MSc+ day Chemometrics – 21/03/2015 Computer Course

This course consists of three parts:

1. Making and interpreting a biplot of the Wine data
2. Alignment of chromatograms and inspection via PCA
3. Running and interpreting the POCHEMON algorithm on simulated data

All three sections will be covered using MATLAB, the standard platform for chemometrics research.

1. Biplots of wine data

This course starts with constructing biplots of the 'Wine data'. This classical data set, often used in data analysis courses, was originally recorded to trace the origin of specific Italian wines—covered by a D.O.C, increasing the price for the wine—through measuring some standard 'quality parameters' of the wine.

The data consists of:

- Wines of three different cultivars (Barolo, Grignolino and Barbera)
- Measurements on 13 different quality parameters:
  1. Alcohol
  2. Malic acid
  3. Ash
  4. Alkalinity of ash
  5. Magnesium
  6. Total phenols
  7. Flavonoids
  8. Nonflavonoid phenols
  9. Proanthocyanins
  10. Color intensity
  11. Hue
  12. OD280/OD315 of diluted wines (OD = optical density)
  13. Proline

Load the 'wine' data by issuing the following command on the command line in MATLAB, while in the '1. Biplots' folder:

```
load Data
```

1. First, we concentrate on one cultivar: Grignolino. This data is in matrix Xgr. This matrix has been 'autoscaled'. Autoscaling basically consists of two operations. Find out what autoscaling is, and why it is a sensible operation in this context. Make sure to separately describe both aspects of autoscaling.

2. Make a PCA model of these Grignolino wines, by issuing the following commands:

```
[Ld_gr, Sc_gr, latent] = princomp(Xgr);
Pex_gr = latent/sum(latent)*100;
clear latent
```

Now you have a PCA model, plot the scores and loadings:

```
figure
plot(Sc_gr(:,1), Sc_gr(:,2), 'b<');
hold on
quiver(zeros(13,1), zeros(13,1), Ld_gr(:,1), Ld_gr(:,2))
text(Ld_gr(:,1), Ld_gr(:,2), Data.Variables)
```

Look at your figure: what is wrong? How could you improve the visibility of the information? Try to adapt the previous four lines of code. If you have a figure you are happy about, you can add titles and axislabels with commands `title`, `xlabel` and `ylabel`.

3. Explain in detail the information that this figure contains: describe the scores, the loadings and their relationship.

4. Using the previous commands, make the same figure for Barolo wines, this data is contained in matrix *Xbo*. Also interpret the information in this figure, analogous to question 3.

5. Now compare the model of Grignolino to that of Barolo wines, using the two earlier biplots. Think about comparisons that you cannot make.

6. Now make a comparison between all three cultivars, using the following commands:

```
[Ld_all, Sc_all, latent] = princomp(Xall);
plot(Sc_all(igr,1), Sc_all(igr,2), 'b<');
hold on
plot(Sc_all(ibo,1), Sc_all(ibo,2), 'gx');
plot(Sc_all(iba,1), Sc_all(iba,2), 'r*');
```

and finally:

```
quiver(zeros(13,1), zeros(13,1), Ld_all(:,1), Ld_all(:,2))
text(Ld_all(:,1), Ld_all(:,2), Data.Variables)
```

with the correct modifications for visibility, comparable to that performed in question 1.

Discuss the information in this figure, how is this different from the information in the figures from questions 1 and 2? Concerning the information from the first two questions, is this reflected in the answer to question 3? Why (not)?

2. Alignment of chromatograms

The next subject of this course is a small introduction to signal alignment. We are going to investigate 16 chromatograms—that have been taken on consecutive days—of one sample. In MATLAB, navigate to the '2. Alignment' folder.

1. Make a plot of the raw data by issuing

```
Q1
```

on the command line. The 16 chromatograms are colored from red to blue (i.e. red is chromatogram 1 and blue is chromatogram 16). Can you explain the order that you see? (You can also zoom in on a specific part of the chromatograms using the zoom button on top of the figure.)

2. Construct a PCA scoreplot by issuing the command

```
Q2
```

on the command line. Does the figure correspond with your answer to question 1? Can you explain what you see?

3. We will now apply the COW algorithm (Correlation Optimized Warping) to the 16 chromatograms and plot the results. Do this by typing

```
Q3
```

on the command line in MATLAB. This will take approximately 30 seconds. One of the chromatograms in the middle of the time series (chromatogram 8) is used as a reference. The slack and segment length have been set to 10 and 50, respectively. Visually inspect the result; again, the coloring from red to blue indicates the time series. What do you think of the result? (You can try playing with the settings for segment length and slack by opening the file Q3.m, changing the values that you want to change, saving the file and re-executing Q3!)

4. Finally, we will again apply PCA, but now the aligned data. Do this by issuing

```
Q4
```

at the command line. Compare the two scoreplots of the data before and after alignment. Can you explain the differences? What do PCA models of the unaligned and (perfectly) aligned data focus on? Can you explain the huge reduction in explained variance after alignment?

3. POCHEMON on simulated data

We simulate a competition experiment on five variables (*e.g.* metabolites) that were measured on 10 replicates, for a monoculture 1, a monoculture 2 and a co-culture 12, in matrices X1, X2 and X12 respectively. To obtain these, issue

```
do_data
```

on the MATLAB command line while in the '3. POCHEMON' directory. This part of the exercises requires some more programming from your side.

1. Make a 'training' PCA model of both monocultures; do you want to mean-center? How many PCs will you need for this model? Use the following commands to build and plot a non-meancentered PCA model:

```
[sc_tr, ld_tr, latent]=pca([X1; X2], 0, [], 2);
hold on
plot(sc_tr(1:10,1), sc_tr(1:10,2), 'bo')
plot(sc_tr(11:20,1), sc_tr(11:20,2), 'g*')
quiver(zeros(5,1), zeros(5,1), ld_tr(:,1), ld_tr(:,2))
text(ld_tr(:,1), ld_tr(:,2), {'1', '2', '3', '4', '5'})
title('Non-meancentered data')
```

and the following commands to plot a mean-centered PCA model:

```
X1m = X1 - repmat(mean(X1), 10, 1);
X2m = X2 - repmat(mean(X2), 10, 1);
[sc_trm, ld_trm, latentm]=pca([X1m; X2m], 0, [], 2);
hold on
plot(sc_trm(1:10,1), sc_trm(1:10,2), 'bo')
plot(sc_trm(11:20,1), sc_trm(11:20,2), 'g*')
quiver(zeros(5,1), zeros(5,1), ld_trm(:,1), ld_trm(:,2))
text(ld_trm(:,1), ld_trm(:,2), {'1', '2', '3', '4', '5'})
title('Meancentered data')
```

to build the PCA model. Inspect the biplots of the mean-centered and non-meancentered data and interpret what you see.

2. Project the co-culture data onto the resulting loadings, this can be done by the following command:

```
sc_co_tr = X12*ld_tr;
```

and plot the resulting co-culture scores onto the training model:

```
figure
hold on
plot(sc_tr(1:10,1), sc_tr(1:10,2), 'bo')
plot(sc_tr(11:20,1), sc_tr(11:20,2), 'g*')
plot(sc_co_tr(:,1), sc_co_tr(:,2), 'r<')
quiver(zeros(5,1), zeros(5,1), ld_tr(:,1), ld_tr(:,2))
text(ld_tr(:,1), ld_tr(:,2), {'1', '2', '3', '4', '5'})
```

How do you interpret this?

3. Calculate the competition-specific information from the data and the model and call it `E_co_tr`:

```
E_co_tr = X12 - sc_co_tr*ld_tr';
figure; bar(sum(E_co_tr.^2,2),'g');
```

The bar plot shows the model residuals for each of the 10 co-cultures. In which co-culture sample is the competition-specific information largest?

4. Fit another PCA model on this `E_co_tr`: how many PCs do you need here? Plot and interpret the results of this model. Try programming this yourself based on the previous commands!

5. **BONUS** question: how can you establish whether the competition effect observed in question 3 is 'significant'?