Announcements

- Google group `chemometrics-events@go.gdch.de`

- Workshop “Chemometrics meets Artificial Intelligence”
  March 31st 2022 – April 1st, Berlin, BAM
  `www.gdch.de/chemometrik2022`

Classification

Chemometrics for Spectroscopists

Intensive Course Kraków
2021-11-29 – 12-03

Claudia Beleites
Chemometric Consulting Claudia Beleites

Classification

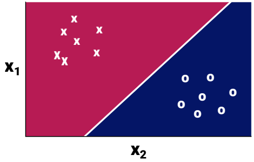
Classification

. . . assigns cases (spectra) to pre-specified groups or classes.

Classification

Classification

…assigns cases (spectra) to pre-specified groups or classes.

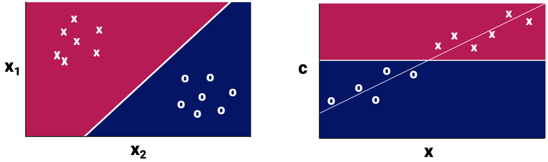


- discriminative: a case always belongs to exactly one class

Classification

Classification

. . . assigns cases (spectra) to pre-specified groups or classes.

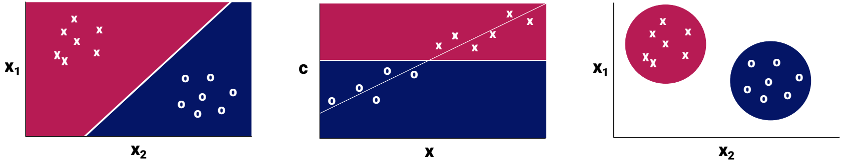


- discriminative: a case always belongs to exactly one class
- threshold-type: regression in disguise

Classification

Classification

. . . assigns cases (spectra) to pre-specified groups or classes.

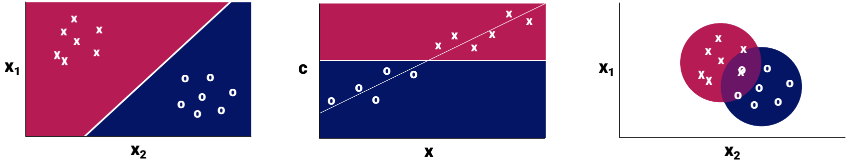


- discriminative: a case always belongs to exactly one class
- threshold-type: regression in disguise
- one-class classification (class models): each class independent of other classes

Classification

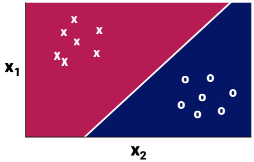
Classification

. . . assigns cases (spectra) to pre-specified groups or classes.



- discriminative: a case always belongs to exactly one class
- threshold-type: regression in disguise
- one-class classification (class models): each class independent of other classes

Comparison of Classification Types



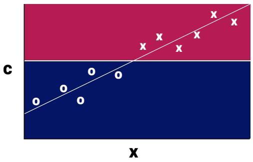
Discriminative Classification

- It is impossible that a case does not belong to one of these classes
- Classes are mutually exclusive

✘ smallest/most difficult class determines uncertainty (class boundary)
✘ if there is one such difficult class, performance for *all* classes can suffer
✘ changes in one class change the whole model
✔ Fewer samples needed iff assumptions met

✘ All classes must be known at training time

Comparison of Classification Types



Threshold-Type Classification

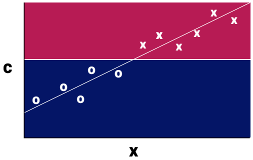
- Regression in disguise: threshold on metric outcome
- Classes are mutually exclusive

✘ Only class labels available for training

⇝ needs classifier that uses only cases close to class boundary

- SVM classifier, Logistic Regression

Comparison of Classification Types



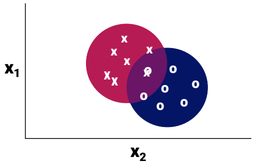
Threshold-Type Classification

- Regression in disguise: threshold on metric outcome
- Classes are mutually exclusive

✔ Metric labels for training available ⇝ regression + threshold
✔ Metric labels more informative than labels ⇝ fewer samples needed

- e.g. PLS-DA

Comparison of Classification Types

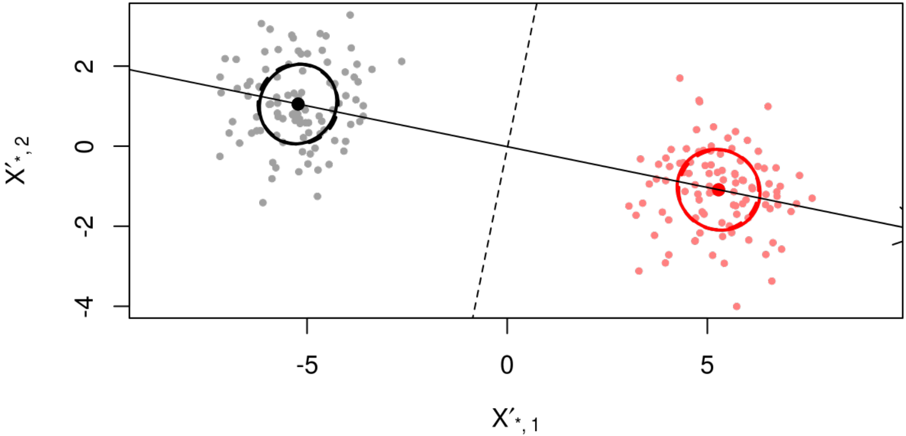


One-Class Classifiers aka. Class Models

- Each *positive* class independent of all others

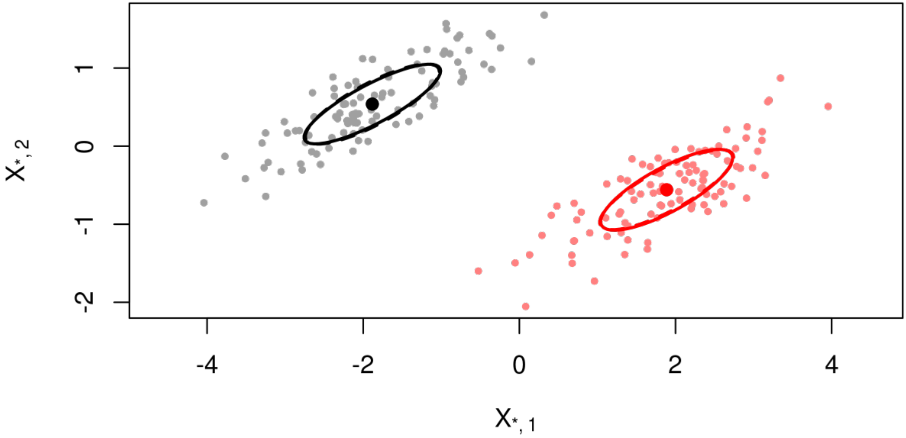
✔ negative classes occur naturally as “not <positive> class”

Linear Discriminant Analysis



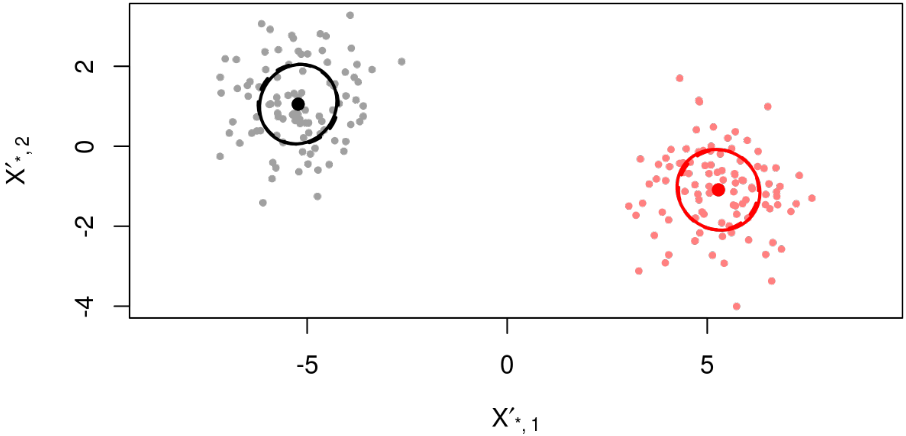
- Maximise variance between classes wrt. variance within classes $\frac{\mathbf{COV}_b}{\mathbf{COV}_w}$
- Finding a separation plane is easy for spherical point clouds.
- Calculate within-class covariance matrix $\mathbf{COV}_w$, and
- project so that $\mathbf{COV}'_w$ is the unit matrix $\mathbf{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

Linear Discriminant Analysis



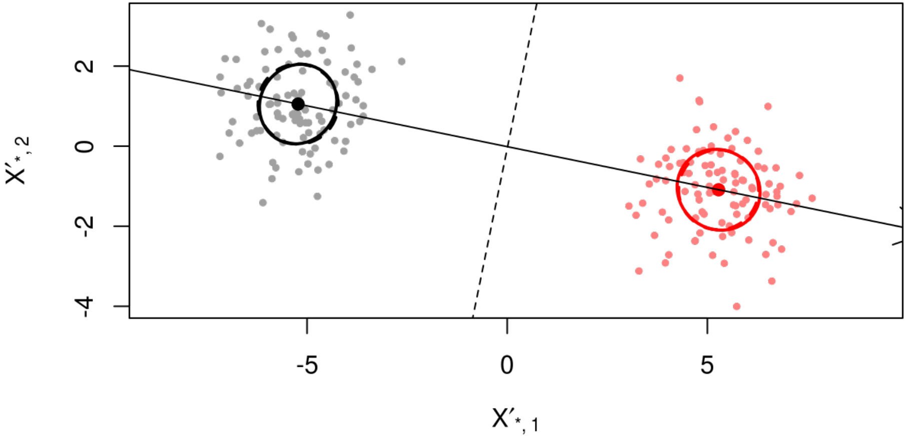
- Maximise variance between classes wrt. variance within classes $\frac{\mathbf{COV}_b}{\mathbf{COV}_w}$
- Finding a separation plane is easy for spherical point clouds.
- Calculate within-class covariance matrix $\mathbf{COV}_w$, and
- project so that $\mathbf{COV}'_w$ is the unit matrix $\mathbf{I} = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$

Linear Discriminant Analysis



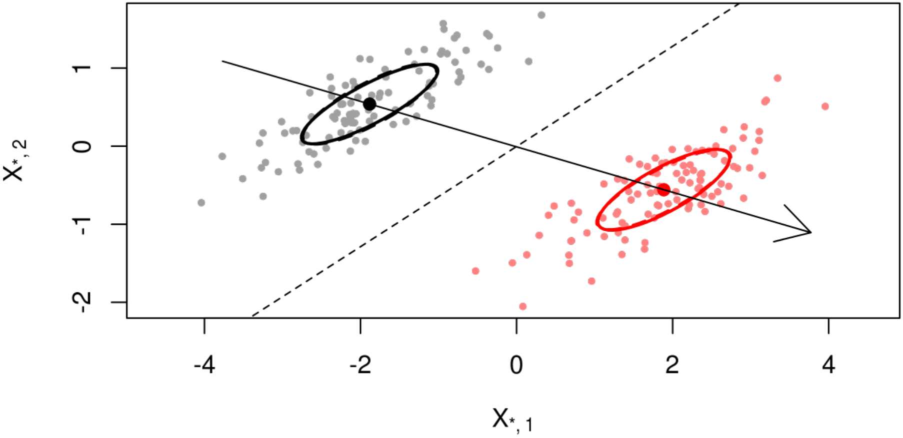
- Maximise variance between classes wrt. variance within classes $\frac{\mathbf{COV}_b}{\mathbf{COV}_w}$
- Finding a separation plane is easy for spherical point clouds.
- Calculate within-class covariance matrix $\mathbf{COV}_w$, and
- project so that $\mathbf{COV}'_w$ is the unit matrix $\mathbf{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

Linear Discriminant Analysis



- Maximise variance between classes wrt. variance within classes $\frac{\mathbf{COV}_b}{\mathbf{COV}_w}$
- Finding a separation plane is easy for spherical point clouds.
- Calculate within-class covariance matrix $\mathbf{COV}_w$, and
- project so that $\mathbf{COV}'_w$ is the unit matrix $\mathbf{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

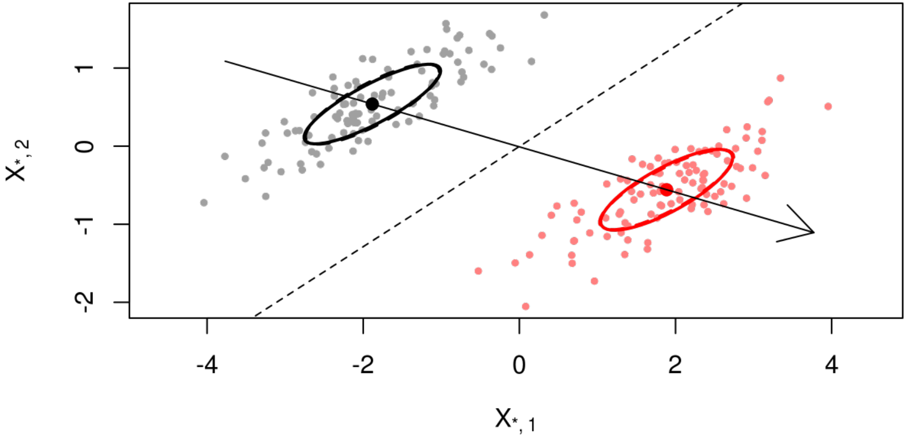
Linear Discriminant Analysis



- Maximise variance between classes wrt. variance within classes $\frac{\mathbf{COV}_b}{\mathbf{COV}_w}$
- Finding a separation plane is easy for spherical point clouds.
- Calculate within-class covariance matrix $\mathbf{COV}_w$, and
- project so that $\mathbf{COV}'_w$ is the unit matrix $\mathbf{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

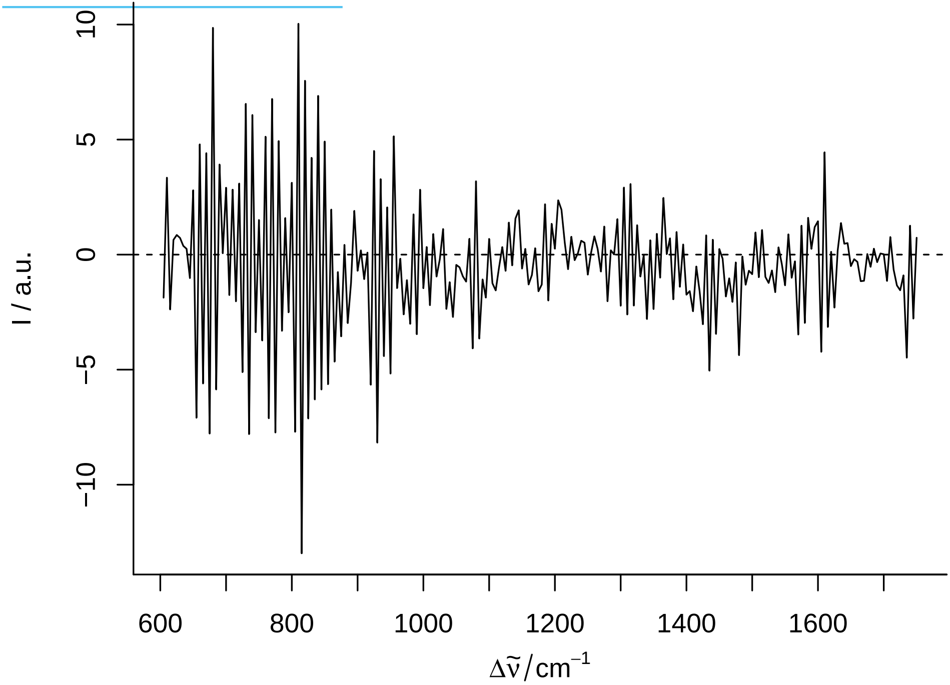
CHEMO METRIX

Linear Discriminant Analysis

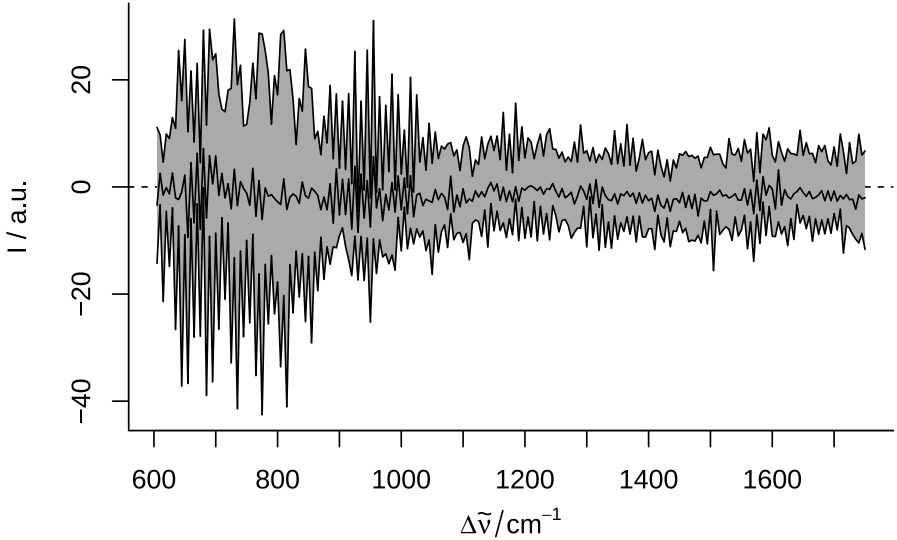


- ✘ Needs $n > p$
- ✔ Like for ILS: do LDA on PLS (or PCA) scores
- ✘ Sensitive to outliers (“heavy tails”)
- ✔ Multivariate normal distribution ⇝ LDA optimal
- ✔ In practice reliable standard technique

LDA: Coefficients



LDA: Stability of Coefficients



PCA-LDA

Bilinear model:

$$\begin{aligned}\mathbf{L}^{(n\times k-1)} &= \mathbf{T}^{(n\times m)}\mathbf{B}'^{(m\times k-1)} + \varepsilon \\ &= \mathbf{X}^{(n\times p)}\mathbf{P}^{T(p\times m)}\mathbf{B}'^{(m\times k-1)} + \varepsilon \\ &= \mathbf{X}^{(n\times p)}\mathbf{B}''^{(p\times k-1)} + \varepsilon\end{aligned}$$

Idea: Do a PCA first and then LDA on the PCA scores

! careful about details: both PCA and LDA center data according to their own criteria

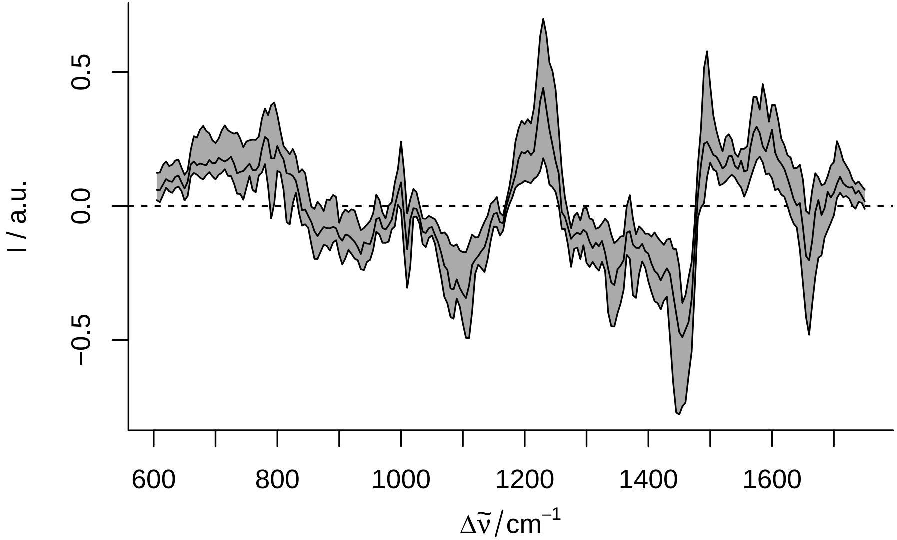
⇒ either do one step after the other (no $\mathbf{B}''$),

⇒ or ensure centering for PCA according to LDA criteria.

✔ 2nd approach: overall coefficients

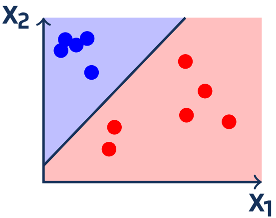
✔ choice of m not very critical

PCA-LDA: Stability of Loadings

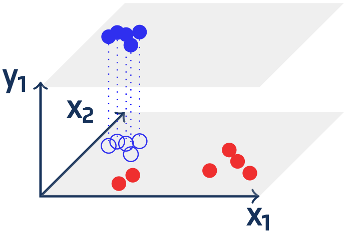


Qualitative Analysis

Dummy Regression



continuous $\longrightarrow$ dummy variable



$$Y = \begin{bmatrix} A \\ \vdots \\ B \end{bmatrix}$$

$$Y = \begin{array}{c} \begin{array}{cc} \text{class A} & \text{class B} \end{array} \\ \begin{bmatrix} 1 & 0 \\ \vdots & \vdots \\ 0 & 1 \end{bmatrix} \end{array}$$

Partial Least Squares Discriminan Analysis PLS-DA

Bilinear model: $\mathbf{Y}^{(n \times m)} = \mathbf{X}^{(n \times p)} \mathbf{B}^{(p \times m)} + \varepsilon$
dependent variable $\mathbf{Y}$ known

Ideas: Do a PCA both on $\mathbf{X}$ and $\mathbf{Y}$.
Rotate $\mathbf{X}$- and $\mathbf{Y}$-scores onto each other.
I.e. do a simultaneous decomposition of $\mathbf{X}$ and $\mathbf{Y}$:

$$\mathbf{X}^{(n \times p)} = \mathbf{T}^{(n \times m)} \mathbf{P}^{(m \times p)} + \mathbf{E}$$

$$\mathbf{Y}^{(n \times m)} = \mathbf{T}^{(n \times m)} \mathbf{C}^{(m \times p)} + \mathbf{F}$$

- loadings not orthogonal: $\mathbf{T} = \mathbf{XW}$

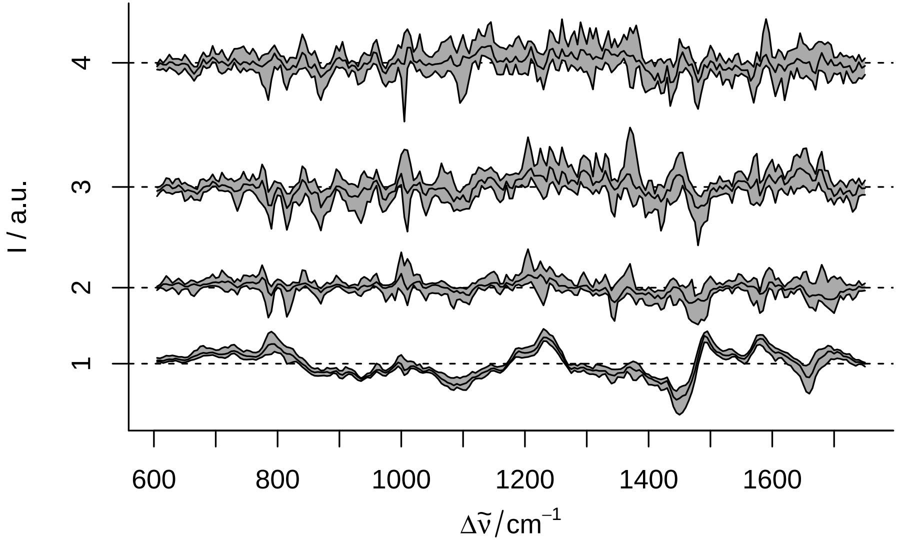
⇒ both $\mathbf{X}$-loadings $\mathbf{P}$ and coefficients $\mathbf{B}$ similar to difference spectra
scores are centered as well

⇒ Coefficients $\mathbf{B}$ are unique

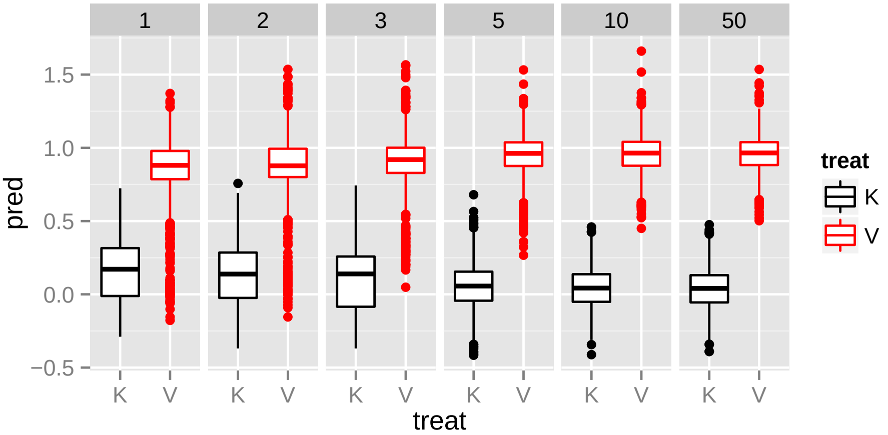
⇒ most important components first

⇒ chemical rank:
1st m components reconstruct concentrations, higher components noise.

PLS-DA: Stability of Loadings



PLS-DA Predictions



✔ 1^{th} st latent variables separate classes

✘ overfitting: higher components desparately try to compress spectra onto exactly 0 and exactly 1

⇒ choice of *m* critical

PLS-LDA

Bilinear model: $$\begin{aligned}\mathbf{L}^{(n\times k-1)} &= \mathbf{T}^{(n\times m)}\mathbf{B}'^{(m\times k-1)} + \varepsilon \\ &= \mathbf{X}^{(n\times p)}\mathbf{W}^{(p\times m)}\mathbf{B}'^{(m\times k-1)} \\ &= \mathbf{X}^{(n\times p)}\mathbf{B}''^{(p\times k-1)}\end{aligned}$$

Idea: Do a PCA first and then LDA on the PCA scores

! careful about details: both PLS and LDA center data according to their own criteria

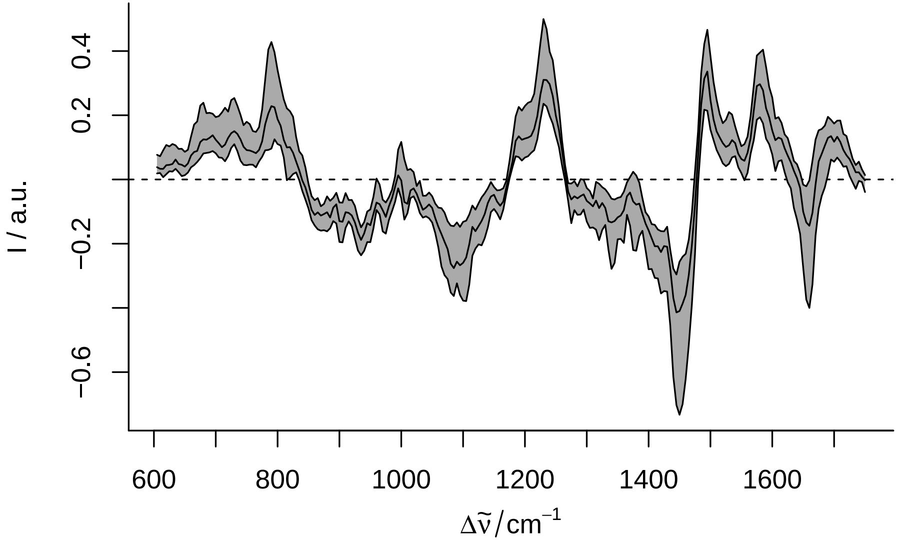
⇒ either do one step after the other (no $\mathbf{B}''$),

⇒ or ensure centering for PLS according to LDA criteria.

✔ 2nd approach: overall coefficients

✔ choice of *m* not very critical

PLS-LDA: Stability of Loadings



Equivalent Models

PCA invariant to flipping
possibly also to rotation within the chosen components
⇒ Procrustes rotation

PLS (-DA) scores and loadings invariant to flipping
possibly also to rotation within the chosen components (Procrustes)
✔ coefficients defined

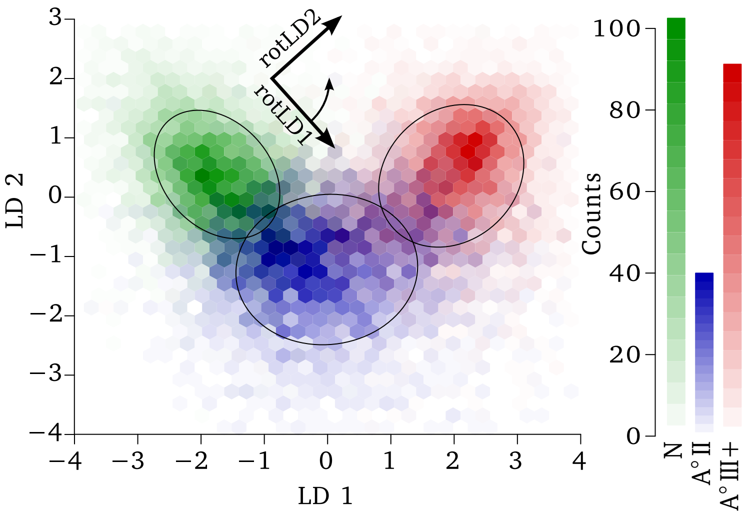
LDA predictions invariant to flipping, rotation, translation
✔ rotate class means onto each other,
align with axes for easier interpretation
✔ applies also to PCA-LDA, PLS-LDA

Spectroscopic Model Interpretation

Keep in mind:

- Coefficients are *no* difference spectra!
  - Difference spectra will always show correlation
    e.g. $\nu_s C - H_2$, $\nu_{as} C - H_2$, $\delta C - H_2$
  - Coefficients may only use one band but omit another
- Coefficients may probe baseline
  e.g. a coefficient pattern $-\frac{1}{2} I(\lambda_1)$, $+I(\lambda_2)$, $-\frac{1}{2} I(\lambda_3)$:
  intensity of band at λ_2 baseline corrected with baseline $I(\lambda_1) - I(\lambda_3)$
- compare to literature findings
- LDA, PCA-LDA, PLS-LDA: rotate for easier interpretation

Rotated LDA Model



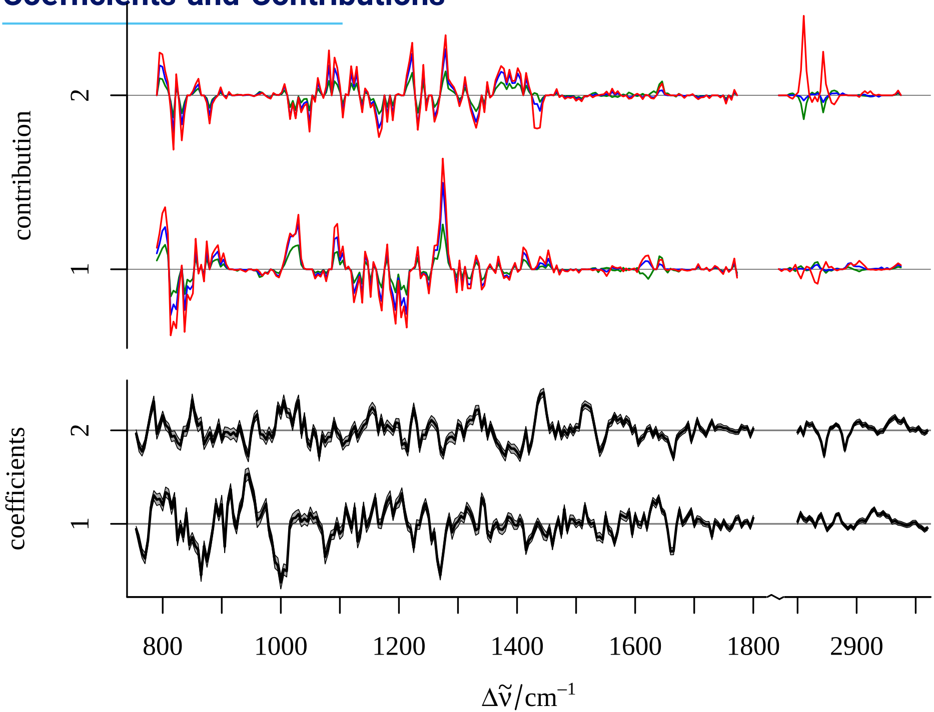
Coefficients and Contributions

- in general, large coefficients ⇒ important
- spectroscopic data: small intensity · large coefficient = importance

Calculation of score for 1 spectrum

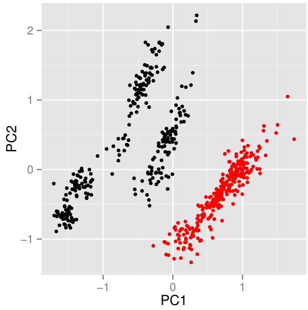
1. multiply each intensity by its coefficient
2. sum up

Coefficients and Contributions



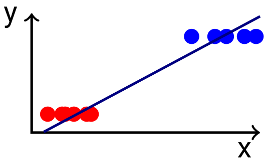
CHEMO METRIX

Focus on Boundaries



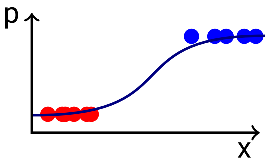
- **LDA:** uses *all* cases equivalently
- **Logistic Regression:** focus on cases close to class boundary, downweight cases far away from boundary
- **Support Vector Machines:**
  use *only* cases that define class boundary

Logistic Regression



- plain dummy regression predicts $y \in (-\infty, +\infty)$

Logistic Regression

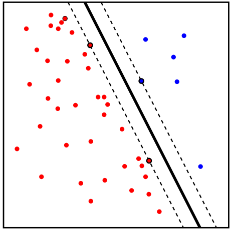


- plain dummy regression predicts $y \in (-\infty, +\infty)$
- logistic function $f(x) = \frac{1}{1+e^{-x}}$
  scales $(-\infty, +\infty) \mapsto [0, 1]$
- predict class membership probability
- Logistic regression: $L(p) = ln\left(\frac{p}{1-p}\right) = \mathbf{BX} + \varepsilon$

Kernel Methods

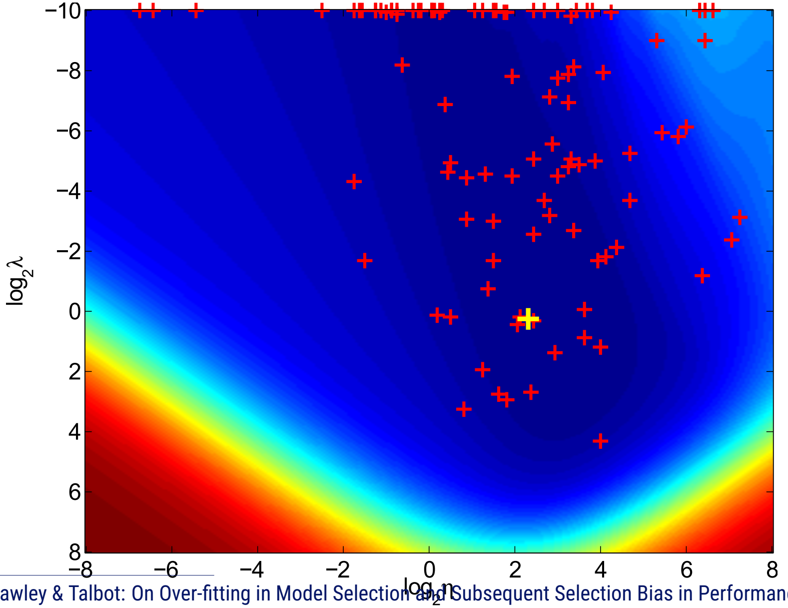
- *R-mode* models: express model in terms of $(\mathbf{X}'\mathbf{X})^{(p \times p)}$
  PCA, PLS, LDA
- *Q-mode* models: express model in terms of $(\mathbf{X}\mathbf{X}')^{(n \times n)}$
  e.g. cluster analysis
- *Kernel* models:
  - need only scalar (dot) product of *spectra* $x_i \cdot x_j$
  - avoid calculation of $n \times n$ matrix of dot products
  - nonlinear models: $\mathit{Kernel}(x_i, x_j) = f(x_i) \cdot f(y_i)$
- Kernel formulation available for many methods
  - ✔ SVM
  - ✔ PCA
  - ✔ PLS
  - ✔ LDA
  - ✔ Logistic Regression

Support Vector Machines



- Select support points from near the class boundary
- put boundary in the middle between those points
- needs linear separability
  - ⇝ use kernel to transform in higher-dimensional space where linear separability is given
  - ⇝ slack variable: allow misclassifications

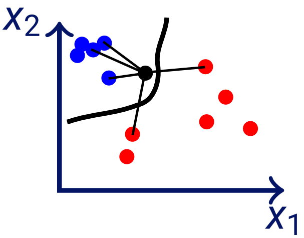
Support Vector Machines: Optimization



Cawley & Talbot: On Over-fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation, Journal of Machine Learning Research 11 (2010) 2079-2107

.

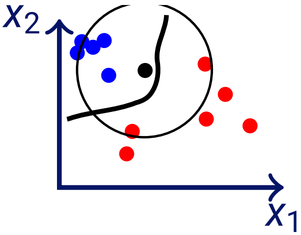
k-Nearest Neighbours (KNN)



- Look up the *k* training points closest to point in question
- Assign to majority class
  relative class frequencies of neighbours ⇝ membership probability
- hyperparameter *k*

✘ prediction time consuming

k-Nearest Neighbours (KNN)



- Look up the *k* training points closest to point in question
- Assign to majority class
  relative class frequencies of neighbours ⇝ membership probability
- hyperparameter *k*

✘ prediction time consuming

Summary

- classification predicts group membership
- Types of classification tasks:
  - discriminative: there is really no possibility outside the specified classes
  - a threshold is put on a metric variable, e.g. limit on analyte concentration
  - one-class: classes are idendependent of each other, may overlap. A case may belong to none of the classes.
- LDA, PCA-LDA, PLS-LDA
- Discriminant methods: Focusing on all cases vs. boundary cases only: LDA, LR, SVM