Lecture 3

STAT161/261 Introduction to Pattern Recognition and Machine Learning

Spring 2018

Prof. Allie Fletcher

Previous lectures

- What is machine learning?
  - Objectives of machine learning
  - Supervised and Unsupervised learning
    - Examples and approaches
- Multivariate Linear regression
  - Predicts any continuous valued target from vector of features
  - Important:
    - Simple to compute, parameters easy to interpret
  - Illustrate basic procedure: Model formulation, loss function, …
  - Many natural phenomena have a linear relationship
- Subsequent lectures build up theory behind such parametric estimation techniques

Outline

- Principles of Supervised Learning
  - Model Selection and Generalization (Alpaydin 2.7 & 2.8, Bishop 1.3)
  - Overfitting and Underfitting
- Decision Theory (1.5 Bishop and Ch3 Alpaydin)
  - Binary Classification
  - Maximum Likelihood and Log likelihood
  - Bayes Methods: MAP and Bayes Risk
  - Receiver operating characteristic
  - Minimum probability of error
- Issues in applying Bayesian classification
- Curse of Dimensionality

Outline

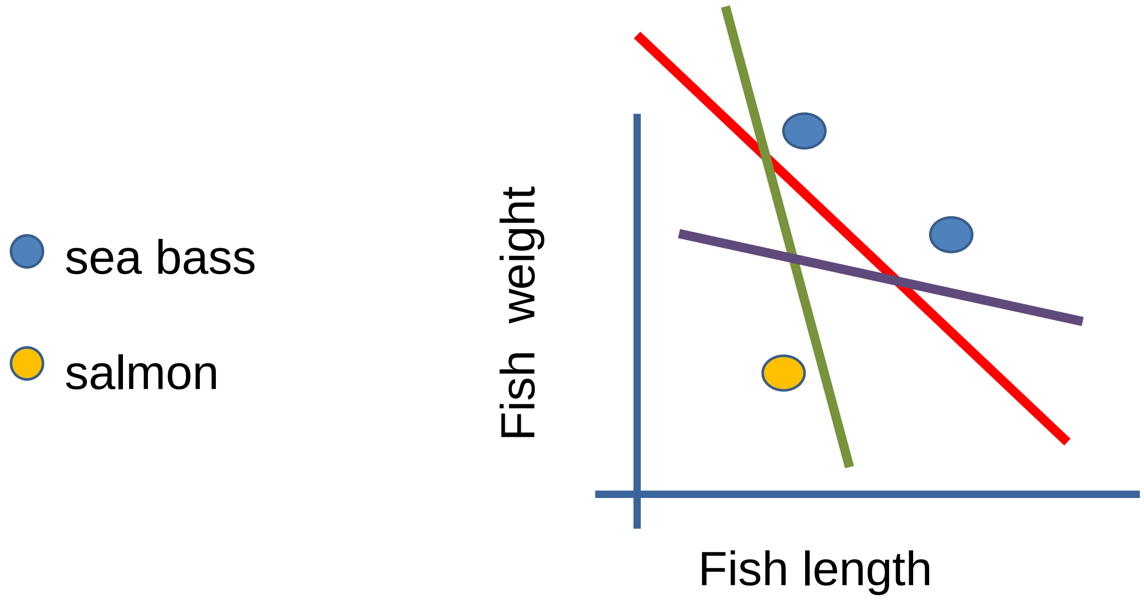
- Principles of Supervised Learning
  - Model Selection and Generalization
  - Overfitting and Underfitting
- Decision Theory
  - Binary Classification
  - Maximum Likelihood and Log likelihood
  - Bayes Methods: MAP and Bayes Risk
  - Receiver operating characteristic
  - Minimum probability of error
- Issues in applying Bayesian classification
- Curse of Dimensionality

Memorization vs. Generalization

- Two key concepts in ML:
  - Memorization: Finding an algorithm that fits training data well
  - Generalization: Gives good results on data not yet seen. Prediction.
- Example: Suppose we have only three samples of fish
  - Can we learn a classification rule? Sure

Memorization vs. Generalization

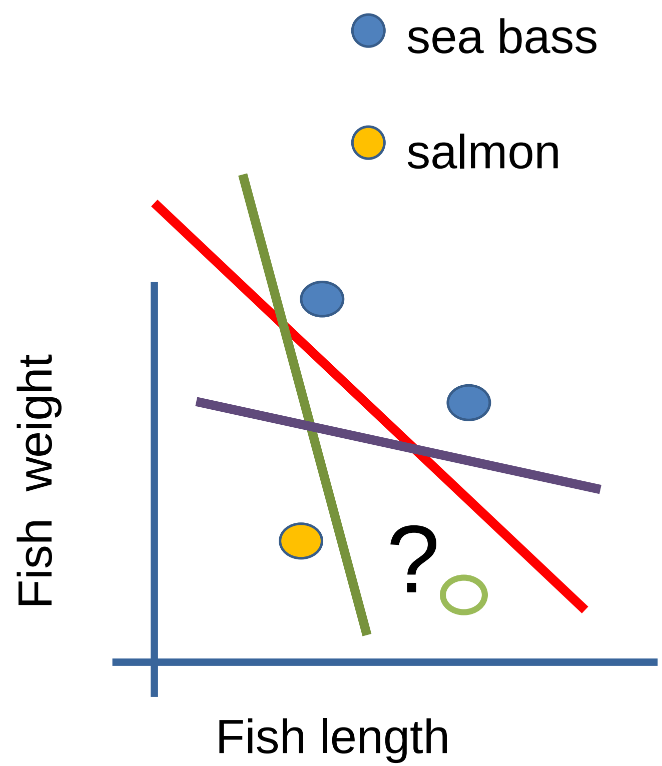
- Many possible classifier fit training data
  - Easy to memorize the data set, but need to generalize to new data
  - All three classifiers below (Classifier 1, C2, and C3) fit data
- But, which one will predict new sample correctly?



Memorization vs. Generalization

- Which classifier predicts new sample correctly?
  - Classifier 1 predicts salmon
  - Classifier 2 predicts salmon
  - Classifier 3 predicts sea bass

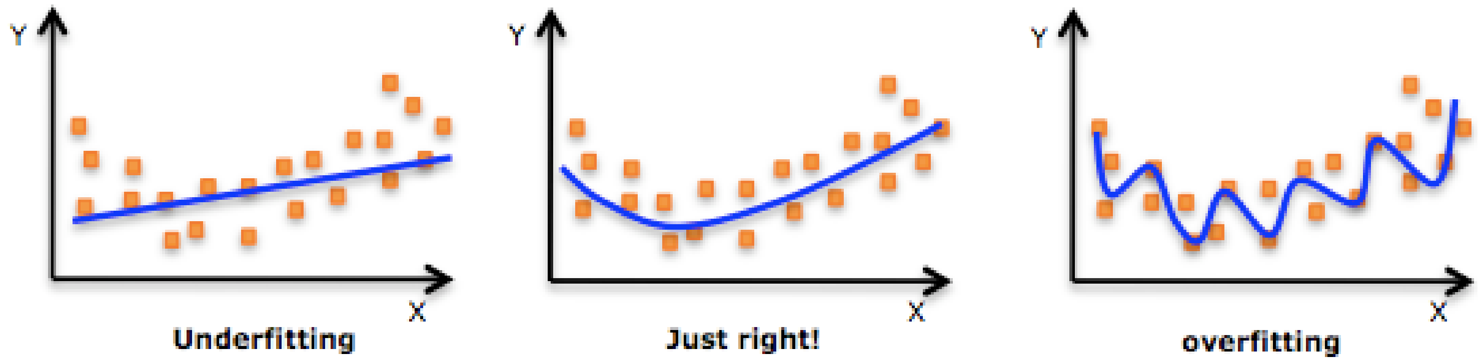
- We do not know which one is right:
  - Not enough training data
  - Need more samples to generalize



Basic Tradeoff

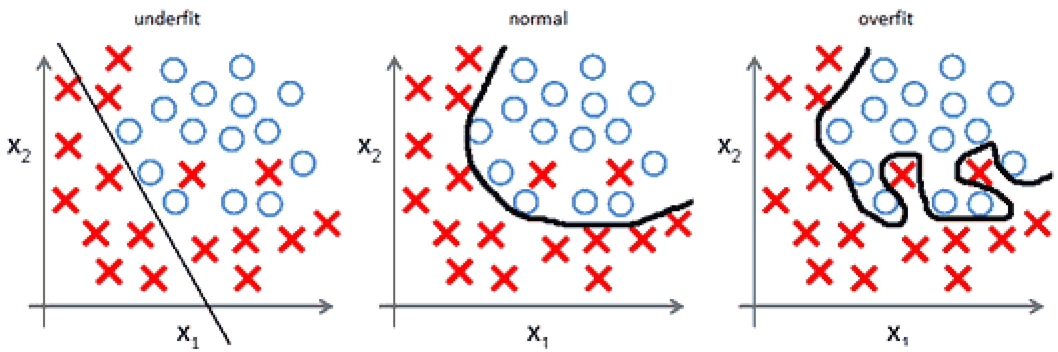
- Generalization requires assumptions
- ML uses a model
- Basic tradeoff between three factors:
  - Model complexity: Allows to fit complex relationships
  - Amount of training data
  - Generalization error: How model fits new samples
- This class: Provides a principled ways to:
  - Formulate models that can capture complex behavior
  - Analyze how well they perform under statistical assumptions

Generalization: Underfitting and Overfitting



- Example: Consider fitting a polynomial
- Assume a low-order polynomial
  - Easy to train. Less parameters to estimate
  - But model does not capture full relation. Underfitting
- Assume too high a polynomial
  - Fits complex behavior
  - But, sensitive to noise. Needs many samples. Overfitting
- This course:
  - How to rigorously quantify model selection and algorithm performance

Generalization: Underfitting and Overfitting



- Example: Consider fitting a polynomial
- Assume a low-order polynomial
  - Easy to train. Less parameters to estimate
  - But model does not capture full relation. Underfitting
- Assume too high a polynomial
  - Fits complex behavior
  - But, sensitive to noise. Needs many samples. Overfitting
- This course:
  - How to rigorously quantify model selection and algorithm performance

Ingredients in Supervised Learning

- Select a model: $\hat{y} = g(x, \theta)$
  - Describes how we predict target y from features x
  - Has parameters θ
- Get training data: $(x_i, y_i), i = 1, \ldots, n$
- Select a loss function $L(y_i, \hat{y}_i)$
  - How well prediction matches true value on the training data
- Design algorithm to try to minimize loss:

$$\hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{n} L(y_i, \hat{y}_i)$$

- The art principled methods to develop models and algorithms for often intractable loss functions and complex large is what machine learning is really all about.

Outline

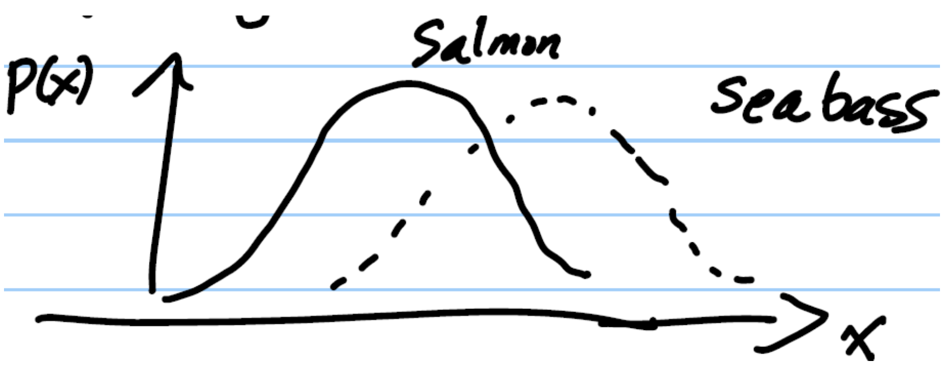
- Principles of Supervised Learning
  - Model Selection and Generalization
  - Overfitting and Underfitting
- Decision Theory
  - Binary Classification
  - Maximum Likelihood and Log likelihood
  - Bayes Methods: MAP and Bayes Risk
  - Receiver operating characteristic
  - Minimum probability of error
- Issues in applying Bayesian classification
- Curse of Dimensionality

Decision Theory

- How to make decision in the presence of uncertainty?
- History: Prominent in WWII:
  radar for detecting aircraft, codebreaking, decryption
- Observed data x ∈ X, state y ∈ Y
- p(x | y): conditional distribution
  Model of how the data is generated
- Example: y ∈ {0, 1} (salmon vs. sea bass) or (airplane vs. bird, etc.)
  x: length of fish

$$p(x|y) = \frac{1}{\sqrt{2\pi}\sigma_y}\exp\left(-\frac{(x-\mu_y)^2}{2\sigma_y^2}\right)$$

- μ_y: mean, σ_y^2: variance



Maximum Likelihood (ML) Decision

- Which fish type is more likely to given the observed fish length x?

If $p(x \mid y = 0) > p(x \mid y = 1)$,
guess salmon;
otherwise classify the fish as sea bass

  - If $\frac{p(x \mid y=1)}{p(x \mid y=0)} > 1$, guess sea bass [*likelihood ratio or LRT*]
  - equivalently: if $\log \frac{p(x \mid y=1)}{p(x \mid y=0)} > 0$ [*log-likelihood ratio*]
  - $\hat{y}_{\text{ML}} = \alpha(x) = Se \arg\max_y p(x \mid y)$

- Seems reasonable, but what if salmon may be much more likely than sea bass?

Maximum a Posteriori (MAP) Decision

- Introduce prior probabilities $p(y=0)$ and $p(y=1)$
  - Salmon more likely than sea bass: $p(y=0) > p(y=1)$
- Now, which type of fish is more likely given observed fish length?
- Bayes' Rule: $p(y \mid x) = \dfrac{p(x \mid y)p(y)}{p(x)}$
- Including prior probabilities:

If $p(y=0 \mid x) > p(y=1 \mid x)$, guess salmon; otherwise, pick sea bass

$$\hat{y}_{\mathrm{MAP}} = \alpha(x) = \arg\max_y p(y \mid x) = \arg\max_y p(x \mid y)\; p(y)$$

UCLA

Making it more interesting, full on Bayes

- What does it cost for a mistake? Plane with a missile, not a big bird?
- Define loss or cost:

  $L(\alpha(x), y)$: cost of decision $\alpha(x)$ when state is y

  also often denoted C_{ij}

	Y = 0	Y = 1
$\alpha(x) = 0$	Correct, cost L(0,0)	Incorrect, cost L(0,1)
$\alpha(x) = 1$	incorrect, cost L(1,0)	Correct, cost L(1,1)

- Classic: Pascal's wager

	God exists (G)	God does not exist (¬G)
Belief (B)	$+\infty$ (infinite gain)	−1 (finite loss)
Disbelief (¬B)	$-\infty$ (infinite loss)	+1 (finite gain)

Risk Minimization

- So now we have: the likelihood functions p(x | y)

  priors p(y)

  decision rule $\alpha(x)$

  loss function $L(\alpha(x), y)$:
- *Risk* is expected loss:

$$E[L] = L(0,0)\, p(\alpha(x) = 0, y = 0) + L(0,1)\, p(\alpha(x) = 0, y = 1) + L(1,0)\, p(\alpha(x) = 1, y = 0) + L(1,1)\, p(\alpha(x) = 1, y = 1)$$

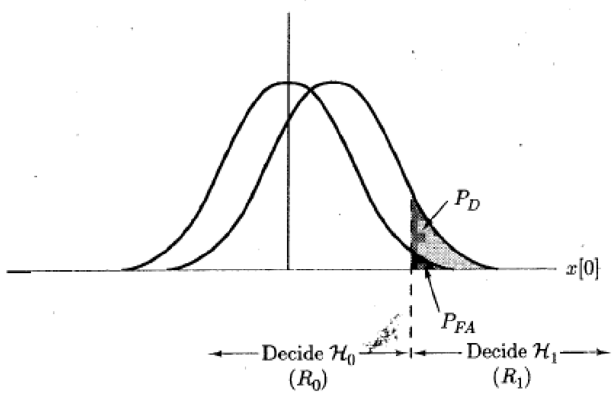
- Without loss of generality, zero cost for correct decisions

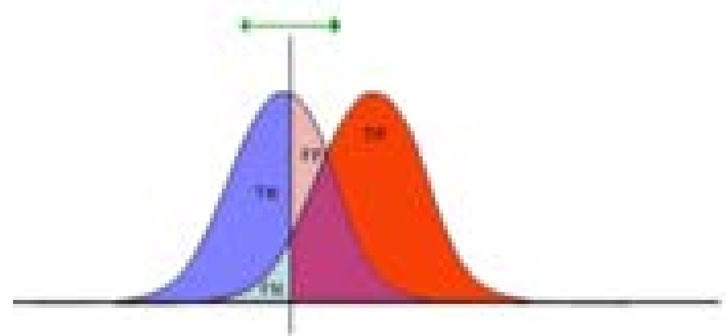
$$E[L] = L(1,0)\, p(\alpha(x) = 1 \mid y = 0) p(y = 0) + L(0,1)\, p(\alpha(x) = 0 \mid y = 1) p(y = 1)$$

- Bayes Decision Theory says "pick decision rule $\alpha(x)$ to minimize risk"

Visualizing Errors

- Type I error (False alarm or False Positive): Decide H1 when H0
- Type II error (Missed detection or False Negative): Decide H0 when H1
- Trade off
- Can work out error probabilities from conditional probabilities





TP	FP
FN	TN
1	1

Often more formally written Hypothesis Testing

- Two possible hypotheses for data
  - H0: Null hypothesis, H1: Alternate hypothesis
- Model statistically:
  - $p(x|H_i), i = 0,1$
  - Assume some distribution for each hypothesis
- Given
  - Likelihood $p(x|H_i), i = 0,1$, Prior probabilities $p_i = P(H_i)$
- Compute posterior $P(H_i|x)$
  - How likely is H_i given the data and prior knowledge?
- Bayes' Rule:

$$P(H_i|x) = \frac{p(x|H_i)p_i}{p(x)} = \frac{p(x|H_i)p_i}{p(x|H_0)p_0 + p(x|H_1)p_1}$$

MAP: Minimum Probability of Error

- Probability of error:

$$P_{err} = P\left(\widehat{H} \neq H\right)$$
$$= P\left(\widehat{H} = 0 \middle| H_1\right)p_1 + P\left(\widehat{H} = 1 \middle| H_0\right)p_0$$

- Write with integral:

$$P\left(\widehat{H} \neq H\right) = \int p(x)P\left(\widehat{H} \neq H \middle| x\right)dx$$

- Error is minimized with MAP estimator

$$\widehat{H} = 1 \Leftrightarrow P(H_1|x) \geq P(H_0|x)$$

- Use Bayes rule:

$$\widehat{H} = 1 \Leftrightarrow P(x|H_1)p_1 \geq P(x|H_0)p_0$$

- Equivalent to an LRT with $\gamma = p_0/p_1$
- Probabilistic interpretation of threshold

Bayes Risk Minimization

- As before, express risk as integration over x:

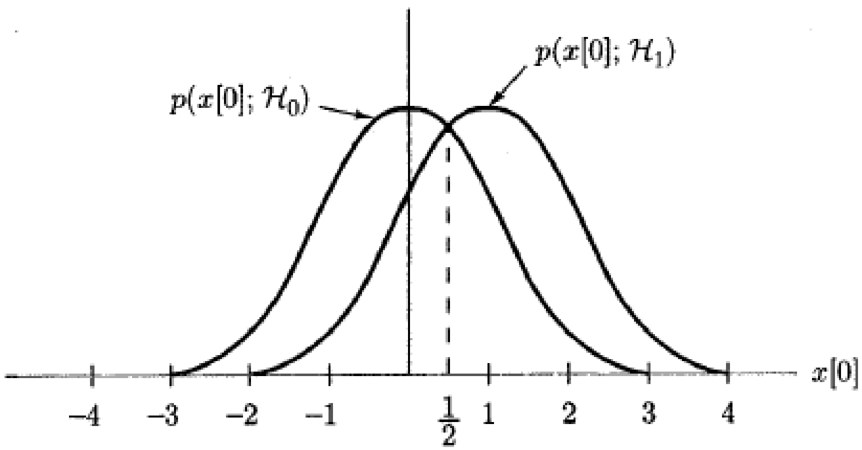
$$R = \int \sum_{ij} C_{ij} \, P(H_j|x) 1_{\{\hat{H}(x)=i\}} \, p(x) dx$$

- To minimize, select $\hat{H}(x) = 1$ when
  - $C_{10}P(H_0|x) + C_{11}P(H_1|x) \le C_{00}P(H_0|x) + C_{01}P(H_1|x)$
  - $P(H_1|x)/P(H_0|x) \ge (C_{10} - C_{00})/(C_{11} - C_{01})$
- By Bayes Theorem, equivalent to an LRT with

$$\frac{P(x|H_1)}{P(x|H_0)} \ge \frac{(C_{10} - C_{00})p_0}{(C_{11} - C_{01})p_1}$$

Same example basically, but posed as additive noise

- Scalar Gaussian
  - H_0: $x = w, \quad w \sim N(0, \sigma^2)$
  - H_1: $x = A + w, \quad w \sim N(0, \sigma^2)$



$A = 1$

- Example: A medical test for some disease
  - x = measured value of the patient
  - H_0 = patient is fine, H_1 = patient is ill
  - Probability model: x is elevated with the disease

Example : Scalar Gaussians

- Hypothesis:
  - H_0: $x = w, \quad w \sim N(0, \sigma^2)$
  - H_1: $x = A + w, \quad w \sim N(0, \sigma^2)$
- Problem: Use the LRT test to define a classifier and compute P_D, P_{FA}
- Step 1. Write the probability distributions:
  - $p(x|H_0) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}, p(x|H_1) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-A)^2}{2\sigma^2}}$

Scalar Guassian continued

- Step 2. Write the log likelihood:

$$L(x) = \ln \frac{p(x|H_1)}{p(x|H_0)} = \frac{1}{2\sigma^2}\left(x^2 - (x-A)^2\right)$$

$$= \frac{1}{2\sigma^2}\left(2Ax + A^2\right)$$

- Step 3.
  - $L(x) \geq \gamma \Rightarrow x \geq t = (2\sigma^2\gamma - A^2)/2A$
  - Write all further answers in terms of t instead of γ
  - Classifier:

$$\hat{H} = \begin{cases} 1 & x \geq t \\ 0 & x < t \end{cases}$$

Scalar Guassian (cont)

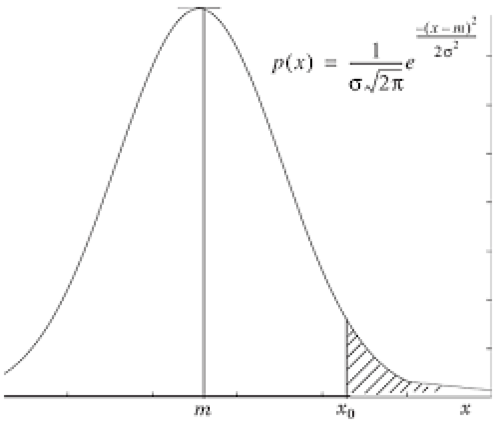
- Step 4. Compute error probabilities
  - $P_D = P\left(\widehat{H} = 1 \middle| H_1\right) = P(x \geq t | H_1)$
  - Under $H_1, x \sim N(A, \sigma^2)$
  - So, $P_D = P(x \geq t | H_1) = Q(\frac{t-A}{\sigma})$
  - Similarly, $P_D = P(x \geq t | H_0) = Q(\frac{t}{\sigma})$
- Here, $Q(z) =$ Marcum Q-function
  - $Q(z) = P(Z \geq z),\ Z \sim N(0,1)$

UCLA

Review: Gaussian Q-Function

- Problem: Suppose $X \sim N(\mu, \sigma^2)$.
  - Often must compute probabilities like $P(X \geq t)$
  - No closed-form expression.
- Define Marcum Q-function:
  $Q(z) = P(Z \geq z),\ Z \sim N(0,1)$
- Let $Z = (X - \mu)/\sigma$
- Then

$$P(X \geq t) = P\left(Z \geq \frac{t-\mu}{\sigma}\right) = Q\left(\frac{t-\mu}{\sigma}\right)$$



Example: Two Exponentials

- Hypothesis:
  - H_i: $p(x|H_i) = \lambda_i e^{-\lambda_i x}$, $i = 0, 1$ Assume $\lambda_0 > \lambda_1$
  - Find ML detector threshold and probability of false alarm…

- Step 1. Write the conditional probability distributions
  - Nothing to do. Already given.
- Step 2: Log likelihood:
  - $L(x) = \ln \frac{p(x|H_1)}{p(x|H_0)} = (\lambda_0 - \lambda_1)x + \ln \frac{\lambda_1}{\lambda_0}$
  - ML: LRT test pick H1 if x $\geq (1/\lambda_0 - \lambda_1) \ln \frac{\lambda_0}{\lambda_1}$

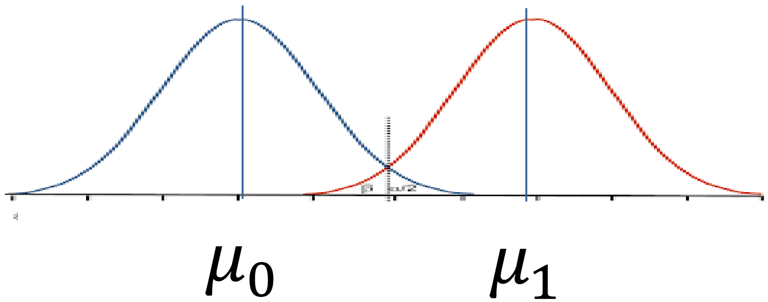
  - $L(x) \geq \gamma \Rightarrow x \geq t$

Two Exponentials (continued)

- Compute error probabilities
  - $P_D = P(\hat{H} = 1 | H_1) = P(x \geq t | H_1)$
  - $P_D = \int_t^\infty p(x|H_1)dx = \int_t^\infty \lambda_1 e^{-\lambda_1 x} dx = e^{-\lambda_1 t}$
  - Similarly, $P_{FA} = e^{-\lambda_0 t}$

MAP Example

- Hypotheses: H_i: $x = N(\mu_i, \sigma^2)$, $p_i = P(H_i)$, $i = 0,1$
- Two Gaussian densities with different means
  - But same variance
- Problem: Find the MAP estimate



- Solution: First, write densities

$$p(x|H_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu_i)^2}{2\sigma^2}\right)$$

- MAP estimate: Select

$$\hat{H} = 1 \Leftrightarrow p(x|H_1)p_1 \geq p(x|H_0)p_0$$

- In log domain:

$$-\frac{(x-\mu_1)^2}{2\sigma^2} + \ln p_1 \geq -\frac{(x-\mu_0)^2}{2\sigma^2} + \ln p_0$$

MAP Example: (Cont)

- More simplifications : $\widehat{H} = 1$ when

$$-\frac{(x-\mu_1)^2}{2\sigma^2} + \ln p_1 \geq -\frac{(x-\mu_0)^2}{2\sigma^2} + \ln p_0$$

$$\Leftrightarrow (x-\mu_0)^2 - (x-\mu_1)^2 \leq 2\sigma^2 \ln \frac{p_0}{p_1}$$

$$\Leftrightarrow 2(\mu_1-\mu_0)x + \mu_1^2 - \mu_0^2 \geq 2\sigma^2 \ln \frac{p_0}{p_1}$$

$$\Leftrightarrow x \geq \frac{\mu_1+\mu_0}{2} + \frac{\sigma^2}{\mu_1-\mu_0} \ln \frac{p_0}{p_1}$$

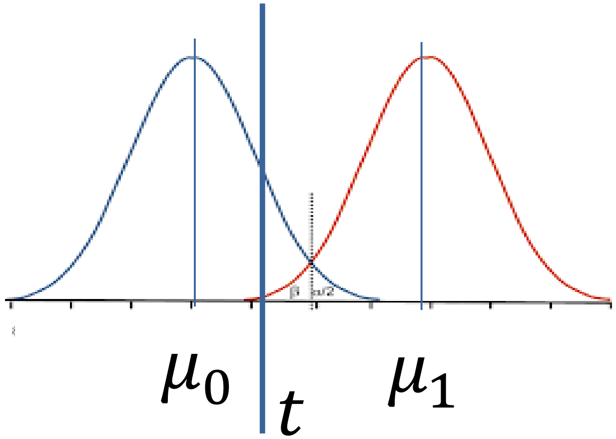
MAP Example (cont)

- MAP estimator: $\widehat{H} = 1$ when $x \geq t$
- Threshold

$$t = \frac{\mu_1 + \mu_0}{2} + \frac{\sigma^2}{\mu_1 - \mu_0} \ln \frac{p_0}{p_1}$$

Midpoint between Gaussians

Shifts to the left when $p_0 \leq p_1$



Multiple Classes

- Often have multiple classes. $y = 1, \ldots, K$
- Most methods easily extend:
  - ML: Take max of K likelihoods:

$$\hat{y} = \arg\max_{i=1,\ldots,K} p(x|y = i)$$

  - MAP: Take max of K posteriors:
  - LRT: Take max of K weighted likelihoods:

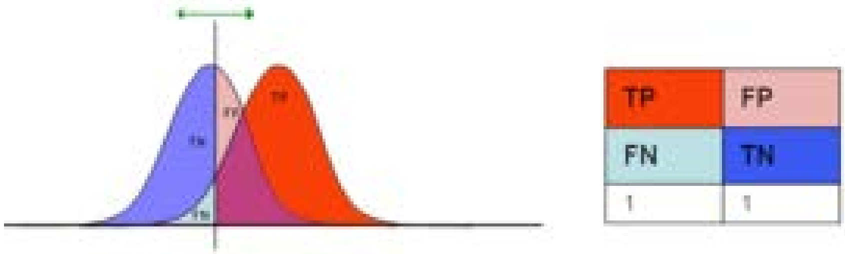
$$\hat{y} = \arg\max_{i=1,\ldots,K} p(x|y = i)\,\gamma_i$$

Outline

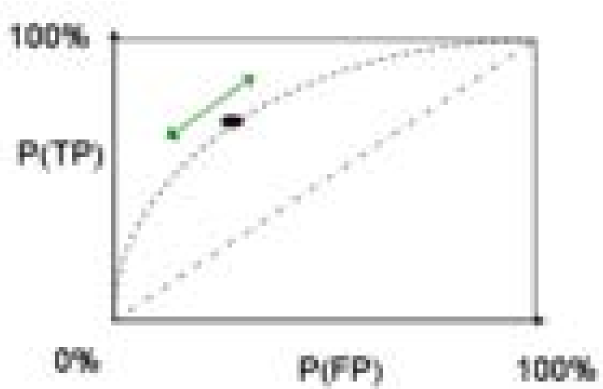
- Principles of Supervised Learning
  - Model Selection and Generalization
  - Overfitting and Underfitting
- Decision Theory
  - Binary Classification
  - Maximum Likelihood and Log likelihood
  - Bayes Methods: MAP and Bayes Risk
  - Receiver operating characteristic
  - Minimum probability of error
- Issues in applying Bayesian classification
- Curse of Dimensionality

ROC curves : error tradeoffs

- Any binary decision strategy has a trade-off in errors
- Reminder of Errors
  - TP = true positive
  - TN = true negative
  - FP = false positive
  - FN = false negative

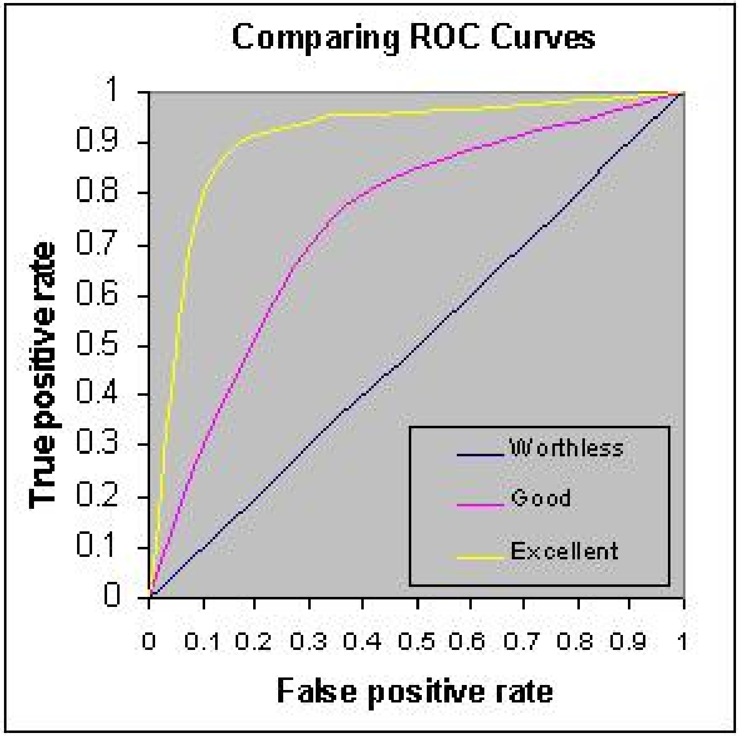


- Typical illustrate: Tradeoff between TP and FP
- Receiver Operating Characteristic



ROC Curve

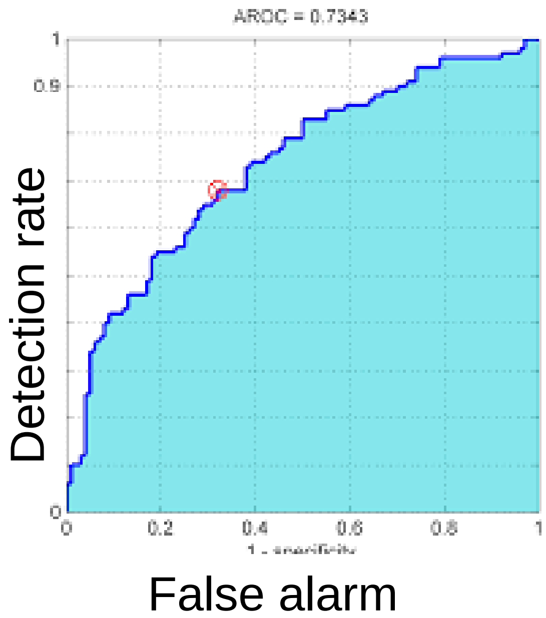
- P_D vs. P_{FA}
- Trace out: $\left(P_{FA}(\gamma), P_D(\gamma)\right)$
- Shows tradeoff
- Random guessing:
  - Select H_1 randomly α per cent of time
  - $P_D = \alpha,\ P_{FA} = \alpha \Rightarrow P_D = P_{FA}$



Area Under The Curve (AUC)

- Simple measure of quality
- $AUC =$ average of $P_D(\gamma)$ with $x = \gamma$ under H_0
- Proof:

$$AUC = \int P_D(\gamma) P'_{FA}(\gamma) d\gamma = \int P_D(\gamma) p(\gamma | H_0) d\gamma$$



ROC Example: Two Exponentials

- Hypotheses:
  - H_i: $p(x|H_i) = \lambda_i e^{-\lambda_i x}, \; i = 0, 1$
- From before, LRT test is $\hat{H} = \begin{cases} 1 & x \geq t \\ 0 & x < t \end{cases}$
- Error probabilities: $P_D = e^{-\lambda_1 t}, P_{FA} = e^{-\lambda_0 t}$
- ROC curve:
  - Write P_D in terms of P_{FA}
  - $t = -\frac{1}{\lambda_0} \ln P_{FA} \Rightarrow P_D = P_{FA}^{\lambda_1/\lambda_0}$

Outline

- Principles of Supervised Learning
  - Model Selection and Generalization
  - Overfitting and Underfitting
- Decision Theory
  - Binary Classification
  - Maximum Likelihood and Log likelihood
  - Bayes Methods: MAP and Bayes Risk
  - Receiver operating characteristic
  - Minimum probability of error
- Issues in applying Bayesian classification
- Curse of Dimensionality

Problems in Using Hypothesis Testing

- Hypothesis testing formulation requires
  - Knowledge of likelihood $p(x|H_i)$
  - Possibly knowledge of prior $P(H_i)$
- Where do we get these?
- Approach 1:
  - Learn distributions from data
  - Then apply hypothesis testing
- Approach 2:
  - Use hypothesis testing to select a form for the classifier
  - Learn parameters of the classifier directly from data

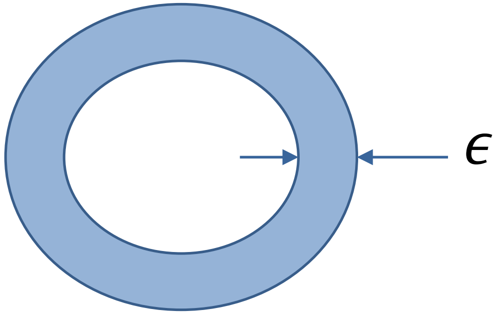
Outline

- Principles of Supervised Learning
  - Model Selection and Generalization
  - Overfitting and Underfitting
- Decision Theory
  - Binary Classification
  - Maximum Likelihood and Log likelihood
  - Bayes Methods: MAP and Bayes Risk
  - Receiver operating characteristic
  - Minimum probability of error
- Issues in applying Bayesian classification
- Curse of Dimensionality

Intuition in High-Dimensions

- Examples of Bayes Decision theory can be misleading because they are given in low dimensional spaces, 1 or 2 dim
  - Most ML problems today have high dimension
  - Often our geometric intuition in high-dimensions is wrong

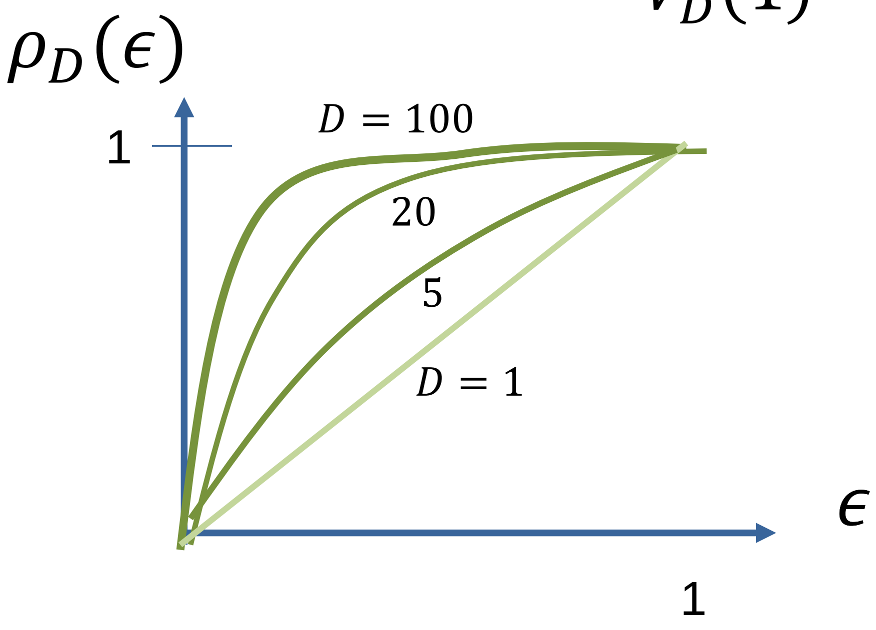
- Example: Consider volume of sphere of radius $r = 1$ in D dimensions
  - What is the fraction of volume in a thin shell of a sphere between $1 - \epsilon \leq r \leq 1$?



Example: Sphere Hardening

- Let $V_D(r)$ = volume of sphere of radius r, dimension D
  - $V_D(r) = K_D r^D$
- Let $\rho_D(\epsilon)$ = fraction of volume in a shell of radius ϵ

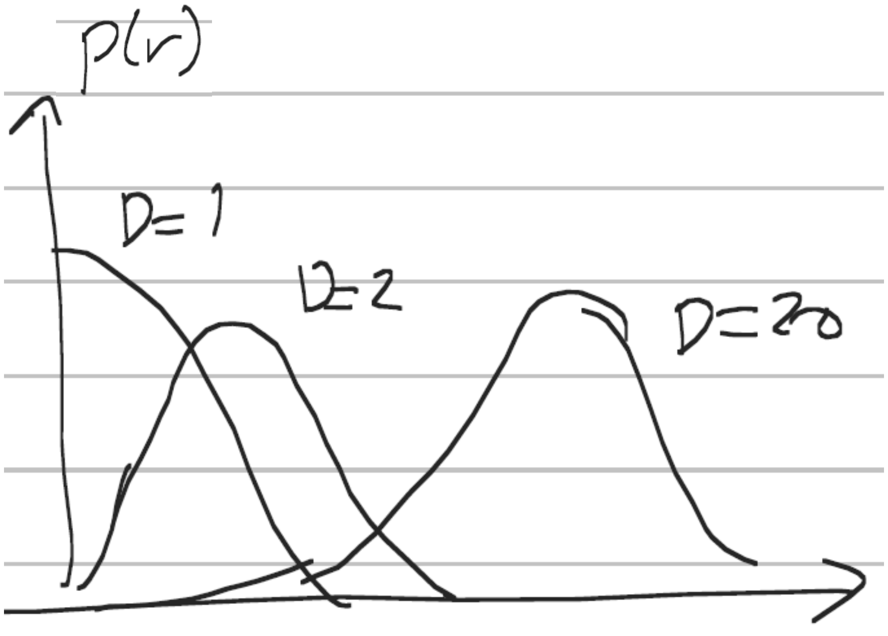
$$\rho_D(\epsilon) = \frac{V_D(1) - V_D(1-\epsilon)}{V_D(1)} = 1 - (1-\epsilon)^D$$



Gaussian Sphere Hardening

- Consider a Gaussian i.i.d. vector
  - $x = (x_1, \ldots, x_D), \;\; x_i \sim N(0,1)$
- As $D \to \infty$, probability density concentrates on shell $\|x\| \approx \sqrt[2]{D}$, even though $x = 0$ is most likely point

- Let $r = \left(x_1^2 + x_2^2 + \cdots + x_D^2\right)^{1/2}$
  - $D = 1$: $p(r) = c\, e^{-r^2/2}$
  - $D = 2$: $p(r) = c\, r\, e^{-r^2/2}$
  - general D: $p(r) = c\, r^{D-1}\, e^{-r^2/2}$



Example: Sphere Hardening

- Conclusions: As dimension increases,
  - All volume of a sphere concentrates at its surface!

- Similar example: Consider a Gaussian i.i.d. vector
  - $x = (x_1, \dots, x_d), \; x_i \sim N(0,1)$
  - As $d \to \infty$, probability density concentrates on shell
    $$\|x\|^2 \approx d$$
  - Even though $x = 0$ is most likely point

Computational Issues

- In high dimensions,
  classifiers need large number of parameters
- Example:
  - Suppose $x = (x_1, \dots, x_d)$, each x_i takes on L values
  - Hence x takes on L^d values
- Consider general classifier $f(x)$
  - Assigns each x some value
  - If there are no restrictions on $f(x)$, needs L^d paramters

Curse of Dimensionality

- Curse of dimensionality: As dimension increases
  - Number parameters for functions grows exponentially
- Most operations become computationally intractable
  - Fitting the function, optimizing, storage

- What ML is doing today
  - Finding tractable approximate approaches for high-dimensions