Introduction to Machine Learning

Brown University CSCI 1950-F, Spring 2012
Prof. Erik Sudderth

Lecture 5:
Decision Theory & ROC Curves
Gaussian ML Estimation

Many figures courtesy Kevin Murphy's textbook,
Machine Learning: A Probabilistic Perspective

Generative Classifiers

$y \longrightarrow$ class label in *{1,…,C}*, observed in training

$x \in \mathcal{X} \longrightarrow$ observed features to be used for classification

$\theta \longrightarrow$ parameters indexing family of models

$$p(y, x \mid \theta) = p(y \mid \theta) p(x \mid y, \theta)$$

prior distribution *likelihood function*

- Compute class *posterior distribution* via Bayes rule:

$$p(y = c \mid x, \theta) = \frac{p(y = c \mid \theta) p(x \mid y = c, \theta)}{\sum_{c'=1}^{C} p(y = c' \mid \theta) p(x \mid y = c', \theta)}$$

- *Inference:* Find label distribution for some input example
- *Classification:* Make decision based on inferred distribution
- *Learning:* Estimate parameters θ from labeled training data

Decision Theory

$y \in \mathcal{Y} \longrightarrow$ unknown hidden state of "nature"

$x \in \mathcal{X} \longrightarrow$ observed data

$a \in \mathcal{A} \longrightarrow$ set of possible actions we can take

$L(y, a) \longrightarrow$ real-valued loss function: the price we pay if we choose action *a*, and *y* is the true hidden state

- Goal: Choose the action which minimizes the expected loss

$$\delta(\mathbf{x}) = \underset{a \in \mathcal{A}}{\operatorname{argmin}} \, \mathbb{E}\left[L(y, a)\right] \qquad \delta : \mathcal{X} \rightarrow \mathcal{A}$$

  - Some averaging is necessary because we don't know *y*
  - Two notions of expectation: Bayesian versus frequentist
- Some communities speak of maximizing expected utility, which is equivalent if utility equals negative loss

Losses for Classification

$y \in \mathcal{Y}$ ⟶ unknown class or category, finite set of options

$x \in \mathcal{X}$ ⟶ observed data, can take values in any space

$\mathcal{A} = \mathcal{Y}$ ⟶ action is to choose one of the categories

$L(y, a)$ ⟶ table giving loss for all possible mistakes

- Most common default choice is the 0-1 loss:

$$L(y,a) = \mathbb{I}(y \neq a) = \begin{cases} 0 & \text{if } a = y \\ 1 & \text{if } a \neq y \end{cases}$$

- For the special case of binary classification:

predicted label $\hat{y}$	true label y	
	0	1
0	0	λ_{01}
1	λ_{10}	0

Minimizing Expected Loss

$y \in \mathcal{Y} \longrightarrow$ unknown class or category, finite set of options

$x \in \mathcal{X} \longrightarrow$ observed data, can take values in any space

$\mathcal{A} = \mathcal{Y} \longrightarrow$ action is to choose one of the categories

$L(y, a) \longrightarrow$ table giving loss for all possible mistakes

- The *posterior expected loss* of taking action *a* is

$$\rho(a|\mathbf{x}) \triangleq \mathbb{E}_{p(y|\mathbf{x})}\left[L(y,a)\right] = \sum_{y} L(y,a)p(y|\mathbf{x})$$

- The optimal *Bayes decision rule* is then

$$\delta(\mathbf{x}) = \arg\min_{a \in \mathcal{A}} \rho(\mathbf{a}|\mathbf{x})$$

- Bayesian classification requires *both* model and loss

Minimizing Probability of Error

$$L(y,a) = \mathbb{I}(y \neq a) = \begin{cases} 0 & \text{if } a = y \\ 1 & \text{if } a \neq y \end{cases}$$

- The *posterior expected loss* of taking action *a* is

$$\rho(a|\mathbf{x}) \triangleq \mathbb{E}_{p(y|\mathbf{x})}\left[L(y,a)\right] = \sum_y L(y,a)p(y|\mathbf{x})$$

$$\rho(a \mid x) = p(a \neq y \mid x) = 1 - p(a = y \mid x)$$

- Optimal decision is the *maximum a posteriori (MAP)* estimate:

$$\hat{y}(x) = \arg\max_{y \in \mathcal{Y}} p(y \mid x)$$

- If classes are equally likely *a priori*, this becomes

$$\hat{y}(x) = \arg\max_{y \in \mathcal{Y}} p(x \mid y) \quad \text{if} \quad p(y) = \frac{1}{C}$$

Binary Classification in General

	$\hat{y}=1$	$\hat{y}=0$
$y=1$	0	L_{FN}
$y=0$	L_{FP}	0

Loss Function

$$p(y=1) = \pi$$
$$p(y=0) = 1-\pi$$

Prior Distribution

$$p(x \mid y=1)$$
$$p(x \mid y=0)$$

Likelihood

- *False positive (FP):* Predict class 1 when truth is class 0
- *False negative (FN):* Predict class 0 when truth is class 1

$$\rho(\hat{y}=0|\mathbf{x}) = L_{FN}\, p(y=1|\mathbf{x})$$
$$\rho(\hat{y}=1|\mathbf{x}) = L_{FP}\, p(y=0|\mathbf{x})$$

- When should the optimal classifier choose class 1?

$$\frac{p(y=1 \mid x)}{p(y=0 \mid x)} > \frac{L_{FP}}{L_{FN}} \qquad \frac{p(x \mid y=1)}{p(x \mid y=0)} > \frac{L_{FP}}{L_{FN}} \cdot \frac{1-\pi}{\pi}$$

- Optimal decision rule is always a *likelihood ratio test*

False Positives vs. False Negatives

- *False positive (FP):* Predict class 1 when truth is class 0
- *False negative (FN):* Predict class 0 when truth is class 1
- *True positive (TP):* Predict class 1 when truth is class 1
- *True negative (TN):* Predict class 0 when truth is class 0

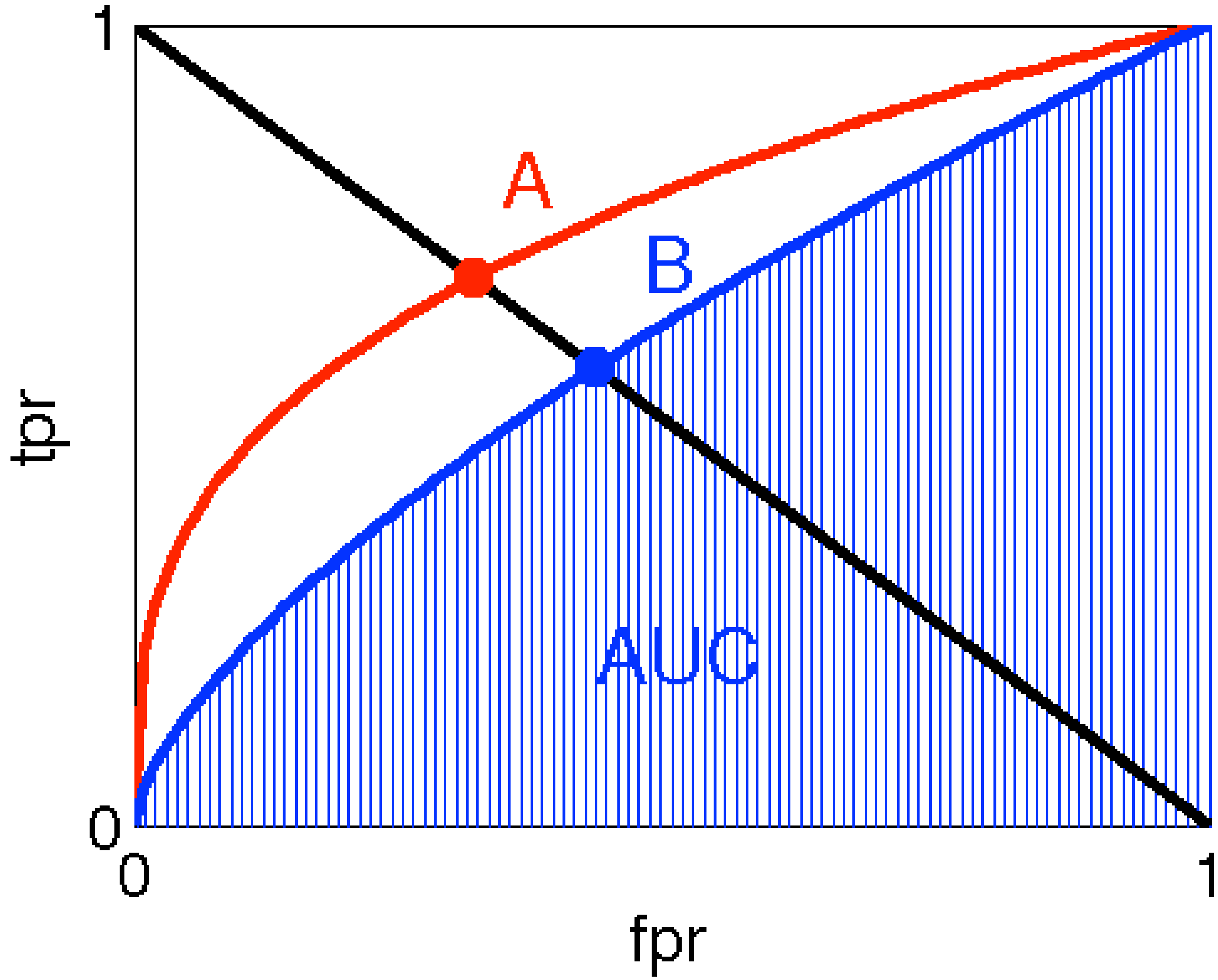
		Truth		
		1	0	Σ
Estimate	1	TP	FP	$\hat{N}_+ = TP + FP$
	0	FN	TN	$\hat{N}_- = FN + TN$
	Σ	$N_+ = TP + FN$	$N_- = FP + TN$	$N = TP + FP + FN + TN$

- *Sensitivity, recall, or true positive rate (TPR)*
- *False alarm rate or false positive rate (FPR)*

$$TPR = \frac{TP}{N_+} \approx p(\hat{y} = 1 \mid y = 1) \qquad FPR = \frac{FP}{N_-} \approx p(\hat{y} = 1 \mid y = 0)$$

- *Receiver operating characteristic (ROC):* Plot of TPR vs FPR

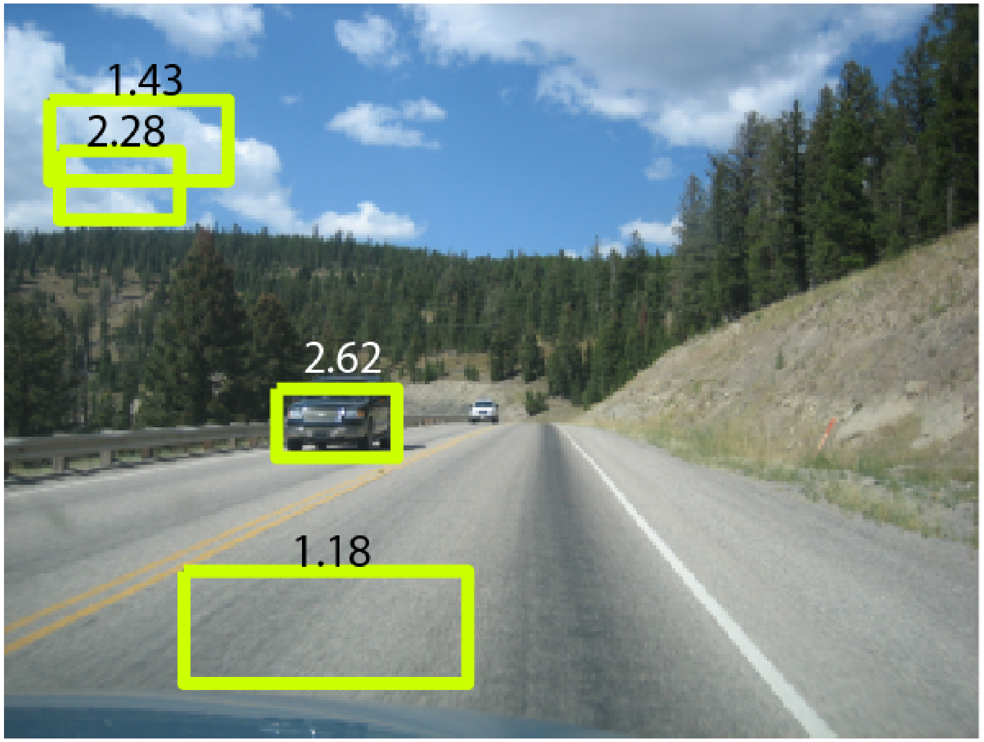
Idealized ROC Curves

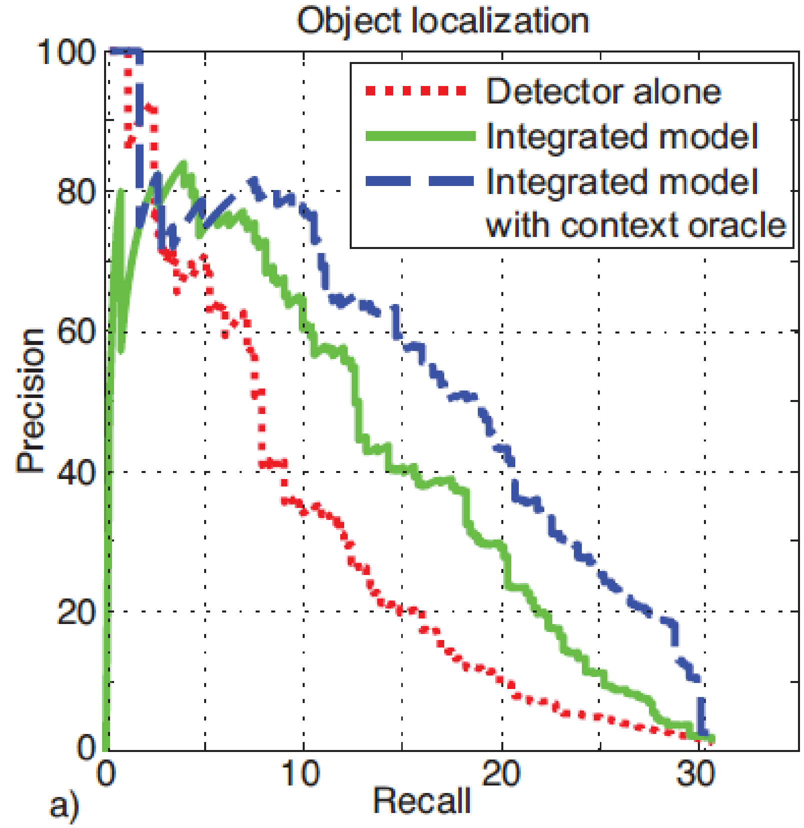


$$\log \frac{p(x \mid y=1)}{p(x \mid y=0)} > \tau$$

EER: Equal Error Rate
AUC: Area Under Curve

Example: Object Detection



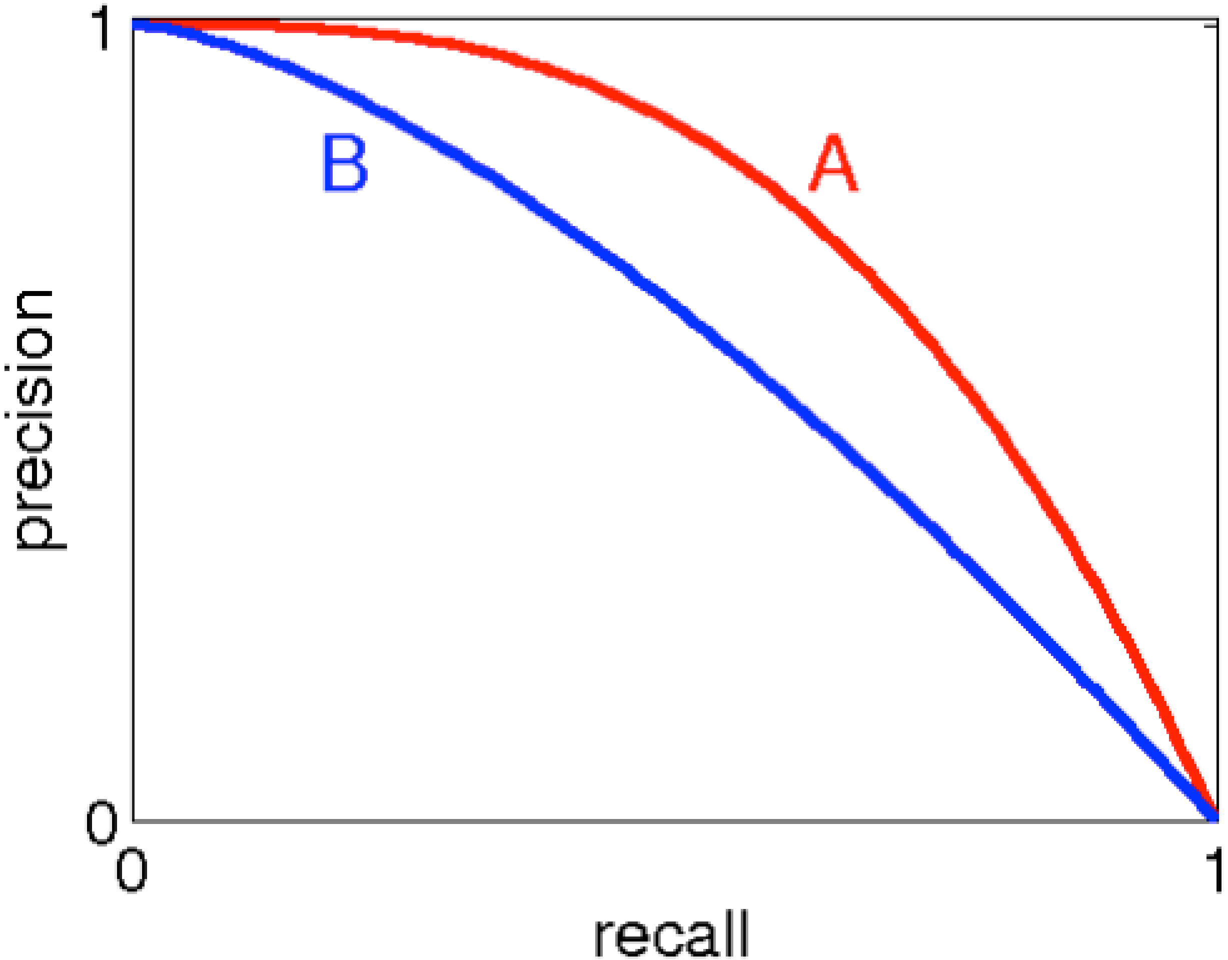


Fei-Fei, Fergus, Torralba, ICCV 2009

The number of *negative* examples may not be well defined:

- How many windows not containing a car are there in an image?
- How many documents not about cars exist in the world?

Idealized Precision-Recall Curves



Recall:

$$\frac{TP}{N_+} \approx p(\hat{y} = 1 \mid y = 1)$$

Precision:

$$\frac{TP}{\hat{N}_+} \approx p(y = 1 \mid \hat{y} = 1)$$

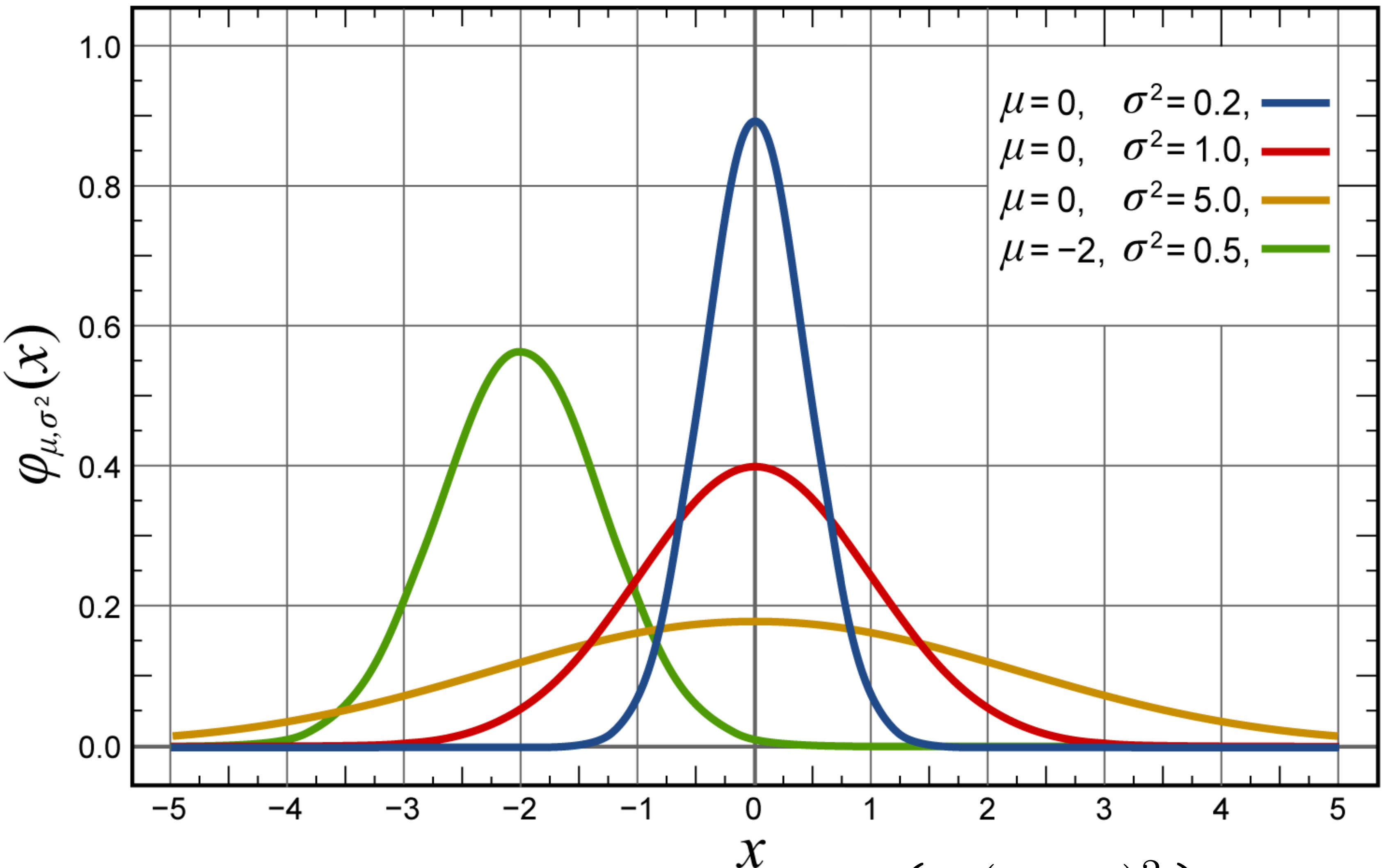
Learning with Generative Models

- *Features:* Encode raw data in some numeric format
- *Models:* Choose families of distributions relating all of the variables of interest (features, labels, etc.).
  Must make sure sample space matches feature format!
- *Learning:* Estimate specific model parameters given training data, using maximum likelihood, Bayesian estimation, …
- *Inference:* For a test example, determine distribution of latent or hidden variables given observed features
- *Decisions:* Use inferred distribution to minimize expected loss

From Discrete to Continuous Random Variables

- What if my observed features are real-valued quantities, not discrete counts?
- What if I want to estimate hidden real-valued quantities, not discrete class labels?

Gaussian ML Estimation



$$\mathcal{N}(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}$$