SF2935: Modern methods of statistical learning
Lecture 3
Supervised Learning, LDA and QDA. k-Nearest Neighbors classifiers.

Tatjana Pavlenko

3 November 2017

KTH Matematik

C) **Minimize the total probability of misclassification (TPM)**:

$$\text{TPM} = p(2|1)\pi_1 + p(1|2)\pi_2 = \pi_1 \int_{R_2} f_1(\mathbf{x})d\mathbf{x} + \pi_2 \int_{R_1} f_2(\mathbf{x})d\mathbf{x}.$$

- This leads (proof is similar to ECM case) to the discriminant rule:

Assign $\mathbf{x}$ to Π_1 if $\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \frac{\pi_2}{\pi_1}$, else to Π_2.

- **Special cases of ECM rule:** if $\pi_1 = \pi_2$ then $\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \frac{c(1|2)}{c(2|1)}$.
- If $c(1|2) = c(2|1)$ then $\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \frac{\pi_2}{\pi_1}$. Same as TPM and Bayes classifier.
- If $c(1|2) = c(2|1)$ and $\pi_1 = \pi_2$ then $\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > 1$. Likelihood ratio classification rule.

KTH Matematik

Two multivariate normal populations: LDA and QDA

- When we use normal (Gaussian) distributions for each population Π_i, this leads to *linear* or *quadratic* discriminant analysis, LDA or QDA.
- However, this approach is quite general, and other distributions can be used as well. We will focus on normal distributions.
- Assume that $f_i(\mathbf{x})$ is $N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ corresponding to Π_i, $i = 1, 2$. $\boldsymbol{\mu}_i$ is a class-specific mean vector, $\boldsymbol{\Sigma}_i$ is the class covariance matrix.
- $\mathbf{X} \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Here $\mathbb{E}(\mathbf{X}) = \boldsymbol{\mu}$, $\mathbb{C}\mathrm{ov}(\mathbf{X}) = \boldsymbol{\Sigma}$ and the density is

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right).$$

KTH Matematik

Two multivariate normal populations: LDA

- Assume that $f_i(\mathbf{x})$ is $N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma})$ corresponding to Π_i. $\boldsymbol{\Sigma}$ is the covariance matrix which is *common* for both populations. Then

$$\log\left(\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}\right) = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}\mathbf{x} - \frac{1}{2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2) = D(\mathbf{x}) \text{ (say)}$$

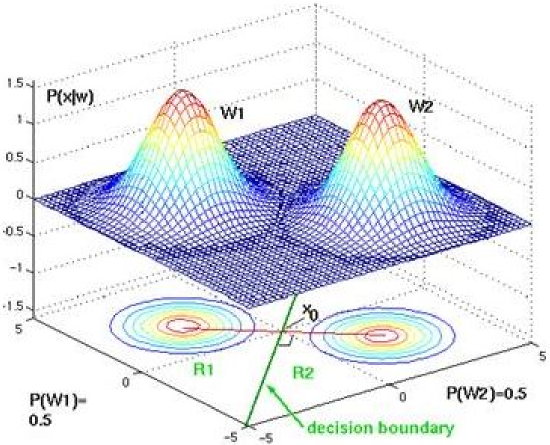
- The discriminant rule that minimizes EPM is:

  Assign $\mathbf{x}$ to Π_1 if $D(\mathbf{x}) > \log \frac{\pi_2 c(1|2)}{\pi_1 c(2|1)}$, else to Π_2.

- Proof on the board.
- For $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$, $D(\mathbf{x})$ is *linear* in $\mathbf{x}$ which is the reason for the name *Linear discriminant function* or *Linear discriminant analysis* (LDA).
- $\boldsymbol{\Sigma}_1 \neq \boldsymbol{\Sigma}_2$ results in a *quadratic* discriminant rule or QDA, will be discussed more later.

KTH Matematik

Linear discriminant functions.



- In practice, population parameters $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}$ are unknown. We estimate $D(\mathbf{x})$ from the data!
- Estimation technique: plug-in estimated $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}$ into $D(\mathbf{x})$. This gives a *sample* discriminant rule.
- Given the data $\mathbf{X}_i : n_i \times p$ from Π_i, calculate $\bar{\mathbf{x}}_i$, $\mathbf{S}_i$ (unbiased).
- Since $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$ use $\mathbf{S}_{\text{pooled}} = \frac{(n_1-1)\mathbf{S}_1+(n_2-1)\mathbf{S}_2}{n_1+n_2-2}$ and obtain

$$\widehat{D}(\mathbf{x}) = (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)'\mathbf{S}_{\text{pooled}}^{-1}\mathbf{x} - \frac{1}{2}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)'\mathbf{S}_{\text{pooled}}^{-1}(\bar{\mathbf{x}}_1 + \bar{\mathbf{x}}_2).$$

- The sample EPM rule is:

Assign $\mathbf{x}$ to Π_1 if $\widehat{D}(\mathbf{x}) > \log \frac{\pi_2 c(1|2)}{\pi_1 c(2|1)}$, else to Π_2.

KTH Matematik

Some remarks on LDA

- Estimation of π_i's: Usually, $\pi_i = \frac{n_i}{n_1+n_2}$ is assumed. Otherwise, use Bayes' prior, guess, ...
- With $\widehat{D}(\mathbf{x})$ there is no assurance that the resulting rule will minimize the ECM in a particular application. This is because the optimal rule was derived assuming population densities $f_i(\mathbf{x})$ were known *completely*. But $\widehat{D}(\mathbf{x})$ is expected to perform well if the sample size n_i is large.
- Denote by $\widehat{\mathbf{d}}' = (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)'\mathbf{S}_{\text{pooled}}^{-1}$ and let $\widehat{y}(\mathbf{x}) = \widehat{\mathbf{d}}'\mathbf{x}$, $\bar{y}_i = \hat{\mathbf{d}}'\bar{\mathbf{x}}_i$, $i = 1, 2$.
- When $\frac{\pi_2 c(1|2)}{\pi_1 c(2|1)} = 1$ the discriminant rule becomes

$$\widehat{y}(\mathbf{x}) > \frac{1}{2}(\bar{y}_1 + \bar{y}_2).$$

- As $\widehat{y}(\mathbf{x})$ and $\bar{y}_i$'s are linear combinations, the multivariate expressions convert to univariate ones.

Some remarks on LDA (cont.)

- Consider the rule (with $\frac{\pi_2 c(1|2)}{\pi_1 c(2|1)} = 1$) again. We have

$$D(\mathbf{x}) = \log\left(\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}\right) = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}\mathbf{x} - \frac{1}{2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2).$$

- Rule: Assign $\mathbf{x}$ to Π_1 is $D(\mathbf{x}) > 0$ otherwise to Π_2
- The rule is called *linear discriminant function* (LDF)
- *Bayes decision boundary* $\{\mathbf{x}|D(\mathbf{x}) = 0\}$ is a *hyperplane* (of size $p-1$) dividing the two classes.
- See ISL, p. 144: Bayes decision boundary represents the set of values $\mathbf{x}$ for which $\delta_1(\mathbf{x}) = \delta_2(\mathbf{x})$ where

$$\delta_i(\mathbf{x}) = \mathbf{x}'\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_i - \frac{1}{2}\boldsymbol{\mu}_i'\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_i, i = 1, 2.$$

(the term $\log(\pi_i)$ disappears since it is the same for Π_1 and Π_2.

Optimality criteria are based on probabilities of misclassification. The smaller they are the more optimal is the classifier performance. Recall that

$$\mathrm{TMP} = \pi_1 \int_{R_2} f_1(\mathbf{x})d\mathbf{x} + \pi_2 \int_{R_1} f_2(\mathbf{x})d\mathbf{x}.$$

- $\min(\mathrm{TMP}) = \text{optimum misclassification rate, OMR}$ is obtained for R_1 and R_2 determined as

$$R_1 : \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \geq \frac{\pi_2}{\pi_1}, \qquad R_2 : \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} < \frac{\pi_2}{\pi_1}$$

- For completely known Π_i's distributions OMR can be computed *exact*.
- Assume that Π_i is $N(\mu_i, \Sigma)$, $i = 1, 2$ and $\pi_1 = \pi_2 = \frac{1}{2}$. Using

$$D(\mathbf{x}) = \log\left(\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}\right) = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}\mathbf{x} - \frac{1}{2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2)$$

we get the rule: assign $\mathbf{x}$ to Π_1 if $D(\mathbf{x}) > 0$, else to Π_2.

$$R_1 : \underbrace{(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}\mathbf{x}}_{y(\mathbf{x})=\boldsymbol{d}'\mathbf{x}} \geq \frac{1}{2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2)$$

$$R_2 : \underbrace{(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}\mathbf{x}}_{y(\mathbf{x})=\boldsymbol{d}'\mathbf{x}} < \frac{1}{2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2)$$

A random variable $Y(\mathbf{X}) = \boldsymbol{d}'\mathbf{X}$ is univariate normal (Why?) with means and a variance given by

$$\mu_{iY} = \boldsymbol{d}'\boldsymbol{\mu}_i = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_i, \quad i = 1, 2,$$

and

$$\sigma_Y^2 = \boldsymbol{d}'\boldsymbol{\Sigma}\boldsymbol{d} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) = \Delta^2,$$

where $\Delta^2 = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$ is the *Mahalanobis distance* between Π_1 and Π_2.

Evaluating performance accuracy (cont)

Now

$$p(2|1) = P(\text{misclassify a } \Pi_1 \text{ observation as } \Pi_2)$$

$$= P(Y(\mathbf{X}) < \frac{1}{2}(\mu_1 - \mu_2)'\mathbf{\Sigma}^{-1}(\mu_1 + \mu_2)) = \Phi\left(-\frac{\Delta}{2}\right).$$

Similarly,

$$p(2|1) = P(\text{misclassify a } \Pi_2 \text{ observation as } \Pi_1)$$

$$= P(Y(\mathbf{X}) \geq \frac{1}{2}(\mu_1 - \mu_2)'\mathbf{\Sigma}^{-1}(\mu_1 + \mu_2)) = 1 - \Phi\left(\frac{\Delta}{2}\right) = \Phi\left(-\frac{\Delta}{2}\right).$$

Therefore

$$\text{OMR} = \min(\text{TMP}) = \frac{1}{2}\Phi\left(-\frac{\Delta}{2}\right) + \frac{1}{2}\Phi\left(-\frac{\Delta}{2}\right) = \Phi\left(-\frac{\Delta}{2}\right).$$

For example, if $\Delta = 2.56$ then $\text{OMR} = 0.1$; if $\Delta = 4.56$ then $\text{OMR} = 0.01$. Figure, see white board!

Evaluating performance accuracy (cont)

- For example, for LDA with Π_i's defined by $N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma})$ we have

$$\text{OMR} = \min(\text{TMP}) = \Phi\left(-\frac{\Delta}{2}\right)$$

where $\Delta^2 = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$ is the *Mahalanobis distance* between Π_1 and Π_2, $\Phi(\cdot)$ is the cdf of $N(0,1)$.

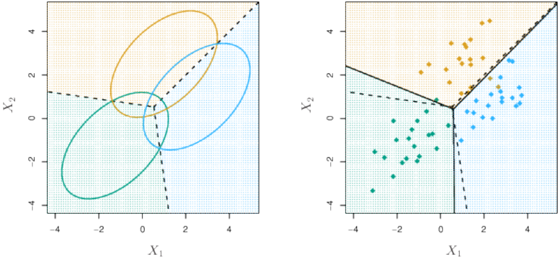
- Apparent error rate (AER): The fraction of observations in a training data that are misclassified by the estimated classifier:

Actual	Assign. to Π_1	Assign. to Π_2
Π_1	$n_1 - m_1$	m_1
Π_2	m_2	$n_2 - m_2$

- $\text{AER} = \frac{m_1+m_2}{n_1+n_2}$

KTH Matematik

Multi-sample LDA



Figur: Three Gaussian classes with $p = 2$. **Left:** Ellipses represent regions containing 95% of the probability for each of three classes. The dashed lines are Bayesian decision boundaries. **Right:** 20 observations were generated from each class and the corresponding LDA decision boundaries are presented as solid lines along with the Bayesian boundaries (dashed lines). Overall, the sample-based LDA is close to Bayesian decision boundaries. See Ch. 4 in ISL.

Two multivariate normal populations: QDA

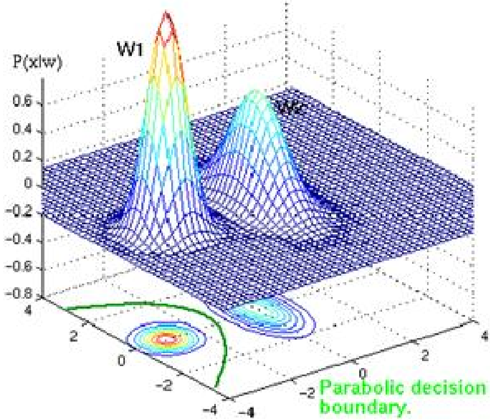
- Assume that $f_i(\mathbf{x})$ is $N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_\mathbf{i})$ corresponding to Π_i. If covariance matrices are *different* for Π_i's, i.e. $\boldsymbol{\Sigma}_1 \neq \boldsymbol{\Sigma}_2$, then (show this by analogy to LDA)

$$Q(\mathbf{x}) = \log\left(\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}\right) = -\frac{1}{2}\mathbf{x}'\left(\boldsymbol{\Sigma}_1^{-1} - \boldsymbol{\Sigma}_2^{-1}\right)\mathbf{x} + \left(\boldsymbol{\mu}_1'\boldsymbol{\Sigma}_1^{-1} - \boldsymbol{\mu}_2'\boldsymbol{\Sigma}_2^{-1}\right)\mathbf{x} + q,$$

$$\text{where } q = \frac{1}{2}\log\left(\frac{|\boldsymbol{\Sigma}_1|}{|\boldsymbol{\Sigma}_2|}\right) + \frac{1}{2}\left(\boldsymbol{\mu}_1'\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2'\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\mu}_2\right).$$

- The rule is: assign $\mathbf{x}$ to Π_1 if $Q(\mathbf{x}) > \log\frac{\pi_2 c(1|2)}{\pi_1 c(2|1)}$, else to Π_2.
- The rule is called *quadratic discriminant function* (QDF).
- $Q(\mathbf{x})$ contains $\mathbf{x}'\left(\boldsymbol{\Sigma}_1^{-1} - \boldsymbol{\Sigma}_2^{-1}\right)\mathbf{x}$, i.e. is *quadratic* in $\mathbf{x}$.
- If $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$ QDA becomes LDA.

Two multivariate normal populations: QDA



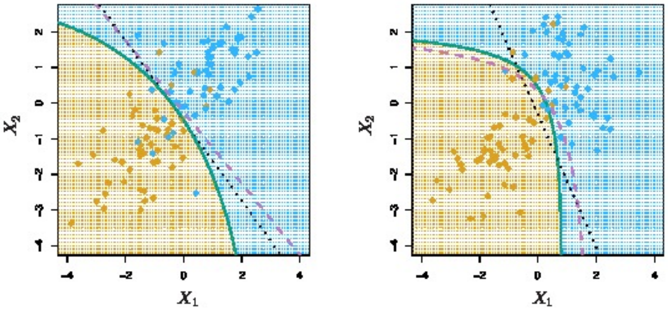
TWO MULTIVARIATE NORMAL POPULATIONS: QDA

- For the sample based $Q(\mathbf{x})$, use the plug-in estimator of $Q(\mathbf{x})$ with $\bar{\mathbf{x}}_i$, $\mathbf{S}_i$, $i = 1, 2$. *Bayes decision boundary* $\{\mathbf{x} | Q(\mathbf{x}) = 0\}$ dividing the two classes is quadratic in $\mathbf{x}$.
- See ISL, p. 149: Bayes decision boundary represents the set of values $\mathbf{x}$ for which $\delta_1(\mathbf{x}) = \delta_2(\mathbf{x})$ where

$$\delta_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)' \boldsymbol{\Sigma}_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) - \frac{1}{2} \log |\boldsymbol{\Sigma}_i|,$$

  (the term $\log(\pi_i)$ disappears under equal priors for Π_1 and Π_2.
- QDA, because it allows for more flexibility for the covariance matrix, tends to fit the data better than LDA, but then it has more parameters to estimate. The number of parameters increases significantly with QDA.
- Comparison of the computational cost: with p variables LDA requires estimating a *pooled* covariance matrix with $p(p+1)/2$ parameters. QDA estimates in total of $cp(p+1)/2$ parameters for c class covariance matrices.

Two multivariate normal populations: QDA



FIGUR: **Left:** The Bayes (purple dashed), LDA (black dotted), and QDA (green solid) decision boundaries for $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2$. Since under $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2$ the true Bayes classifier is linear, it is more accurately approximated by LDA than by QDA. **Right:** Data are generated for $\mathbf{\Sigma}_1 \neq \mathbf{\Sigma}_2$. The true Bayes classifier is non-linear, better fit is obtained with estimated QDA.

KTH Matematik

LDA vs QDA: some remarks

- For the sample size n_i small ($n_i \leq 25$) and p approaching n_i, and $\mathbf{\Sigma}_1 \approx \mathbf{\Sigma}_2$ LDA outperforms QDA.
- For p small ($p \leq 6$) and $\mathbf{\Sigma}_1 \approx \mathbf{\Sigma}_2$, LDA and QDA have similar performance accuracy
- When $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ differ too much the misclassification probabilities obtained for LDA are not reliable. In such cases QDA performs much better, even when both p and n_i are large.
- When sample sizes n_i are small QDA performance is usually poor.
- QDA is relatively robust to deviation from normality.
- LDA is not robust to non-normality.

Model-based vs model-free approach in supervised classification

- **Previously:** Model-based approach and probabilistic methods. Model assumption, e.g. p-variate normal population distribution, which we hope describes the reality accurately enough. But

  All models are wrong, but some are useful – George Box

- **Instead:** Non-parametric approach, no assumptions about the model and the shape of the decision boundary. These methods belong to the intersection of statistics and data mining.
- **Idea:** Given is a set of training data with known classes. Without any model assumption, we can use heuristics and suggest rules for classifying a new observation using the training data.
- Evaluation can then be performed by CV, i.e by splitting the given data into a training and test set.
- Neares Neighbour (NN-) classifier: *do as your neighbour does*. See the idea on the board.

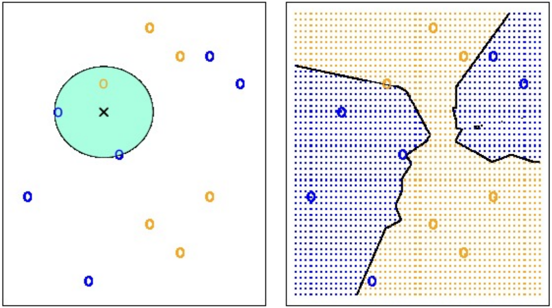
All models are wrong, and increasingly you can succeed without them.
Perer Norvig, Research Director at Google

- Example: kNN classifier assigns the new observation by letting the k Neares Neighbors – the k points that are closest the new observation point – to *vote* about the class of the new point.
- kNN classification is completely non-parametric approach: Let $\mathcal{C}$ be the class variable, $\mathcal{C} \in \{1, \ldots, c\}$. Given is a positive integer k and a test observation $\mathbf{x}$. The kNN technique consists of the following steps:
  - I) identify the k points of *training data* closest to $\mathbf{x}$ and form a subset $\mathcal{N}$. Then
  - II) estimate the class posterior probability $P(\mathcal{C} = j | X = \mathbf{x}) = \frac{1}{k} \sum_{i \in \mathcal{N}} \mathbb{I}(C_i = j)$, which is the fraction of points in $\mathcal{N}$ that belongs to the class j, $j = 1, \ldots, c$ and
  - III) apply Bayes rule, i.e. assign $\mathbf{x}$ to the class j with highest class posterior probability (majority vote among the k neighbors, ties are broken at random).

See more details about kNN technique in Ch2, ISL.

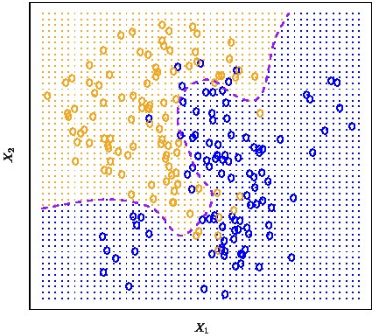
KTH VETENSKAP OCH KONST
KTH Matematik

kNN CLASSIFICATION



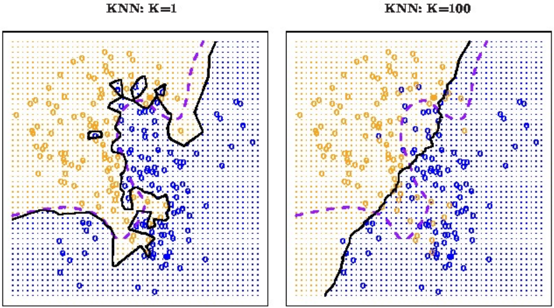
FIGUR: kNN with $k = 3$. **Left:** A test observation $\mathbf{x}$ (marked by $\times$) at which a predicted class label is desired along with three closest observations from the training data forming the neighborhood $\mathcal{N}$ of $\mathbf{x}$. **Right:** kNN decision boundary is shown in black. Color grid shows classes.

BAYES DECISION BOUNDARY: EXAMPLE



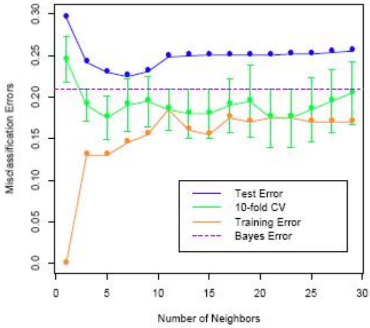
FIGUR: Example using a simulated data with $n_1 = n_2 = 100$ and $\mathbf{X}' = (X_1, X_2)$. The orange region is the set of points for which $P(\mathcal{C} = \text{orange}|\mathbf{X} = \mathbf{x}) > 0.5$ and for the blue region this prob. is < 0.5. Since we know how the data were generated we can calculate these probabilities for each value of (X_1, X_2). The purple dashed line represents the *Bayes decision boundary*, i.e the points where the probability is exactly 0.5.

kNN CLASSIFICATION BOUNDARY – EFFECT OF k



FIGUR: A comparison of the kNN decision boundaries obtained for $k = 1$ and $k = 100$ on the simulated data. The Bayes decision boundary is shown in dashed-purple. The bias of 1NN estimated classifier is low but the variance is high. The larger k is, the smoother the decision boundary. Or we can think of the complexity of kNN as lower when k increases.

CV APPROACH FOR CHOOSING k



FIGUR: Error rates for *test* and *training* data, and for 10-fold cross-validation are plotted against k, the number of neighbors. The test error behavior indicates that there is a balance between k and error rate, i.e there is a preference for k in a certain range. The broken purple line in the background represents the error of Bayes classifier.

kNN classification – choosing k

- k as a "hyper-parameter". Use *cross validation* strategy:
  - split the training set into two independent parts, *training* and *validation* set,
  - train *kNN* classifier on training sets for a different values of k,
  - test the different learned classifiers on validation set and pick k giving best performance on the validation set.
- Some more practical issues with kNN classification.
  - kNN is very sensitive to the scale of input variables if we use e.g. Euclidean distance in feature space $d(\boldsymbol{x}_j, \boldsymbol{x}) = \|\boldsymbol{x}_j - \boldsymbol{x}\|$ a measure of dissimilarity.
  - Good practice is to *normalize* data, i.e to rescale to $[0, 1]$ or $[-1, 1]$. Usually we standardize each of the features to have mean zero and variance one by
$$x \longrightarrow \frac{x - \bar{x}}{\hat{\sigma}}.$$
- kNN classification needs efficient data structure to look for the closes point, especially in high-dimensional feature space. One idea: *discriminant adaptive NN*, (DANN).

Discriminant adaptive NN

- Implicit in NN classification is the assumption that the class probabilities are roughly constant in the neighborhood, and hence the averaging provides good estimates. Suppose, for a moment that $p = 2$ and that the class probabilities vary only in the horizontal direction. If we would know this, we would *stretch* the neighborhood in the vertical direction to reduce the bias of estimates (Obs! variance remains the same).
- This idea is called for *adapting* the metric used in NN classification, so that the resulting neighborhoods stretch out in the directions for which the class probabilities do not change much.
- In high-dimensional feature space, the class probabilities might change only in a low-dimensional subspaces and adapting the dissimilarity measure can provide considerable improvement in the classification accuracy.

Discriminant adaptive NN (cont.)

Adaptive NN methods are presented in Ch. 13.4 of ESL.

The discriminant adaptive NN (DANN) metric is defined as

$$\mathcal{D}(\boldsymbol{x}_j, \boldsymbol{x}) = (\boldsymbol{x}_j - \boldsymbol{x})' \boldsymbol{\Omega} (\boldsymbol{x}_j - \boldsymbol{x}),$$

$$\boldsymbol{\Omega} = \boldsymbol{W}^{-1/2} \left[\boldsymbol{W}^{-1/2} \boldsymbol{B} \boldsymbol{W}^{-1/2} + \epsilon \boldsymbol{I} \right] \boldsymbol{W}^{-1/2} = \boldsymbol{W}^{-1/2} \left[\boldsymbol{B}^* + \epsilon \boldsymbol{I} \right] \boldsymbol{W}^{-1/2},$$

$\boldsymbol{W} = \sum_{c=1}^{C} \pi_c \boldsymbol{W}_c$ is the pooled within-class covariance matrix, and $\boldsymbol{B} = \sum_{c=1}^{C} \pi_c (\bar{\boldsymbol{x}}_c - \bar{\boldsymbol{x}})(\bar{\boldsymbol{x}}_c - \bar{\boldsymbol{x}})'$ is the between-class covariance matrix. ϵ (usually $\epsilon = 1$ works well) rounds the infinite strip to ellipsoid, see ESL, fig. 13.13 and 13.14 on. p. 476 and 478, resp.

- A neighborhood of (say) 50 points is formed for the novel observation $\boldsymbol{x}$ and $\boldsymbol{W}$ and $\boldsymbol{B}$ are first obtained using only these 50 nearest neighbors of $\boldsymbol{x}$.
- The resulting (adapted) metric is then used in a NN classifier.

- In words: at each novel $\boldsymbol{x}$ a neighborhood of 50 points is formed and class distribution among these points are used to decide how to deform the neighborhood, i.e how to obtain the *adapted* $\mathcal{D}(\boldsymbol{x}_j, \boldsymbol{x})$.
- Thus for each new observation $\boldsymbol{x}$, a potentially new (adapted) metric is used in the classification rule.
- Interpretation of adaptation above: we first sphere the data w. r. to $\boldsymbol{W}$, and then stretch the neighborhood in the zero-eigenvalue directions of $\boldsymbol{B}^*$, the between-class covariance matrix for the sphered data.

Model-based vs model-free approach: strengths

Tabell: **Strengths**

Model-based methods	**Model-free methods**
• Probabilistic foundation	• No assumption is needed for the underlying model
• Allow for deriving optimal methods/properties	• *Optimization* using the test data
• Provide natural and helpful interpretation of the results	• Often is based on a natural heuristic foundation
• Allow to control error rates by e.g. specifying the level of significance	• Often work well for $p > n$ and even $p \gg n$ cases

Model-based vs model-free approach: weaknesses

Tabell: **Weaknesses**

Model-based methods	**Model-free methods**
• Might be based on the asymptotic properties which do not work well in small sample size settings	• Are based on training data which may not be representative
• The model might be a poor description of reality	• Impossible to specify optimal methods, method's accuracy using test data
•The inference is only under the model assumption	• Usually not as good as model-based method if model assumptions are correct
• Difficult to evaluate whether model assumptions are correct	• Weak theoretical support, relies on heuristics.

KTH Matematik