CS4220: Machine Learning

Yuhang Tian[1]

[1]Technology University of Delft

y.tian-13@student.tudelft.nl

1. Bayesian Classifier Theorem

1.1. Decision Boundary

$$p(x|w_1)p(w_1) \gtrless p(x|w_2)p(x|w_2) \tag{1}$$

1.2. Minimizing the Classification Error Probability (Bayes Error: the minimum error for a certain number of features)

$$P_e = p(w_2)\int_{-\infty}^{x_0} p(x|w_2)dx + p(w_1)\int_{x_0}^{\infty} p(x|w_1)dx \tag{2}$$

where $p(x_0|w_1)p(w_1) = p(x_0|w_2)p(x|w_2)$

1.3. Minimizing the Average Risk

Loss Matrix

$$L = \begin{bmatrix} 0 & \lambda_{12} \\ \lambda_{21} & 0 \end{bmatrix} \tag{3}$$

Risk Function

$$r = \lambda_{21}p(w_2)\int_{-\infty}^{x_0} p(x|w_2)dx + \lambda_{12}p(w_1)\int_{x_0}^{\infty} p(x|w_1)dx \tag{4}$$

1.4. Gaussian pdf in the l-dimensional space

$$p(x) = \frac{1}{(2\pi)^{l/2}|\Sigma|^{l/2}} exp(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)) \tag{5}$$

where $\Sigma = E[(x-\mu)(x-\mu)^T]$

1.5. Bayesian Classifier

discriminant functions

$$g_i(x) = ln(p(x|w_i)p(w_i)) = lnp(x|w_i) + lnp(w_i) \tag{6}$$

Normally Distributed Classifier

$$g_i(x) = -\frac{1}{2}x^T\Sigma_i^{-1}x + \frac{1}{2}x^T\Sigma_i^{-1}\mu_i - \frac{1}{2}\mu_i^T\Sigma_i^{-1}\mu_i + \frac{1}{2}\mu_i^T\Sigma_i^{-1}x + lnp(w_i) + c_i \tag{7}$$

where $c_i = -(l/2)ln2\pi - (1/2)ln(det(\Sigma_i))$

- if l=2, corelation=0

$$g_i(x) = -\frac{1}{2\sigma_i^2}(x_1^2 + x_2^2) + \frac{1}{\sigma_i^2}(\mu_i 1x_1 + \mu_i 2x2) - \frac{1}{2\sigma_i^2}(\mu_{i1}^2 + \mu_{i2}^2) + lnp(w_i) + c_i \tag{8}$$

$g_i(x) - g_j(x) = 0$ are quadratics (i.e., ellipsoids, parabolas, hyperbolas, pairs of lines)

- if covariance matrix is the same in all classes

$$g_i(x) = w_i^T x + b \tag{9}$$

where $w_i = \Sigma^{-1}\mu_i$ and $b = lnp(w_i) - \frac{1}{2}\mu_i^T\Sigma^{-1}\mu_i$

- if Diagonal covariance matrix with equal elements ($\Sigma = \sigma^2 I$)

$$g_i(x) = \frac{1}{\sigma^2}\mu_i^T x + b \tag{10}$$

2. ESTIMATION OF UNKNOWN PROBABILITY DENSITY FUNCTIONS

2.1. ML

we considered θ as an unknown parameter.

$$\hat{\theta}_{ML} = \textit{arg}\ max_\theta \prod_{k=1}^{N} p(x_k;\theta) \tag{11}$$

ML estimate of σ^2

$$\hat{\sigma}^2_{ML} = \frac{1}{N}\sum_{k=1}^{N}(x_k - \mu)^2 \tag{12}$$

ML estimate of μ

$$\hat{\mu}_{ML} = \frac{1}{N}\sum_{k=1}^{N} x_k \tag{13}$$

2.2. MAP

we considered θ as a random vector.

$$p(\theta|X) = \frac{p(\theta)p(X|\theta)}{P(X)} \tag{14}$$

then,

$$\hat{\theta}_{MAP} : \frac{\partial}{\partial\theta}p(\theta|X) = 0\ or\ \frac{\partial}{\partial\theta}p(X|\theta)p(\theta) = 0 \tag{15}$$

2.3. Bayesian Inference

Given the set X of the N training vectors and the *a priori information* about the pdf $p(\theta)$, the goal is to compute the conditional pdf $p(x|X)$.

$$p(x|X) = \int p(x|\theta)p(\theta|X)d\theta \tag{16}$$

with

$$p(\theta|X) = \frac{p(X|\theta)p(\theta)}{p(X)} \tag{17}$$

$$p(X|\theta) = \prod_{k=1}^{N} p(x_k|\theta) \tag{18}$$

3. Normal-based Classifier: Quadratic Discriminant, Linear Discriminant and Nearest Mean

Let's assume that we have two classes:

3.1. Quadratic Discriminant

by eq.7, the quadratic classifier,

$$f(x) = x^T W x + w^T + w_0 \tag{19}$$

with

$$W = \frac{1}{2}(\Sigma_2^{-1} - \Sigma_1^{-1}) \tag{20}$$

$$w = \mu_1^T \Sigma_1^{-1} - \mu_2^T \Sigma_2^{-1} \tag{21}$$

$$w_0 = -\frac{1}{2} ln(det(\Sigma_1)) - \frac{1}{2}\mu_1^T \Sigma_1^{-1} \mu_1 + lnp(y_1) + \frac{1}{2} ln(det(\Sigma_2)) + \frac{1}{2}\mu_2^T \Sigma_2^{-1} \mu_2 - lnp(y_2) \tag{22}$$

(i.e., ellipsoids, parabolas, hyperbolas, pairs of lines)

3.2. Linear Discriminant

by eq.9, the linear classifier,

$$f(x) = w^T x + w_0 \tag{23}$$

with

$$w = \Sigma^{-1}(\mu_2 - \mu_1) \tag{24}$$

$$w_0 = \frac{1}{2}\mu_1^T \Sigma^{-1} \mu_1 - \frac{1}{2}\mu_2^T \Sigma^{-1} \mu_2 + ln\frac{p(y_1)}{p(y_2)} \tag{25}$$

Therefore, the Linear Discriminant has a **strong assumption** on that $\Sigma_1 = \Sigma_2$

3.3. Nearest Mean Classifier

by eq.10, the nearest mean classifier,

$$f(x) = w^T x + w_0 = \sigma^2(g_1(x) - g_2(x)) \tag{26}$$

with

$$w = \mu_2 - \mu_1 \tag{27}$$

$$w_0 = \frac{1}{2}\mu_1^T \mu_1 - \frac{1}{2}\mu_2^T \mu_2 + \sigma^2 ln\frac{p(y_1)}{p(y_2)} \tag{28}$$

Therefore, the Nearest Mean has a **strong assumption** on mutually uncorrelated and of the same variance ($\Sigma_1 = \Sigma_2 = \sigma^2 I$)

4. More Parametric Classifiers

4.1. Logistic Classifier

$$\begin{cases} p(y_1|x) = \frac{1}{e^{-(w^T x + w_0)} + 1} \\ p(y_2|x) = \frac{1}{e^{(w^T x + w_0)} + 1} \end{cases} \tag{29}$$

Maximize Log Likelihood

$$lnp(y|x) = \sum_{i=1}^{N} ln(\frac{1}{e^{-y_i(w^T x_i + w_0)} + 1}) \tag{30}$$

4.2. Fisher Classifier

$$y_i = w^T x_i \begin{cases} \geq 0 & \text{if class 1} \\ < 0 & \text{if class 2} \end{cases} \tag{31}$$

Minimize Square Loss

$$L(w) = \sum_{i=1}^{N} (w^T x_i - y_i)^2 \tag{32}$$

4.3. The Perception

Forward&backward propagation and update w

$$w \leftarrow w + \eta y x \tag{33}$$

5. Non-parametric Classification

In the above sections, all the classifiers are based on **Parametric Classification** method, more precisely, based on normal distribution and Bayes Theorem. In this section, it will mainly demonstrate two non-parametric classifiers - Parzen Classifier and Nearest Neighbour Classifier (Both methods are sensitive to the scaling of the features).

5.1. Parametric vs. Non-parametric

- Parametric: Assumptions can greatly simplify the learning process, but can also limit what can be learned. Algorithms that simplify the function to a known form are called parametric machine learning algorithms.
- Non-parametric: Algorithms that do not make strong assumptions about the form of the mapping function are called non-parametric machine learning algorithms. By not making assumptions, they are free to learn any functional form from the training data.

5.2. Histogram method

it's easy, so I just skip it.

5.3. Parzen Density Estimation

- Kernel

$$f(x) = \begin{cases} 0 & \text{if |r|>h} \\ \frac{1}{V} & \text{if |r|}\leq \text{ h} \end{cases} \tag{34}$$

- Parzen Classifier

$$p(z|h) = \frac{1}{n}\sum_{i=1}^{n} K(||z - x_i||, h) \tag{35}$$

- Parzen plugs in the Gaussian density:

$$p(x|w_i) = \frac{1}{n_i}\sum_{j=1}^{n_i} N(x|x_j^{(i)}, hI) \tag{36}$$

5.4. Nearest Neighbour Classification

$$p(x) = p(x|w_m)p(w_m) = \frac{k_m}{n_m V_k} \cdot \frac{n_m}{n} \tag{37}$$

where V_k is the volume of the sphere centered at x with radius r (the distance to the k-th nearest neighbor)

6. More Non-parametric Classifiers

GitHub Markdown

6.1. SVM

By putting some constraints on the linear classifier, the VC dimension can be reduced. Why do that? Ans: When h is small, the true error is close to the apparent error

$$\begin{cases} w^T x_i + b \geq +1 & \text{for } y_i = +1 \\ w^T x_i + b \leq -1 & \text{for } y_i = -1 \end{cases} \tag{38}$$

Core Idea of SVM: Find the decision boundary, while maximize the margin

1. The above two equation can be merged into one

$$y_i(w^T x_i + b) - 1 \geq 0 \tag{39}$$

2. The distance between the two boundaries

$$maximize\ \frac{2}{||w||} \rightarrow minimize\ \frac{1}{2}||w||^2 \tag{40}$$

3. by Lagrange Multiplier

$$L = \frac{1}{2}||w||^2 - \sum \alpha_i[y_i(wx_i + b) - 1] \tag{41}$$

4. by $\frac{\partial L}{\partial w} = 0$

$$w = \sum \alpha_i y_i x_i \tag{42}$$

5. by $\frac{\partial L}{\partial b} = 0$

$$\sum \alpha_i y_i = 0 \tag{43}$$

6. put eq.42 back to eq.41

$$L = \sum \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j x_i \cdot x_j \tag{44}$$

7. put eq.42 back to decision rule

$$\begin{cases} \sum \alpha_i y_i x_i \cdot u_i + b - 1 \geq 0 & \text{Then, +} \\ \sum \alpha_i y_i x_i \cdot u_i + b - 1 \leq 0 & \text{Then, -} \end{cases} \tag{45}$$

8. kernelize

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) \tag{46}$$

6.2. Agglomerative Hierarchical Clustering

1. Determine distances between all clusters
   - Two nearest objects in the clusters: single linkage
   - Two most remote objects in the clusters: complete linkage
   - Cluster centers: average linkage
2. Merge clusters that are closes
3. IF #clusters>1 THEN GOTO 1

- Dendrogram: Cut at "largest jump"$\rightarrow$ Clustering
- Fusion Graph: Cut at "largest drop"$\rightarrow$ Clustering

7. Regression

7.1. Intuitively Understanding

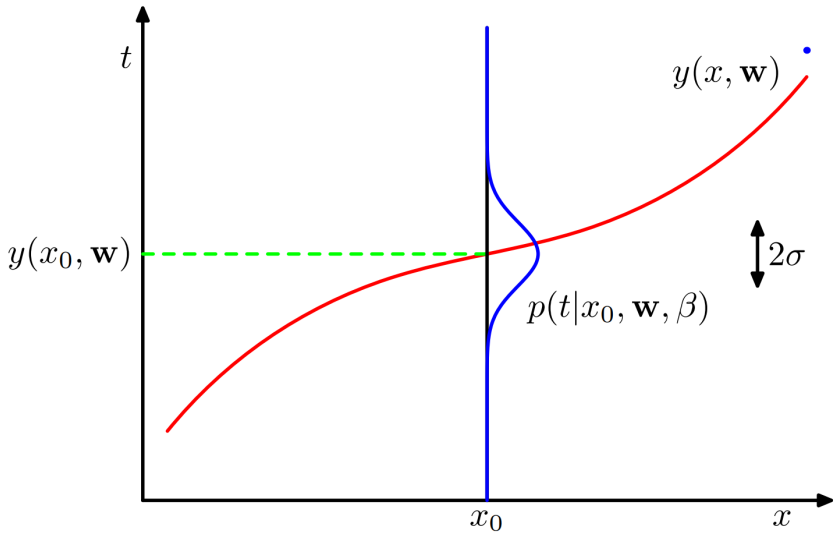


Figura 1. Regression Curve

Suppose that there is a fixed x_0 and a bunch of choices of θ (here are μ and $\sigma^2(\beta^{-1})$). If we expect the predicted value $\hat{y}(x_0, w)$ to locate on the true value t_0, the probability of this occurrence $p(t_0|x_0, w, \beta)$ should be maximized.

7.2. Maximum Likelihood Regression

$$p(t|x,w,\beta) = \prod_{n=1}^{N} N(t_n|y(x_n,w),\beta^{-1}) \tag{47}$$

Log Likelihood function

$$lnp(t|x,w,\beta) = -\frac{\beta}{2}\sum_{n=1}^{N} y(x_n,w) - {t_n}^2 + \frac{N}{2}ln\beta - \frac{N}{2}ln(2\pi) \tag{48}$$

To solve

$$\frac{\partial}{\partial w} lnp(t|x_0,w,\beta) = 0 \tag{49}$$

and

$$\frac{\partial}{\partial \beta^{-1}} lnp(t|x_0,w,\beta) = 0 \tag{50}$$

If we use linear regression, the solutions of eq.49 and eq.50

$$w_{ML} = (X^T X)^{-1} X^T Y \ \beta_{ML}^{-1} = \frac{1}{N}(w_{ML}^T X - Y) \tag{51}$$

7.3. Max a Posterior Regression

Suppose that we have some knowledge about w $N(0,\alpha I)$

$$w_{MAP} : (\prod_{i=1}^{N} p(y_i|w^T x_i,\sigma^2))p(w|0,\alpha I) \tag{52}$$

$$\frac{\partial}{\partial w}(\prod_{i=1}^{N} p(y_i|w^T x_i,\sigma^2))p(w|0,\alpha I) = 0 \rightarrow w_{MAP} = (X^T X + \frac{\sigma^2}{\alpha} I)^{-1} X^T Y \tag{53}$$

8. Regularization

8.1. Keep Eigenvalues Away From Zero

Add identity to XX^T

$$\hat{w} = (X^T X + \lambda I)^{-1} X^T Y \tag{54}$$

8.2. LASSO, L1 Norm

$$min_\beta \frac{1}{N}\sum_{i=1}^{n}(y_i - w^T x_i)^2 + \lambda||w|| \tag{55}$$

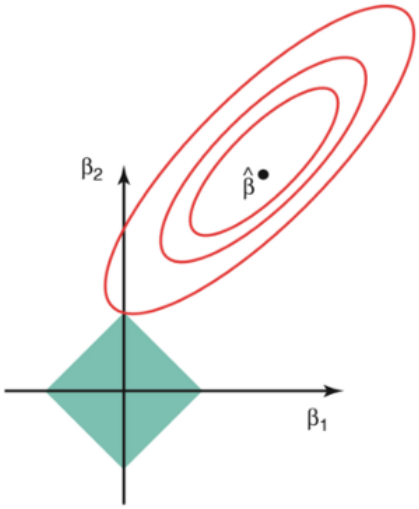


Figura 2. LASSO/L1 Regularization

$\lambda \propto \frac{1}{\tau}$

8.3. Ridge, L2 Norm

$$min_{\beta} \frac{1}{N} \sum_{i=1}^{n} (y_i - w^T x_i)^2 + \lambda ||w||^2 \tag{56}$$

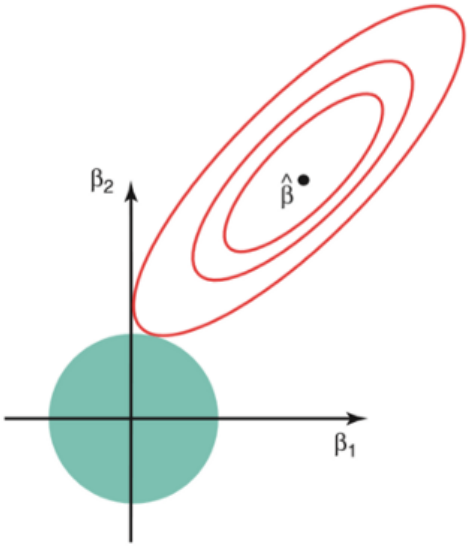


Figura 3. Ridge/L2 Regularization

$\lambda \propto \frac{1}{\tau}$

8.4. L1 vs. L2

- L1 is for feature selection

- L2 is for avoiding overfitting

Read More

9. Data Pre-processing

9.1. LDA vs. PCA

10. The Number of Parameter For Estimation

Assume we have a training set consisting of n objects and d features with 2 classes.

10.1. Bayesian Classifiers

- for mean: $2 \times d$
- for variance:
  - if correlation$= 0$: $2 \times d$
  - if correlation$\neq 0$: $2 \times \frac{d \times (d+1)}{2}$
- for class prior: 1
- totally: $1 + 3d + d^2$ or $1 + 4d$

10.2. SVC

If it is a linear one

- constrain C: 1
- Lagrange multipliers: n
- totally: $1 + n$

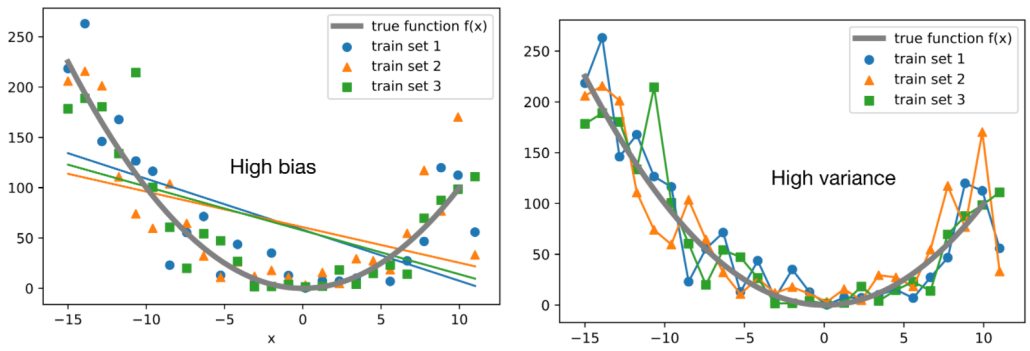
10.3. Parzen

If the kernel is fixed

- kernel width h: 1
- totally: 1

11. Curves

11.1. Bias-Variance Decomposition



(a) High Bias (b) High Variance

Figura 4. High Bias vs. High Variance

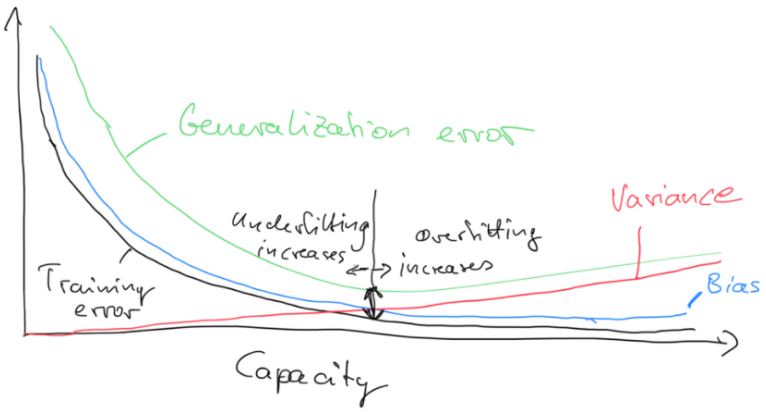


Figura 5. Decomposition Of Loss

11.2. Cross Validation Curve

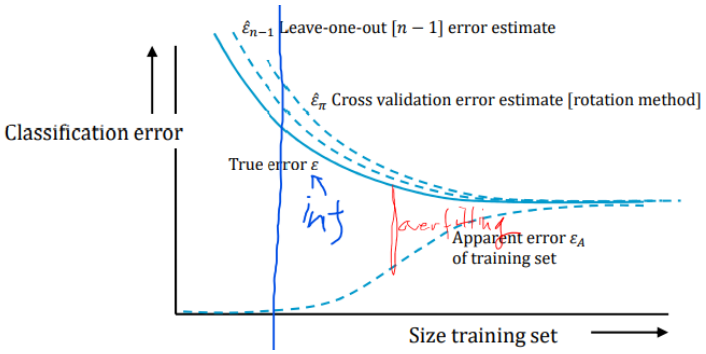


Figura 6. Cross Validation Curve

11.3. Learning Curve

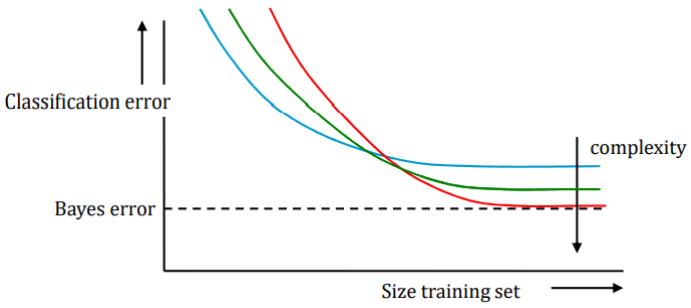


Figura 7. Learning Curve

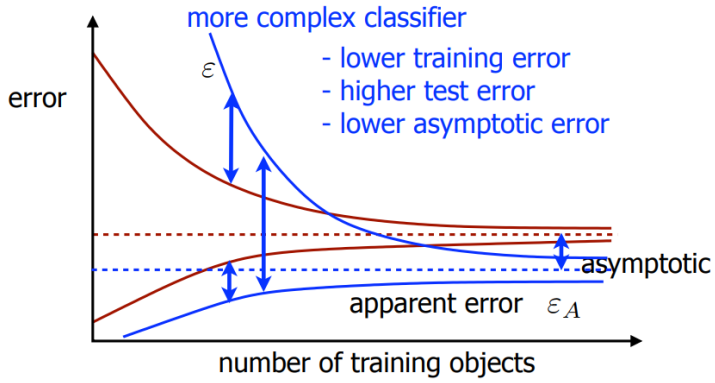


Figura 8. Learning Curve 2

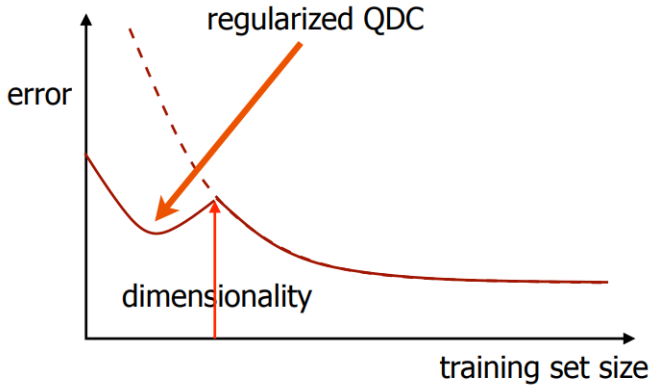


Figura 9. Regularized QDC

11.4. Feature Curve

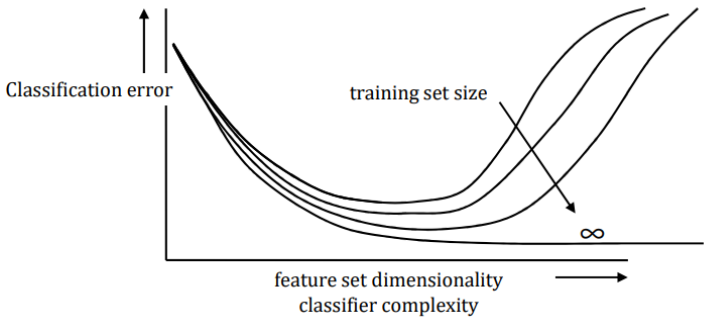


Figura 10. Feature Curve

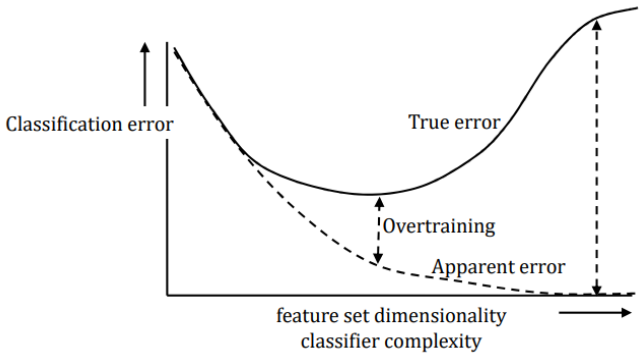


Figura 11. Curse of Dimensionality

11.5. ROC: receiver operating characteristic curve

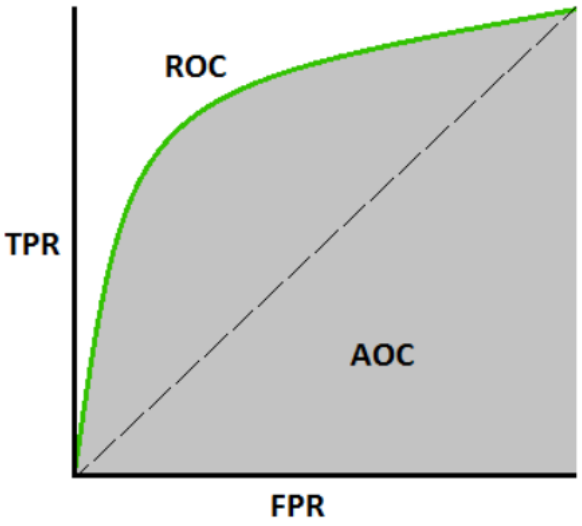
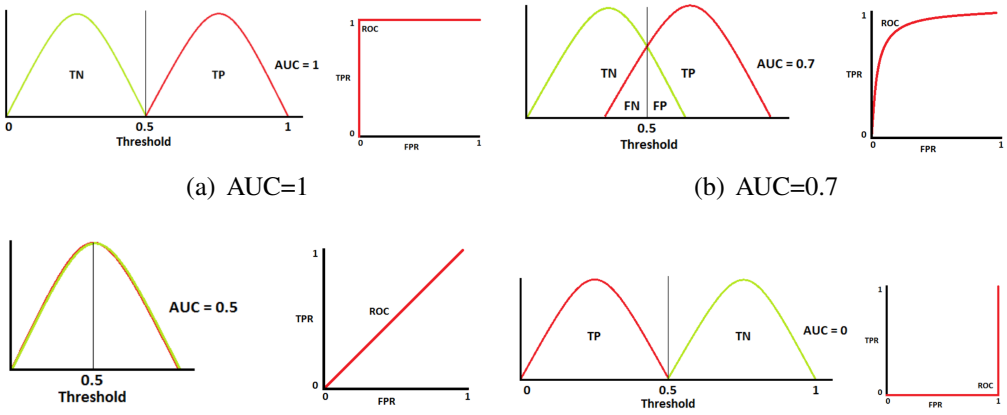


Figura 12. ROC Curve

$$TPR/Recall/Senstivity = \frac{TP}{TP + FN} \tag{57}$$

$$FPR = \frac{FP}{TN + FP} \tag{58}$$



(a) AUC=1 (b) AUC=0.7

(c) AUC=0.5 (d) AUC=0

Figura 13. Different ROCs

Area under the Curve of ROC (AUC ROC)

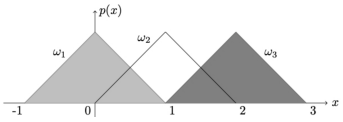
12. DAG

Directional-separation (D-separation): If every path between A and B is blocked then $A \perp B|C$ conditionally independent: Given that I have observed C, observing B will not give me additional information about A

13. Exam Question

13.1. Bayes Classifier & Bayes Error

For the Bayes classifier, one has to know the true class conditional probabilities. **False**
The Bayes error is always smaller or equal to the estimated true error. **False**
The Bayes error is always larger or equal to the apparent error. **False**
The Quadratic Discriminant classifier is a generative classifier. **True**
Even if the Bayes classifier is nonlinear, the nearest mean may still outperform the k-nearest neighbor classifier. **True**
In three dimensions, the nearest mean classifier can typically perfectly separate any configuration of four points from two classes. **False**
You want to know if you are one of the few people that knows all there is to know about artificial intelligence. To find out, you decide to go through a host of exams and other tests. The overall score you receive indicates that you are indeed one of the happy few. Knowing so much about AI, however, you do not jump to conclusions and you check with your local expert how accurate these scores are. "Well," the person in the lab coat tells you, "of course they are not perfect, but these test will correctly identify 99% of the people that know all of ML and will be incorrect in a mere 0.1% when people do not belong to this select group." True or false: the probability of you knowing all of ML will be greater than 0.99. **False**
By reducing the dimensionality one can decrease the Bayes error. **False**



- Assume that all classes are equiprobable. Where is the optimal decision boundary (Bayes' classifier) is located? **At x=0.5 and x=1.5**
- What is the expected confusion matrix for the Bayes' classifier when the classes are equiprobable, and we classify 1000 points (rounded to the nearest integer).

$$\begin{pmatrix} 292 & 42 & 0 \\ 42 & 250 & 42 \\ 0 & 42 & 292 \end{pmatrix}$$

- Assume that the class priors are changed such that class 1 is now twice as likely as class 2 and 3: $p(w_1) = 2p(w_2) = 2p(w_3)$. How does the optimal decision boundary change? How will the expected confusion matrix look like.

$$\begin{pmatrix} 389 & 111 & 0 \\ 56 & 163 & 31 \\ 0 & 31 & 219 \end{pmatrix}$$

13.2. Regularization & MAP

Consider the standard probabilistic form of (unregularized) linear regression based on a Gaussian noise model. Denote the likelihood of this model by p(X|w) with X the observed data. Let q(w) be a pdf that is uniform on all values of w for which the L1 norm is smaller than some constant a. The MAP solution, using this prior, is equivalent to L1-regularized least squares regression. **True**
Consider the standard probabilistic form of (unregularized) linear regression based on a Gaussian noise model. Denote the likelihood of this model by p(X|w) with X the observed data. Let q(w) be a pdf that is uniform on all values of w for which the L2 norm is smaller than some constant a. The MAP solution, using this prior, is equivalent to... **L2 regularized least squares regression**

13.3. SVC

The VC dimension is only defined for support vector classifiers. **False**
To find the Support Vector classifier on a training set, you need to know the class prior probabilities $p(y_i)$. **False**

13.4. Logistic Classifier/Regression

The logistic classifier assumes that the decision boundary can be modeled by a logistic function. **False**
Linear regression is a special form of logistic regression. **False**

13.5. Cluster

k-Means clustering results reproduce exactly when the initialization of the cluster means is non-random. **True**
In hierarchical clustering, one way to determine the number of clusters is to cut the dendrogram at its largest jump. **True**
Average linkage clustering typically results in more compact clusters than single linkage clustering. **True**
The Davies-Bouldin index looks at the worst-case for each cluster in terms of overlap with other clusters. **True**
Hierarchical clustering results depend on the initialization. **False**

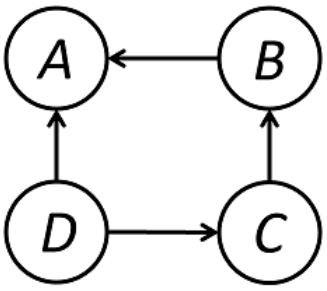
13.6. Feature Transformation/Reeducation

The Parzen classifier, i.e., the generative classifier in which the underlying classes are modeled by Parzen estimators, is insensitive to feature scaling. **False**
Principal component analysis reduces the dimensionality such that the class separability is optimized. **False**
We trained a classifier on some dataset, and we obtained the following feature curve. What happens when we increase the number of features further? (We assume that each of the features is informative for the classification problem) **The curve will go up eventually**
We trained a classifier on some dataset, and we obtained the following feature curve: What happens when we increase the number of training objects? **The curve will move down overall**

13.7. Bias-Variance

Consider 30 draws of 10 training samples from a 2-class classification problem p(x,y) in 3 dimensions. Say we train the same classifier on all of these data sets and measure their performances on the same, large and independent test set. It turns out that the trained classifiers misclassify the same part of the test set. Is this a result of classifier bias or classifier variance? **Bias**

13.8. DAG



When are D and B independent of each other? **Conditional upon observing C**