Recap 5

Classification

ISLP 4, ESL 4

Jilles Vreeken
Krikamol Muandet

UNIVERSITÄT DES SAARLANDES

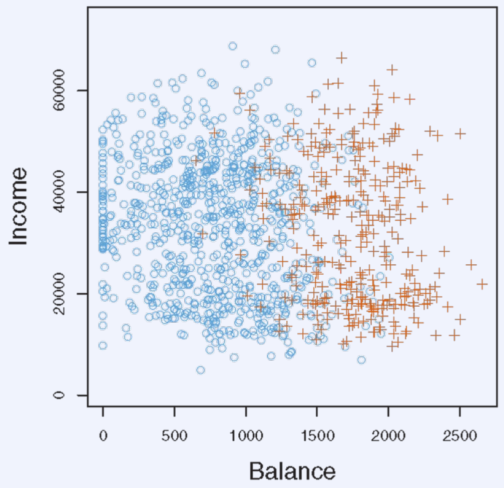
CISPA HELMHOLTZ CENTER FOR INFORMATION SECURITY

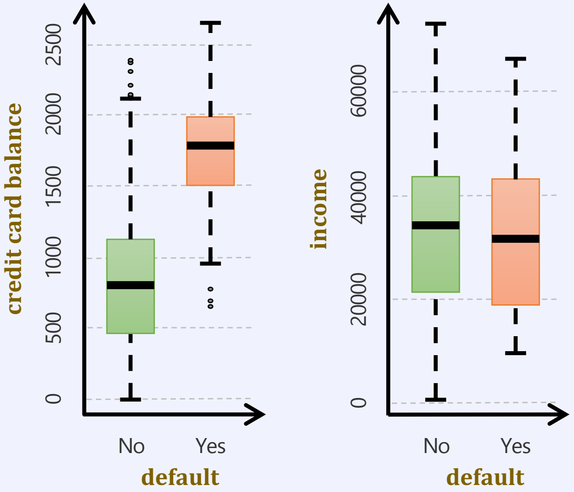
Classification Overview

In classification, we want to predict categorical outputs

Example will someone pay back their loan? yes or no?

- inputs: annual **income**, monthly **balance, student** status





Linear Discriminant Analysis

Bayesian classification for K classes

- Use Bayes' formula to determine posterior density per class $\Pr(Y = k \mid X = x)$

$$p_k(x) = \Pr(Y = k \mid X = x) = \frac{\pi_k f_k(x)}{\sum_{\ell=1}^{K} \pi_\ell f_\ell(x)}$$

- Classify each point to its most probable class

Univariate LDA

- Assume each $f_k(x)$ is a univariate gaussian with the same variance

  → Bayesian classfier $p_k(x) = \frac{\pi_k f_k(x)}{\sum_{\ell=1}^{K} \pi_\ell f_\ell(x)} \propto \pi_k \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{1}{2\sigma_k^2}(x - \mu_k)^2\right)$

- Discriminant: $\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log \pi_k$
- Assign sample to class with the largest discriminator
- Decision boundary for two classes is the set of points for which the discriminator are equal

Multivariate LDA

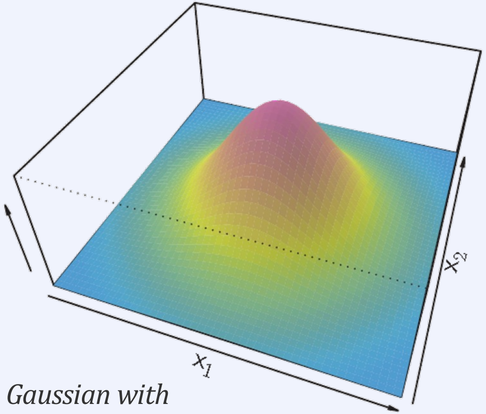
Model assumptions

- each class is a multivariate Gaussian
- the covariance matrix is the same for all classes

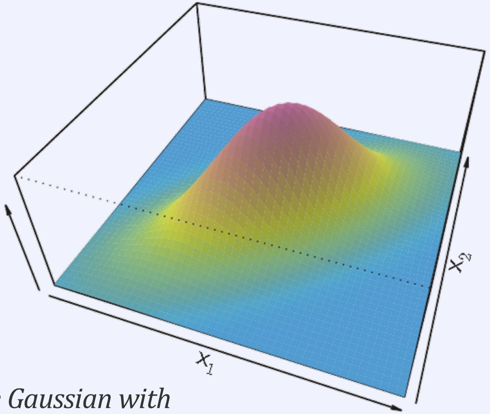
$$f_{k(x)} = \frac{1}{(2\pi)^{\frac{p}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}^{-1}(x-\mu_k)\right)$$

$$\delta_k(x) = x^T\mathbf{\Sigma}^{-1}\mu_k - \frac{1}{2}\mu_k^T\mathbf{\Sigma}^{-1}\mu_k + \log\pi_k$$

- $\mathbf{\Sigma}$ is the $p \times p$ covariance matrix of the inputs $\mathbf{\Sigma} = \mathrm{Cov}(x)$
- model is fitted using sample estimates similar to the 1D case
- μ easy, but $\mathbf{\Sigma}$ is the hardest to estimate



Multivariate Gaussian with two uncorrelated predictors



Multivariate Gaussian with two correlated predictors (0.7)

Quadratic Discriminant Analysis (QDA)

We give up the assumption that the covariances of all classes are all the same

For QDA we have

$$f_{k(x)} = \frac{1}{(2\pi)^{\frac{p}{2}}|\mathbf{\Sigma}_k|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}_k^{-1}(x-\mu_k)\right)$$

$$\delta_k(x) = -\frac{1}{2}x^T\mathbf{\Sigma}_k^{-1}x + x^T\mathbf{\Sigma}_k^{-1}\mu_k - \frac{1}{2}\mu_k^T\mathbf{\Sigma}_k^{-1}\mu_k + \log \pi_k$$

- Discriminator is quadratic in x
- One covariance matrix per class
- #parameters $Kp(p+3)/2$

For LDA we had

$$f_{k(x)} = \frac{1}{(2\pi)^{\frac{p}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}^{-1}(x-\mu_k)\right)$$

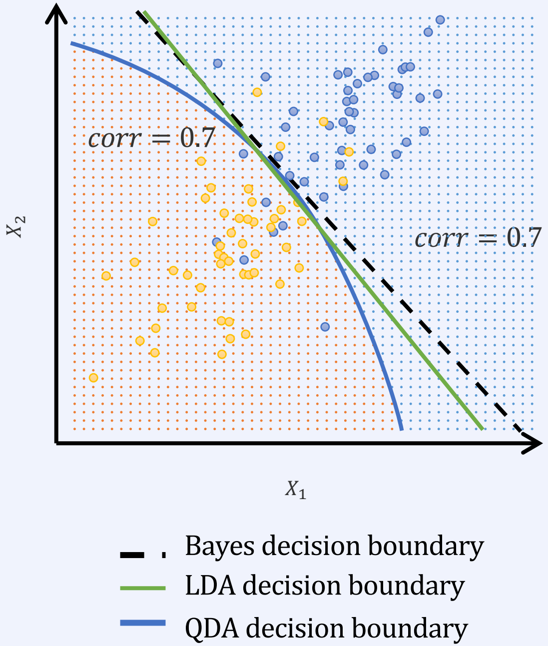
$$\delta_k(x) = x^T\mathbf{\Sigma}^{-1}\mu_k - \frac{1}{2}\mu_k^T\mathbf{\Sigma}^{-1}\mu_k + \log \pi_k$$

- Discriminator is linear in x
- One covariance matrix for all classes
- #parameters $(2K+p+1)p/2$

Example LDA vs. QDA

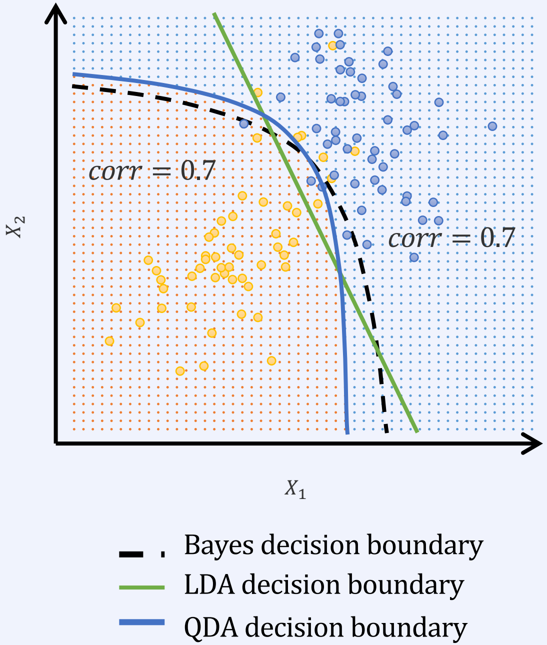
Two-class problem with $\Sigma_1 = \Sigma_2$

QDA overtrains



Two-class problem with $\Sigma_1 \neq \Sigma_2$

LDA overtrains



Fitting LDA and QDA Models

Again, we use sample estimates

- $\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i$
- $\widehat{\boldsymbol{\Sigma}} = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T$
- $\widehat{\boldsymbol{\Sigma}}_k = \frac{1}{n_k-K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T$
- $\pi_k = n_k/n$

To simplify calculation we use the eigenvalue decomposition of the covariance matrices

$$\widehat{\boldsymbol{\Sigma}}_k = \boldsymbol{U}_k \boldsymbol{D}_k \boldsymbol{U}_k^T$$

- $\boldsymbol{U}_k$ is a $p \times p$ orthonormal matrix
- $\boldsymbol{D}_k$ is a diagonal matrix of decreasing positive eigenvalues $\boldsymbol{d}_{kl}$

The main terms in the discriminants,

$$\delta_k(x) = -\frac{1}{2} \log\left|\widehat{\boldsymbol{\Sigma}}_k\right| - \frac{1}{2}(x - \mu_k)^T \widehat{\boldsymbol{\Sigma}}_k^{-1}(x - \mu_k) + \log \pi_k$$

then turn into

$$\log\left|\widehat{\boldsymbol{\Sigma}}_k\right| = \sum_l \log d_{kl}$$

$$(x - \hat{\mu}_k)^T \hat{\Sigma}_k^{-1}(x - \hat{\mu}_k) = \left[\boldsymbol{U}_k^T(x - \hat{\mu}_k)\right]^T D_k^{-1}\left[U_k^T(x - \hat{\mu}_k)\right]$$

The LDA estimator

- Step 1: Normalize $\boldsymbol{X}$ to spherical covariance
  $$X^* \leftarrow \boldsymbol{D}^{-1/2} \boldsymbol{U}^T X$$
- Step 2: Classify to the closest class centroid in the transformed space, where distance is weighted by the class prior probabilities $\boldsymbol{\pi}_k$

Comparison of the Classification Methods

We now know four classifiers: LDA, QDA and logistic regression

- when should we use which?

Logistic regression and LDA are surprisingly closely related

- univariate binary setting $p_2(x) = 1 - p_1(x)$
- log-odds for LDA are $\log \frac{p_1(x)}{1-p_1(x)} = c_0 + c_1 x$

  (difference of two linear discriminants)
- while for logistic regression $\log \frac{p_1(x)}{1-p_1(x)} = \beta_0 + \beta_1 x$

Similar, but different

- β_0 and β_1 are maximum likelihood estimates
- c_0 and c_1 are estimated from sample mean and variance of Gaussian distribution
- relationship extends to multivariate data: LR and LDA often give similar results – but not always!
- LDA makes stronger assumptions

Error Types: Sensitivity vs. Specificity

Example **default** with **balance** and **student** as inputs

- training error for LDA is 2.75%
- data is highly unbalanced, we have only 3,33% positives
- the No-only classifier has an error of already only 3,33%

Sensitivity $\text{Sens} = TP/(TP + FN) = TP/P^*$

- fraction of correctly predicted positives

Specificity $\text{Spec} = TN/(TN + FP) = TN/N^*$

- fraction of correctly predicted negatives

- No $\text{Sens} = \frac{0}{333} = 0\%$, $\text{Spec} = \frac{9{,}667}{9{,}667} = 100\%$
- LDA $\text{Sens} = \frac{81}{333} = 24.3\%$, $\text{Spec} = \frac{9{,}644}{9{,}667} = 99.8\%$
- LDA approximates the Bayes classifier, it minimizes error on all observations

LDA Model Results

Prediction	True Default Status No (−)	Yes (+)	Total
No (−)	9,644	252	9,896
Yes (+)	23	81	104
Total	**9,667**	**333**	**10,000**

Prediction	True Default Status No (−)	Yes (+)	Total
No (−)	TN	FN	N
Yes (+)	FP	TP	P
Total	N^*	P^*	n

Confusion matrix

Type-1 error
False positive

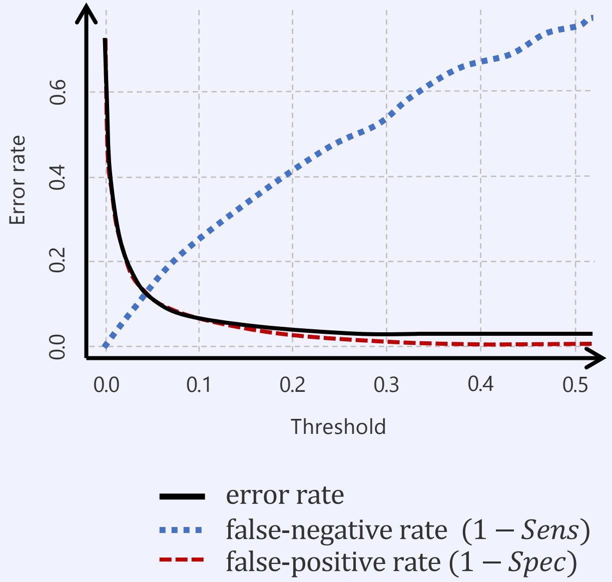
Type-2 error
False negative

Error Types: Sensitivity vs. Specificity

Biasing the classifier trades sensitivity for specificity

$$\log((p_k(x))/(p_l(x))) = \delta_k(x) - \delta_l(x)$$

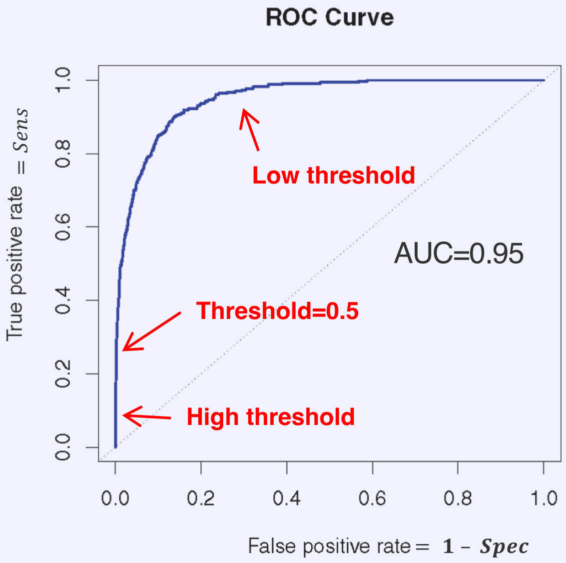
- move the decision threshold between class **no** or **yes** from
  $$\Pr(\textbf{default} = \text{yes} \mid X = x) = 0.5$$
- we can increase sensitivity by choosing
  $\Pr(\textbf{default} = \text{yes} \mid X = x) < 0.5$
  as this assigns more points to class **yes**
- for $\Pr(\textbf{default} = \text{yes} \mid X = x) < 0.2$
  - Sens = $195/333 = 58.6\%$
  - Spec = $9{,}432/9{,}667 = 97.6\%$
  - Error = $373/10{,}000 = 3.73\%$
- error rates change smoothly when we move the threshold



ROC Curves

Receiver-Operating Characteristic (ROC) curves plot $Sens$ against $1 - Spec$ for all thresholds

- Area Under the ROC-Curve (AUC) measures the quality of a classifier independent of the choice of that threshold
- optimally $Spec = Sens = 1$ for any threshold ($AUC = 1$)
- random classifier performs on the diagonal ($AUC = 0.5$)
- if the ROC curve goes below the diagonal, we can improve accuracy by inverting the classifier



ROC curves are not influenced by imbalance of the data

- balance only affects locations of a threshold along the curve