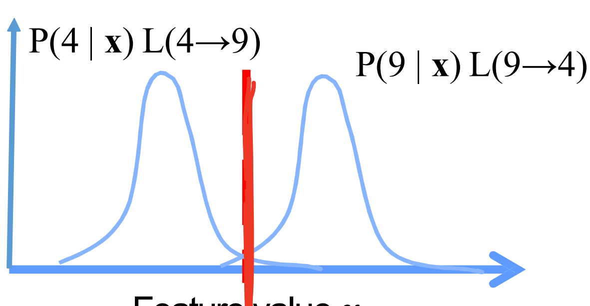
- So, best decision boundary is at point $\mathbf{x}$ where:
  - $P(\text{class is } 9|\mathbf{x})\ L(9 \to 4) = \text{P}(\text{class is } 4|\ \mathbf{x})L(4 \to 9)$
- To classify a new point, choose class with lowest expected loss; i.e., choose

"four" if: $P(4\,|\,\mathbf{x})L(4 \to 9) > P(9\,|\,\mathbf{x})L(9 \to 4)$

Kristen Grauman

Supervised classification: minimal risk

x is feature



Feature value x

At best decision boundary, either choice of label yields same expected loss.

So, best decision boundary is at point $\mathbf{x}$ where:

$$\boxed{\text{P}(\text{class is } 9|\mathbf{x})}\ L(9 \to 4) = \boxed{\text{P}(\text{class is } 4|\ \mathbf{x})}L(4 \to 9)$$

How to evaluate these probabilities?

Kristen Grauman

How risky the decision is

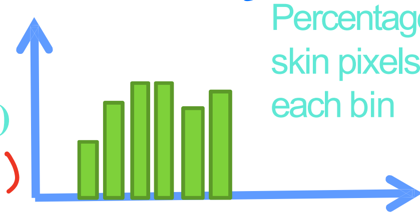
Example: learning skincolors

$P(\text{face} \mid \text{hue}) \underset{\text{not face}}{\overset{\text{face}}{\gtrless}} P(\text{not face} \mid \text{hue})$

Distribution of colors that belong to faces

P(x| skin)

P(hue | face)



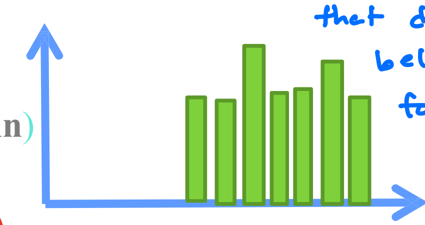
Percentage of skin pixels in each bin

Feature x = Hue

Same for colors that don't belong to faces

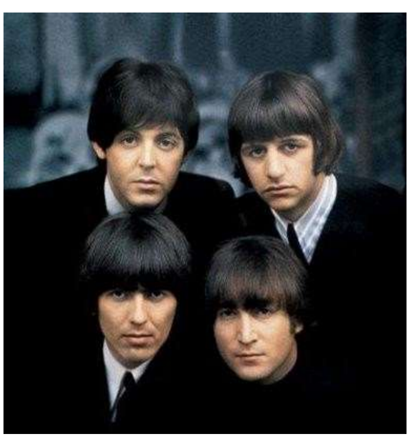
P(x| not skin)

P(hue | not face)



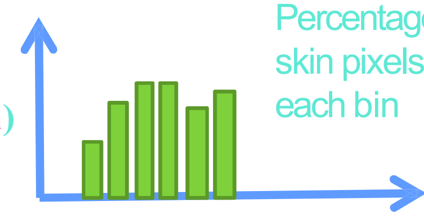
Kristen Grauman

Example: learning skincolors



Now we get a new image, and want to label each pixel as skin or non-skin.

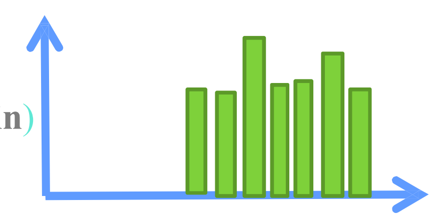
P(x| skin)



Percentage of skin pixels in each bin

Feature x = Hue

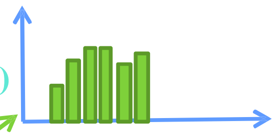
P(x| not skin)



Kristen Grauman

Bayes rule

P(x| skin)



$$\overbrace{P(\text{skin} \mid x)}^{\text{posterior}} = \frac{\overbrace{P(x \mid \text{skin})}^{\text{likelihood}}\;\overbrace{P(\text{skin})}^{\text{prior}}}{P(x)}$$

hue face face

face hue hue

$$P(\text{skin} \mid x) \propto P(x \mid \text{skin})\,P(\text{skin})$$

Where does the prior come from?

Bayes rule in (ab)use

Likelihood ratio test (assuming cost of errors is the same):

If $P(skin|\mathbf{x}) > P(\sim skin|\mathbf{x})$ classify $\mathbf{x}$ as skin

…so ….

Get from distributions (two slides previous)

If $P(\mathbf{x}|skin)P(skin) > P(\mathbf{x}|\sim skin)P(\sim skin)$

classify $\boldsymbol{x}$ as skin (Bayes rule)

(if the costs are different just re-weight)

Bayes rule in (ab)use

Count number of pixels that are a face

…but I don't really know prior $P(skin)$…

…but I can assume it some constant Ω…

…so with some training data I can ***estimate*** Ω …

….and with the same training data I can ***measure likelihood densities*** of ***both*** $P(\boldsymbol{x}|skin)$ and $P(\boldsymbol{x}|{\sim}skin)$…

So….I can more or less come up with a rule…

Steve Seitz

Example: classifying skin pixels

Now for every pixel in a new image, we can estimate probability that it is generated by skin:

If $p(skin \mid x) > \theta$ classify as skin; otherwise not

Brighter pixels are higher probability of being skin

Kristen Grauman

Example: classifying skin pixels

Figure 6: A video image and its flesh probability image

Figure 7: Orientation of the flesh probability distribution marked on the source video image

Gary Bradski, 1998

More general generative models

(feature)

For a given measurement **x** and set of classes c_i choose c^* by:

$$c^* = \arg\max_{c} p(c \mid \mathbf{x}) = \arg\max_{c} p(c)\, p(\mathbf{x} \mid c)$$

More than 2 classes

P(chair | feature) = 0.7

P(person | feature) = 0.02

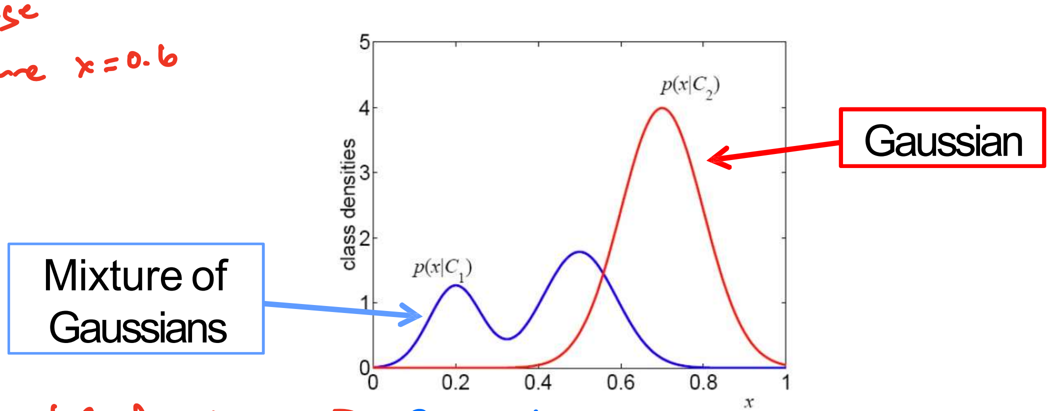
P(whiteboard | feature) = 0.28

Continuous generative models

- If $\mathbf{x}$ is continuous, need *likelihood* density model of $p(\mathbf{x}|c)$
- Typically parametric – Gaussian or mixture of Gaussians

Image

Measure $x = 0.6$



$P(0.6|C_1)\,P(C_1) \gtrless P(0.6|C_2)\,P(C_2)$

Continuous generative models

- Why not just some histogram or some KNN (Parzen window) method?
  - You might…
  - But you would need lots and lots of data everywhere you might get a point
  - The whole point of modeling with a parameterized model is not to need lots of data.

Don't need tons of data

Summary of generative models:

+ Firm probabilistic grounding

+ Allows inclusion of prior knowledge

+ *Parametric modeling of likelihood permits using small number of examples*

+ *New classes do not perturb previous models*

+ Others:

Can take advantage of unlabelled data

Can be used to generate samples

Summary of generative models: $P(C_1)$, $P(C_2)$, etc.

$L(C_1 \rightarrow C_2)$??

- And just where did you get those priors?

- Why are you modeling those obviously non-C points?

- The example hard cases aren't special

- If you have lots of data, doesn't help

Next…

- A really cool way of building a generative model for face recognition (not detection)