# Latent Variable Methods Course

## Learning from data

Instructor: Kevin Dunn
kevin.dunn@connectmv.com
http://connectmv.com



Revision: 268:adfd compiled on 15-12-2011

# Copyright, sharing, and attribution notice

We appreciate:

- if you let us know about **any errors** in the slides
- **any suggestions to improve the notes**
- telling us if you use the slides, especially commercially, so we can inform you of major updates
- emailing us to ask about different licensing terms

All of the above can be done by writing us at

`courses@connectmv.com`

If reporting errors/updates, please quote the current revision number: `268:adfd`

# Projects I

- Preferably combine it with your research (2 for 1)
  - Chapter/section of your thesis
  - Alternative way of looking at an existing data set
- Theoretical investigation
  - Cross-validation (e.g. data randomization)
  - Missing data handling alternatives
  - Robust PCA and PLS
  - Adaptive PCA and PLS (handles drift, disturbances)
  - Orthogonal signal correction (OSC)
- Many data sets on the internet; freely available
  - Kaggle.com data analysis competitions (win some money!)
    - Prediction credit score
    - Predict if a car will be a "kick" (bad purchase)
    - Predict when supermarket shoppers will next visit and how much they will spend

# Projects II

- Your own data is always the most interesting. Some ideas:
  - Image analysis data: identifying defects reliably
  - Soft sensor development (e.g. distillation column). Open- *vs* closed-loop
  - Multiblock data analysis (e.g. lab data from multiple steps/instruments)
  - Control system performance: data from closed-loop systems to determine if performance has degraded
  - QSAR: review literatures and compare alternative approaches
  - Financial data: some examples freely available online.

- 1 page outline of ideas: 4 November, or earlier (email is OK)
- Class presentations of 15 minutes: 9 and 16 December 2011
- Report
  - printed version and PDF version
  - Due 9 January 2012 (tentative)
  - No more than 25 pages, all included.

# Presentation expectations

- Should clearly state objectives
- Describe why you have selected preprocessing
- Any special pre-treatment to the data?
- Why PCA and/or PLS is appropriate to achieving your objective
- What was learned that was new?
- How was objective achieved with the model

- 12 minutes of slides
- 8 minutes of questions

# Presentation dates

**9 December**

- Cheng
- Mudassir
- Harry
- Matthew
- Sharleen
- Caroline
- Ran
- Jake

**16 December**

- Brandon
- Yasser
- Rummana
- Lily
- Yanan
- Pavan
- Abdul

# Two-blocks instead of one

*Discussion on the board*

* categorical variables * process measurements * raw material properties from certificates of analysi
Y: quality of product (continuous measurements) outcome from a process (good/OK/bad) concentration values from a sensory panel

# Review: Covariance

| | Cylinder temperature (K) | Cylinder pressure (kPa) | Room humidity (%) |
|---|---|---|---|
| | 273 | 1600 | 42 |
| | 285 | 1670 | 48 |
| | 297 | 1730 | 45 |
| | 309 | 1830 | 49 |
| | 321 | 1880 | 41 |
| | 333 | 1920 | 46 |
| | 345 | 2000 | 48 |
| | 357 | 2100 | 48 |
| | 369 | 2170 | 45 |
| | 381 | 2200 | 49 |
| **Mean** | 327 | 1910 | 46.1 |
| **Variance** | 1188 | 38940 | 7.3 |

# Review: Covariance

**Formal definition for covariance**

$$\text{Cov}\{x, y\} = \mathcal{E}\{(x - \overline{x})(y - \overline{y})\} \qquad \text{where} \qquad \mathcal{E}\{z\} = \overline{z}$$

- Covariance with itself = variance:
  $\text{Cov}\{x, x\} = \mathcal{V}(x) = \mathcal{E}\{(x - \overline{x})(x - \overline{x})\}$
- (Co)variance of centered vector = (co)variance of uncentered vector
- Covariance describes overall tendency of 2 variables

# Review: Covariance

**Formal definition for covariance**

$$\text{Cov}\left\{x, y\right\} = \mathcal{E}\left\{(x - \overline{x})(y - \overline{y})\right\} \qquad \text{where} \qquad \mathcal{E}\left\{z\right\} = \overline{z}$$

Covariance matrix for example:

- variances are on the diagonal
- covariances on the off-diagonals (symmetric matrix!)

$$\text{Covariance} = \begin{bmatrix} & \text{Temperature} & \text{Pressure} & \text{Humidity} \\ \text{Temperature} & 1188 & 6780 & 35.4 \\ \text{Pressure} & 6780 & 38940 & 202 \\ \text{Humidity} & 35.4 & 202 & 7.3 \end{bmatrix}$$

# Review: Correlation

- (Co)variance depends on units: e.g. different covariance for grams vs kilograms
- Correlation removes the scaling effect:

**Formal definition for correlation**

$$r(x, y) = \frac{\mathcal{E}\left\{(x - \overline{x})(y - \overline{y})\right\}}{\sqrt{\mathcal{V}\left\{x\right\}\mathcal{V}\left\{y\right\}}} = \frac{\text{Cov}\left\{x, y\right\}}{\sqrt{\mathcal{V}\left\{x\right\}\mathcal{V}\left\{y\right\}}}$$

- Divides by the units of $x$ and $y$: dimensionless result
- $-1 \leq r(x, y) \leq 1$

$$\text{Correlation} = \begin{bmatrix} & \text{Temperature} & \text{Pressure} & \text{Humidity} \\ \text{Temperature} & 1.0 & 0.997 & 0.380 \\ \text{Pressure} & 0.997 & 1.0 & 0.379 \\ \text{Humidity} & 0.380 & 0.379 & 1.0 \end{bmatrix}$$

# Review: Least squares

We have 2 vectors of data, $\mathbf{x}$ and $\mathbf{y}$. Presume the relationship between them:

$$\mathbf{y} = \beta_0 + \beta_1 \mathbf{x} + \epsilon$$

$\epsilon$ term:

- unmodelled components of the linear model
- measurement error
- other random variation

**Important**: error is from $y$, not from $x$.

We want parameter estimates:

- $b_0 = \hat{\beta}_0$
- $b_1 = \hat{\beta}_1$
- $e = \hat{\epsilon}$
-

# Review: Least squares

To make derivations easier here, we will center both $\mathbf{x}$ and $\mathbf{y}$.

Least squares model is: $\mathbf{y} = \beta_1 \mathbf{x} + \epsilon$

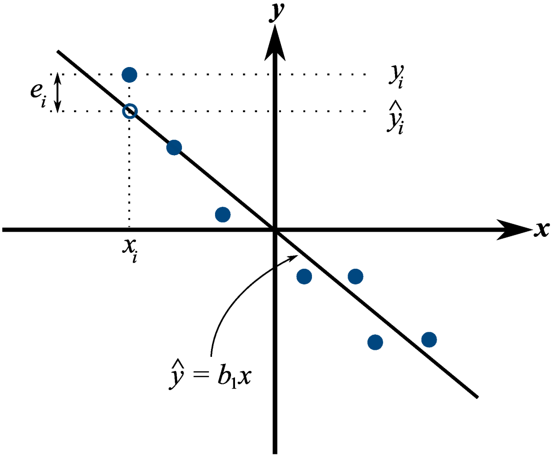
We can always recover the intercept, if we need it:

- $b_0 = \overline{\mathbf{y}} - b_1 \overline{\mathbf{x}}$

We want predictions from our model:

- For a new $x$-observation: $x_{\text{new}}$
- prediction is $= \hat{y}_{\text{new}} = b_1 x_{\text{new}}$

# Review: Least squares



# Review: solving the least squares model

Has to be an optimization problem: **minimizing** the sum of squared errors

- Easy to solve! Unconstrained optimization problem

$$\min f(b_1) = \sum_{i=1}^{n} (e_i)^2 = \sum_{i=1}^{n} (y_i - b_1 x_i)^2$$

$$\begin{aligned} \frac{\partial f(b_1)}{\partial b_1} &= -2 \textstyle\sum_i^n (x_i)(y_i - b_1 x_i) = 0 \\ b_1 &= \frac{\sum_i (x_i y_i)}{\sum_i (x_i)^2} = \frac{\mathbf{x}'\mathbf{y}}{\mathbf{x}'\mathbf{x}} \end{aligned}$$

# Remarks

1. $\sum_i e_i = 0$
2. Easily prove that $\sum_i (x_i e_i) = \mathbf{x}^T \mathbf{e} = 0$
   - The residuals are uncorrelated with the input variables, $\mathbf{x}$
   - There is no information in the residuals that is in the $\mathbf{x}$'s
3. Prove and interpret that $\sum_i (\hat{y}_i e_i) = \hat{\mathbf{y}}^T \mathbf{e} = 0$
   - The fitted values are uncorrelated with the residuals

# Notation for MLR

The general linear model for observation $i$

$$
\begin{aligned}
y_i &= \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_K x_K + \epsilon_i \\
y_i &= [x_1, x_2, \ldots, x_K] \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_K \end{bmatrix} + \epsilon_i \\
y_i &= \underbrace{x^T}_{(1\times K)} \underbrace{\beta}_{(K\times 1)} + \epsilon_i
\end{aligned}
$$

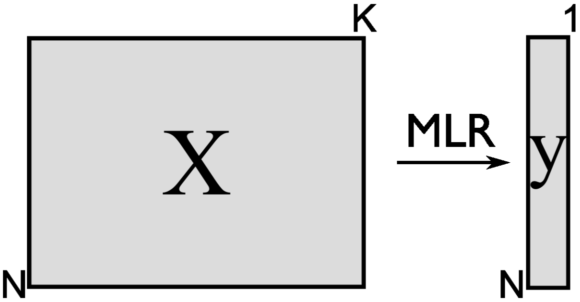
- where each $x_k$ column (variable) and the $y$ column have been centered

# Notation for MLR

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,K} \\ x_{2,1} & x_{2,2} & \dots & x_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N,1} & x_{N,2} & \dots & x_{N,K} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_K \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix}$$

$$\mathbf{y} = \mathbf{X}\mathbf{b} + \mathbf{e}$$

- $\mathbf{y}$: $N \times 1$
- $\mathbf{X}$: $N \times k$
- $\mathbf{b}$: $K \times 1$
- $\mathbf{e}$: $N \times 1$



# Estimating the model parameters via optimization

**Objective function**: minimize sum of squares of the errors

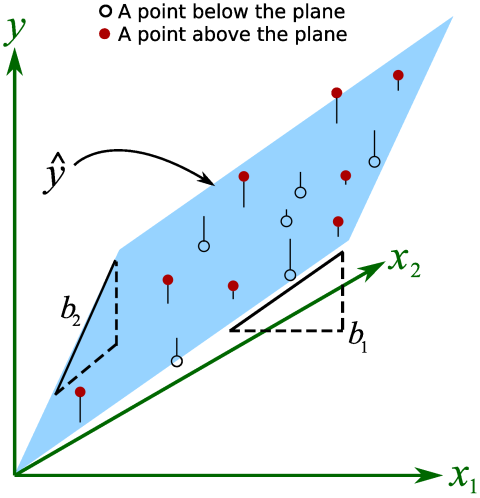
$$
\begin{aligned}
f(\mathbf{b}) &= \mathbf{e}^T\mathbf{e} \\
&= (\mathbf{y} - \mathbf{X}\mathbf{b})^T(\mathbf{y} - \mathbf{X}\mathbf{b}) \\
&= \mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T\mathbf{X}\mathbf{b} + \mathbf{b}\mathbf{X}^T\mathbf{X}\mathbf{b}
\end{aligned}
$$

- Solving $\dfrac{f(\mathbf{b})}{\partial \mathbf{b}} = 0$ gives $\boxed{\mathbf{b} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}}$
- $\mathcal{V}(\mathbf{b}) = (\mathbf{X}'\mathbf{X})^{-1} S_E^2$
- $S_E = \sqrt{\dfrac{\mathbf{e}'\mathbf{e}}{N-K}} \approx$ standard deviation of the residuals

# Interpretation of the model coefficients

## The coefficients have meaning

$$y = b_1 x_1 + b_2 x_2$$

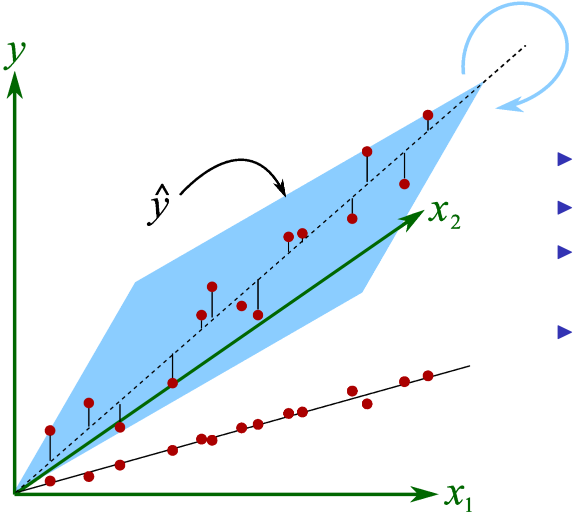


# Least squares: What can go wrong?

1. Missing values
   - $\hat{y}_{\text{new}} = b_1 x_{1,\text{new}} + b_2 x_{2,\text{new}} + \ldots + b_K x_{K,\text{new}}$
   - There is nothing we can do if any $x_{k,\text{new}}$ terms go missing

# Least squares: What can go wrong?

2. Highly correlated variables in $\mathbf{X}$




- $\mathbf{b} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$
- $\mathcal{V}(\mathbf{b}) = \left(\mathbf{X}'\mathbf{X}\right)^{-1} S_E^2$
- Inflated confidence intervals for **b**
- Cannot interpret coefficients reliably

Leads to unstable regression coefficients. *Example on your own.*

# Least squares: What can go wrong?

3. Noisy **x**-variables

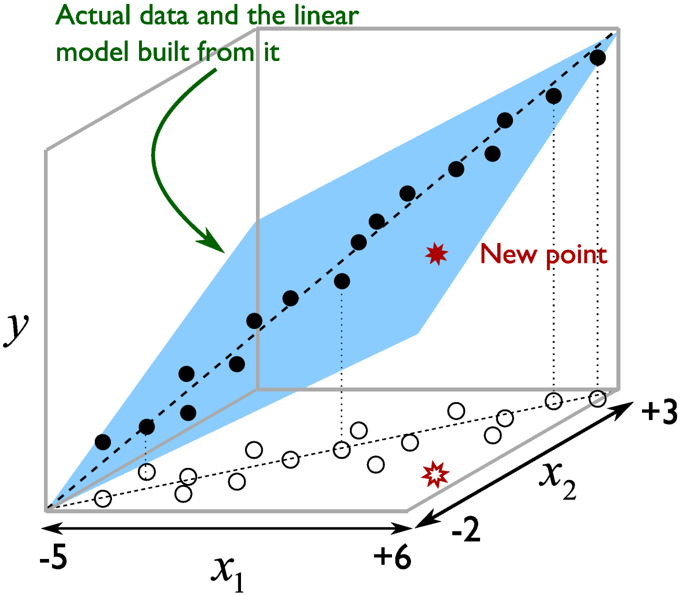
- LS model is: $\mathbf{y} = \beta_1 \mathbf{x} + \epsilon$
- Note that model assumes error in **y**.
- We say, "LS has a model for error" in the **y**'s.
- Or alternatively, "model for error in the **y**-space". This means:
  - We can always compare our $y$ error to $S_E$
  - see if error is large; then try to find out why
- LS assumes that **x** is exact (no model for **x**-space error)

# Least squares: What can go wrong?

4. Non-sensical input (related to previous point)
   - Extreme noise in **x**'s, or garbage input
   - Will go undetected, and you will always get a prediction:
   - $\hat{y}_{\text{new}} = b_1 x_{1,\text{new}} + b_2 x_{2,\text{new}} + \ldots + b_K x_{K,\text{new}}$
   - There is no **x**-space error model to catch these problems

# Least squares: What can go wrong?

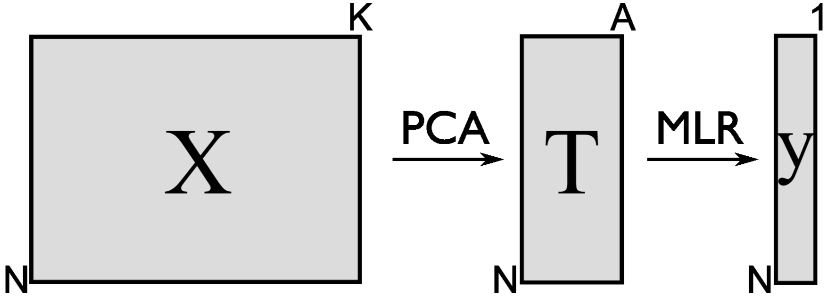
Misleading strategy that's often-used by people:



# Other problems with linear regression

- MLR requires $N > K$. Problem with spectral data, and other data sets.
- If you have multiple $\mathbf{Y}$ variables: one MLR model per column in $\mathbf{Y}$

# Principal component regression (PCR)




Two step model:

1. $\mathbf{T} = \mathbf{XP} + \mathbf{E}$ ordinary PCA
2. $\widehat{\mathbf{y}} = \mathbf{Tb}$ and can be solved as $\mathbf{b} = \left(\mathbf{T}'\mathbf{T}\right)^{-1}\mathbf{T}'\mathbf{y}$
   Regress the **y** onto the scores **T** to get regression coefficients **b**

# Principal component regression (PCR)

Advantages:

- $\mathbf{T}$ is orthogonal: $(\mathbf{T}'\mathbf{T})^{-1}$ easily calculated
- so less need for variable selection to get a full rank $\mathbf{X}$
- PCA step handles missing values
- $\mathbf{T}$ has much less error than $\mathbf{X}$
- **Best part**: a free consistency check from $T^2$ and SPE
- PCA step uses fewer variables ($A < K$), we will likely meet the $N > K$ requirement in the regression step

**Important point**

If PCA step uses $A = K$, then predictions from PCR are same as MLR

# Principal component regression (PCR)

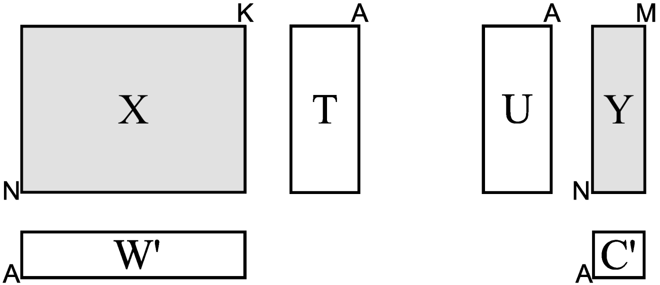
Using a PCR model on new data

1. Center and scale the raw data as usual for PCA: $\mathbf{x}'_{\text{new}}$
2. Calculate the new scores: $\mathbf{t}'_{\text{new}} = \mathbf{x}'_{\text{new}}\mathbf{P}$
3. Consistency check: are $\text{SPE}_{\text{new}}$ and $T^2_{\text{new}}$ below the limits?
4. Use the MLR prediction: $\hat{y}_{\text{new}} = \mathbf{t}'_{\text{new}}\mathbf{b}$

# PCR: disadvantages

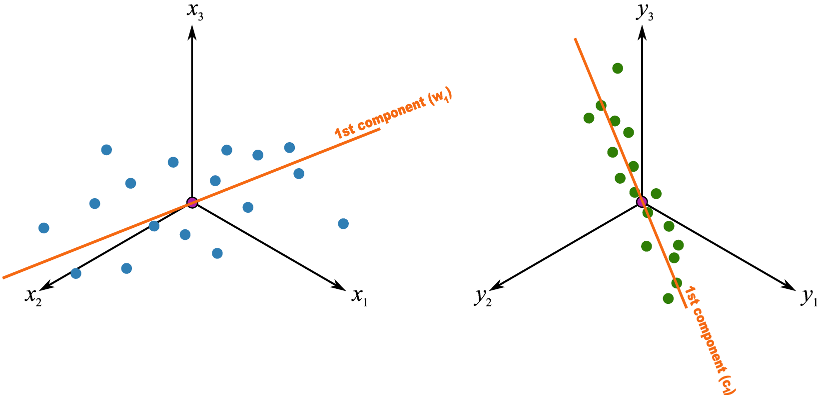
1. PCA components calculated without knowledge of **y**
   - not necessarily predictive of **y**
   - because steps 1 and steps 2 are performed sequentially
2. As a result, we often need to add additional, noisy components in PCA step
   - Add components beyond usual cross-validation

# Simple PLS (SIMPLS)

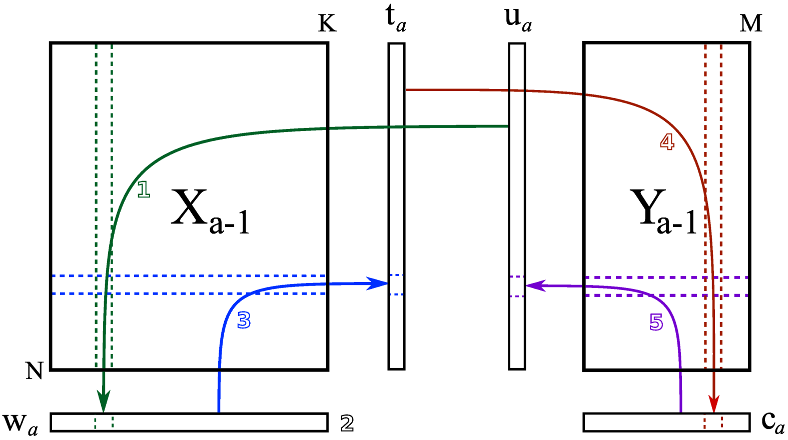



1. PLS scores explain $\mathbf{X}$:
   - $\mathbf{t}_a = \mathbf{X}_a \mathbf{w}_a$ for the $\mathbf{X}$-space
   - $\max : \mathbf{t}_a' \mathbf{t}_a$ subject to $\mathbf{w}_a' \mathbf{w}_a = 1.0$
2. PLS scores also explain $\mathbf{Y}$:
   - $\mathbf{u}_a = \mathbf{Y}_a \mathbf{c}_a$ for the $\mathbf{Y}$-space
   - $\max : \mathbf{u}_a' \mathbf{u}_a$ subject to $\mathbf{c}_a' \mathbf{c}_a = 1.0$
3. PLS maximizes relationship between $\mathbf{X}$- and $\mathbf{Y}$-space
   - maximizes covariance: $\mathrm{Cov}\left(\mathbf{t}_a, \mathbf{u}_a\right)$
   - $\mathrm{Cov}\left(\mathbf{t}_a, \mathbf{u}_a\right) = \mathrm{Corr}\left(\mathbf{t}_a, \mathbf{u}_a\right) \cdot \sqrt{\mathbf{t}_a' \mathbf{t}_a} \cdot \sqrt{\mathbf{u}_a' \mathbf{u}_a} \cdot \frac{1}{N}$

# PLS: geometric interpretation



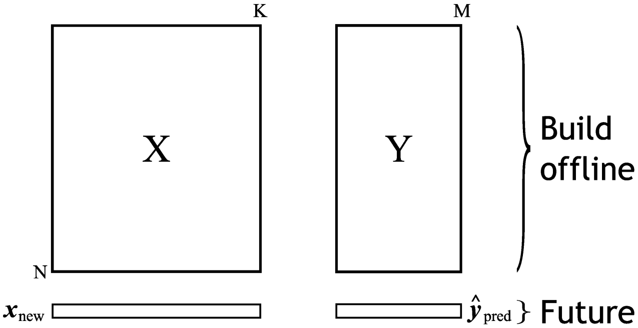
# NIPALS algorithm for PLS



# The weights in PLS

- Scores are calculated from deflated matrices:
  - $\mathbf{t}_1 = \mathbf{X}_{a=0}\mathbf{w}_1 = \mathbf{X}_0\mathbf{w}_1$
  - $\mathbf{t}_2 = \mathbf{X}_{a=1}\mathbf{w}_2 = (\mathbf{X}_0 - \mathbf{t}_1\mathbf{p}_1)\,\mathbf{w}_2$
- $\mathbf{w}_2$: relates score $\mathbf{t}_2$ to $\mathbf{X}_{a=1}$, the deflated matrix
- This is hard to interpret. We would like instead:
  - $\mathbf{t}_1 = \mathbf{X}_{a=0}\mathbf{w}{*}_1 = \mathbf{X}_0\mathbf{w}{*}_1$
  - $\mathbf{t}_2 = \mathbf{X}_{a=0}\mathbf{w}{*}_2 = \mathbf{X}_0\mathbf{w}{*}_2$
  - *etc*
- We calculate matrix $\mathbf{W}^* = \mathbf{W}\left(\mathbf{P}'\mathbf{W}\right)^{-1}$
- So $\mathbf{T} = \mathbf{X}_0\mathbf{W}^*$, or simply: $\boxed{\mathbf{T} = \mathbf{X}\mathbf{W}^*}$
  - $\mathbf{w}{*}_1 = \mathbf{w}_1$
  - $\mathbf{w}{*}_a \neq \mathbf{w}_a$ for $a > 1$
- We get a clearer interpretation of the variable relationships using $\mathbf{W}^*$ instead of $\mathbf{W}$

# Using PLS on new data

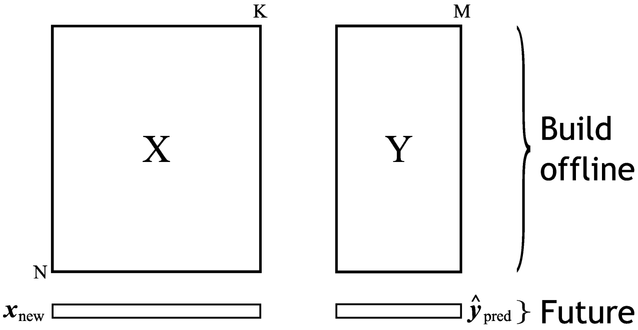



$$
\begin{aligned}
t_{1,\text{new}} &= \mathbf{x}'_{\text{new}}\mathbf{w}_1 \\
\mathbf{x}'_{\text{new}} &= \mathbf{x}'_{\text{new}} - t_{1,\text{new}}\mathbf{p}'_1 \qquad \text{(deflate)} \\
t_{2,\text{new}} &= \mathbf{x}'_{\text{new}}\mathbf{w}_2 \\
\mathbf{x}'_{\text{new}} &= \mathbf{x}'_{\text{new}} - t_{2,\text{new}}\mathbf{p}'_2 \\
\textit{etc} &
\end{aligned}
$$

Collect all the $t_{a,\text{new}}$ score values in $\mathbf{t}_{\text{new}}$

Alternatively use $\mathbf{t}_{\text{new}} = \mathbf{x}'_{\text{new}}\mathbf{W}^*$ to get $\mathbf{t}_{\text{new}}$ without deflation

# Using PLS on new data



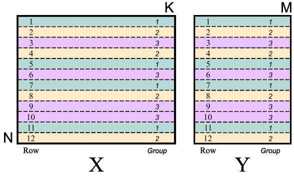

$$
\begin{aligned}
\widehat{\mathbf{y}}'_{\text{new}} &= \mathbf{t}'_{\text{new}}\mathbf{C}' \\
\widehat{\mathbf{y}}'_{\text{new}} &= \mathbf{x}'_{\text{new}}\mathbf{W}^*\mathbf{C}'
\end{aligned}
$$

- Then uncenter and unscale the $\widehat{\mathbf{y}}'_{\text{new}}$

# Cross-validation to calculate $Q^2$

Similar procedure as with PCA

Split the rows in $\mathbf{X}$ and $\mathbf{Y}$ into $G$ groups.

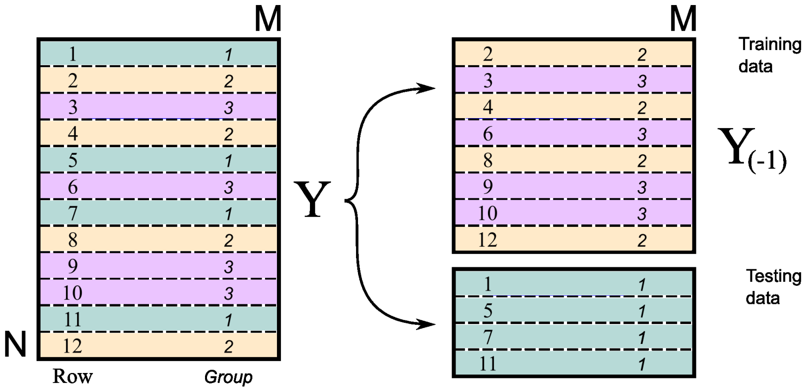



$G = 3$ in this illustration

- Typically $G \approx 7$ [ProSensus, Simca-P use $G = 7$]
- Rows can be randomly grouped, or
- ordered *e.g.* 1, 2, 3, 1, 2, 3, ...
- ordered *e.g.* 1, 1, 2, 2, 3, 3, ...

# Cross-validation concept for PLS

Fit a PLS model using $\mathbf{X}_{(-1)}$ and $\mathbf{Y}_{(-1)}$; use $\mathbf{X}_{(1)}$ as testing data



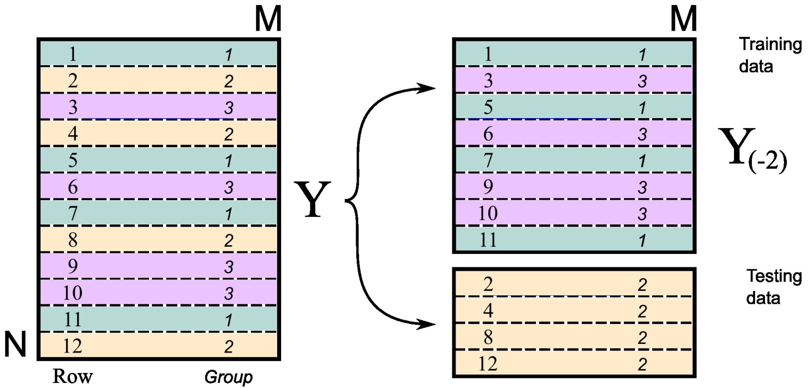

Split the X-matrix along the same rows, but only calculate PRESS using F matrix.

$$F_{(1)} = Y_{(1)} - \hat{Y}_{(1)}$$

$\mathbf{F}_{(1)}$ = prediction error for testing group 1

# Cross-validation concept for PLS

Fit a PLS model using $\mathbf{X}_{(-2)}$ and $\mathbf{Y}_{(-2)}$; use $\mathbf{X}_{(2)}$ as testing data



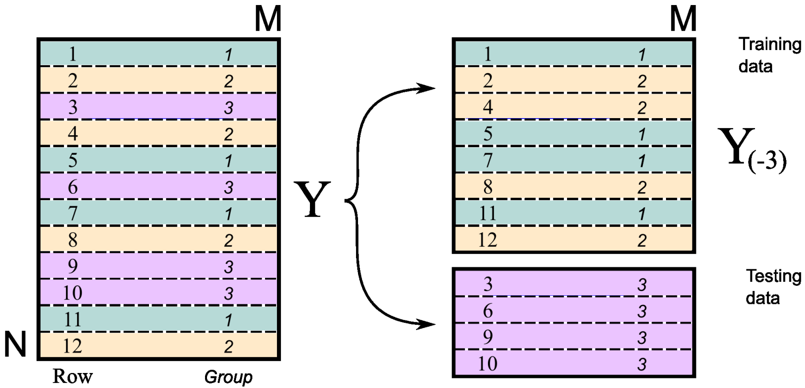

Split the X-matrix along the same rows, but only calculate PRESS using F matrix.

$$F_{(2)} = Y_{(2)} - \hat{Y}_{(2)}$$

$\mathbf{F}_{(2)}$ = prediction error for testing group 2

# Cross-validation concept for PLS

Fit a PLS model using $\mathbf{X}_{(-3)}$ and $\mathbf{Y}_{(-3)}$; use $\mathbf{X}_{(3)}$ as testing data




Split the X-matrix along the same rows, but only calculate PRESS using F matrix.

$$F_{(3)} = Y_{(3)} - \hat{Y}_{(3)}$$

$\mathbf{F}_{(3)}$ = prediction error for testing group 3

# Cross-validation concept for PLS

- PRESS $= \text{ssq}(\mathbf{F}_{(1)}) + \text{ssq}(\mathbf{F}_{(2)}) + \ldots + \text{ssq}(\mathbf{F}_{(G)})$
- PRESS = prediction error sum of squares from each prediction group
- $Q^2 = 1 - \dfrac{\mathcal{V}(\text{predicted } \mathbf{F}_A)}{\mathcal{V}(\mathbf{Y})} = 1 - \dfrac{\text{PRESS}}{\mathcal{V}(\mathbf{Y})}$
- $Q^2$ is calculated and interpreted in the same way as $R^2$
- $Q^2_k$ can be calculated for variable $k = 1, 2, \ldots K$
- You should always find $Q^2 \leq R^2$
- If $Q^2 \approx R^2$: that component is useful and predictive in the model
- If $Q^2$ is "small": that component is likely fitting noise

To read: Esbensen and Geladi, 2010, "Principles of proper validation"

# PLS plots

- Score plots: $\mathbf{t}$ and $\mathbf{u}$ show relationship between rows
- Weight plots: $\mathbf{w}$: relationship between $\mathbf{X}$ columns
- Loading plots: $\mathbf{c}$: relationship between $\mathbf{Y}$ variables
- Weight and loading plots: $\mathbf{w}^*\mathbf{c}$: relationship between $\mathbf{X}$ and $\mathbf{Y}$
- SPE plots (X-space, Y-space)
- Hotelling's $T^2$ plot
- Coefficient plots
- VIP plot
- $R^2$ plots (X-space, Y-space, per variable)

# Variable importance to prediction

Important variables in the model?

- Have large (absolute) weights: why?
- Come from a component that has a high $R^2$

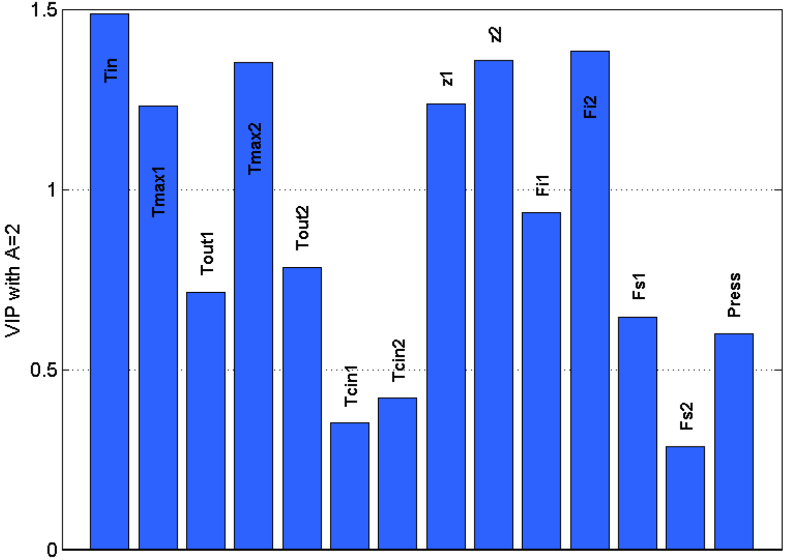
Combining these two concepts we calculate *for each variable*:

**Importance of variable $k$ using $A$ components in PLS**

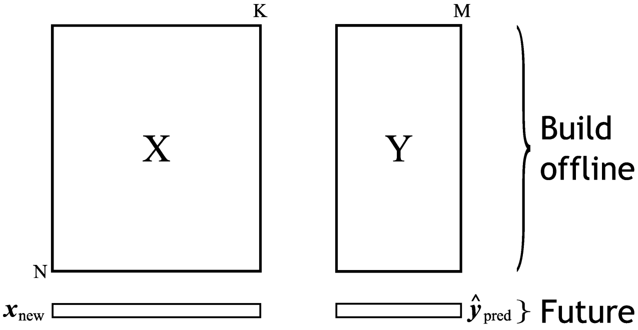
$$VIP^{\mathbf{2}}_{A,k} = \frac{K}{\mathrm{SSX}_0 - \mathrm{SSX}_A} \cdot \sum_{a=1}^{A} \left(\mathrm{SSX}_{a-1} - \mathrm{SSX}_a\right) W^2_{a,k}$$

- $SSX_a$ = sum of squares in the $\mathbf{X}$ matrix after $a$ components
- $\frac{\mathrm{SSX}_{a-1} - \mathrm{SSX}_a}{SSX_A}$ = incremental $R^2$ for $a^{\text{th}}$ component
- $\frac{\mathrm{SSX}_0 - \mathrm{SSX}_A}{SSX_A} = R^2$ for model using $A$ components
- Messy, but you can show that $\sum_k VIP^2_{A,k} = K$
- Reasonable cut-off = 1
- VIP for PCA models: use $P^2_{a,k}$ instead of $W^2_{a,k}$

# Variable importance to prediction



# Coefficient plot

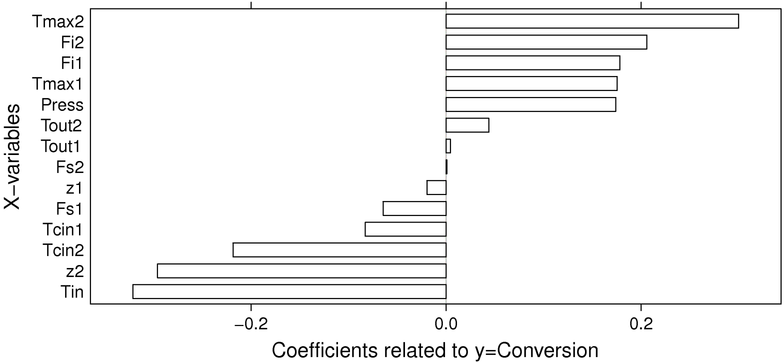



$$
\begin{aligned}
\widehat{\mathbf{y}}'_{\text{new}} &= \mathbf{t}'_{\text{new}}\mathbf{C}' \\
\widehat{\mathbf{y}}'_{\text{new}} &= \mathbf{x}'_{\text{new}}\mathbf{W}^{*}\mathbf{C}' \\
\widehat{\mathbf{y}}'_{\text{new}} &= \mathbf{x}'_{\text{new}}\boldsymbol{\beta}
\end{aligned}
$$

- $\boldsymbol{\beta}$ is a $K \times M$ matrix
- Each column in $\boldsymbol{\beta}$ contains the regression coefficients for column $m$ from $\mathbf{Y}$ matrix
- **Never implement PLS using $\boldsymbol{\beta}$ matrix**

# Coefficient plot

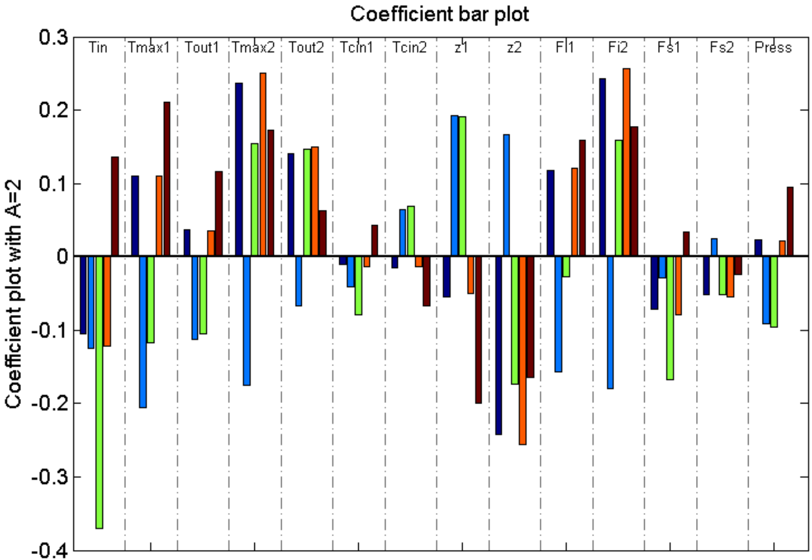
For a single $y$-variable:




- $\hat{y} = \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_K x_K$
- where $x_k$ and $\hat{y}$ are the preprocessed values
- *Again* – never implement PLS this way.

# Coefficient plot

For all **Y**-variables



# Jackknifing

We re-calculate the model $G + 1$ times during cross-validation:

- $G$ times, once per group
- The "+1" is from the final round, where we use **all** observations

We get $G + 1$ estimates of the model parameters:

- loadings
- VIP values
- coefficients

for every variable $(1, 2, \ldots K)$.

Calculate "reliability intervals" (don't call them confidence intervals)

- Martens and Martens (paper 43) describe jackknifing.
- Efron and Tibshirani describe the bootstrap and jackknife.