Department of Political Science, University of Aarhus Tel.: 89 42 12 71
Universitetsparken, DK-8000 Århus C Fax.: 86 13 98 39
Søren Risbjerg Thomsen E-mail:srt@ps.au.dk

February 2006

Models for Multi-Level Voting Behaviour

Exercises, Session 7 – Cluster analysis of aggregate party choice

In session 6 we investigated the following approximate relations in equations (6.15) and (6.16)

$$x_{gj} = a_{oj} + a_{1j}\mu_{g1} + a_{2j}\mu_{g2} + \cdots + a_{Kj}\mu_{gK} \qquad \text{(6.15, repeated)}$$

$$y_{gj} = b_{oj} + b_{1j}\mu_{g1} + b_{2j}\mu_{g2} + \cdots + b_{Kj}\mu_{gK}\,, \qquad \text{(6.16, repeated)}$$

where x_{gj} is the logit transformed share of votes for party j in district g at the first election and y_{gj} is the same variable at the second election.

We found that we could estimate the most important latent variables using factor analysis. We hinted on page 6.2 that these relationships might be different from one political region to another, which we shall explore further in this session using cluster analysis. Since it is unrealistic to assume that we can completely predict the aggregate outcome in each district from the latent variables we first add a random component to each equation as shown in equation (7.1) and (7.2),

$$x_{gj} = a_{oj} + a_{1j}\mu_{g1} + a_{2j}\mu_{g2} + \cdots + a_{Kj}\mu_{gK} + u_{gj} \qquad (7.1)$$

$$y_{gj} = b_{oj} + b_{1j}\mu_{g1} + b_{2j}\mu_{g2} + \cdots + b_{Kj}\mu_{gK} + v_{gj}\,. \qquad (7.2)$$

The random variation can for example be caused by random local political events. We assume that these events are happening independent of each other and that the random variation is also independent of the latent variables and of the same magnitude for a certain party at both elections. Thus we assume that

$$Var(u_{*j}) = Var(v_{*j}) = \sigma_j^2 \qquad (7.3)$$

i.e., the unexplained variance (across districts) is the same for party j at both elections and

$$Cov(u_{*j}, v_{*j}) = 0\,, \qquad (7.4)$$

i.e., the covariance between the random components for the party is equal to 0.
[Deviate the model where a_1=b_1, a_2 = b_2, etc. and show the graph for this model]
It is common in electoral research to analyze the aggregate choice at the second election as a regression function of the aggregate choice at the first election. We will first do so using the

same data file as in Session 6 and by adapting the do-file DKdis01.do to a new do-file DKdis02.do. We keep the commands in DKdis01.do that compute party percentages and logits for the 1998 and the 2001 Danish Parliament election.

As a first inspection of the change of support for the Socialist People's Party, we draw in a scattergram the logit support for the party in 2001 against the logit support in 1998. Table 7.1 shows all the first commands for data preparation in DKdis02.do including the command for drawing the graph.

Table 7.1 Commands for data preparation and simple scattergram

```
* DKdis02.do - Exercises, Set 7
* Cluster analysis of change in aggregate party choice
* Adapted from DKdis01.do

use "C:\Data\DKdistricts.dta", clear

* Preparing data
*****************

* Inspect DP98
*sum dpoe98-votdp98

* Compute percent of all votes for all parties
foreach v of varlist dpoe98 dpf98 dpu98 dpa98 dpb98 dpd98 dpq98 dpc98 /*
  */ dpv98 dpo98 dpz98 xdp98 spldp98 absdp98 {
gen p`v' = `v'/votdp98*100
  }

* Weighted percentages
sum pdpoe98-pabsdp98 [aw=votdp98]

* Compute logit shares for all parties
foreach v of varlist dpoe98 dpf98 dpu98 dpa98 dpb98 dpd98 dpq98 dpc98 /*
  */ dpv98 dpo98 dpz98 xdp98 spldp98 absdp98 {
gen l`v' = ln(p`v'/(100-p`v'))
  }
sum ldpoe98-labsdp98 [aw=votdp98]

* Compute percentages and logits for DP 2001
*******************************************

* Inspect DP01
*sum dpoe01-votdp01

* Compute percentages for all parties
foreach v of varlist dpoe01 dpf01 dpa01 dpb01 dpd01 dpq01 dpc01 /*
  */ dpv01 dpo01 dpz01 xdp01 spldp01 absdp01 {
gen p`v' = `v'/votdp01*100
  }
sum pdpoe01-pabsdp01 [aw=votdp01]

* Compute logits for all parties
foreach v of varlist dpoe01 dpf01 dpa01 dpb01 dpd01 dpq01 dpc01 /*
  */ dpv01 dpo01 dpz01 xdp01 spldp01 absdp01 {
gen l`v' = ln(p`v'/(100-p`v'))
  }
sum ldpoe01-labsdp01 [aw=votdp01]

* Factor analysis
*****************

*obliraw ldpoe98-ldpz98 labsdp98 ldpoe01-ldpz01 labsdp01 [aw=votdp01],
factors(2)

*greigen

* Study of party swing
```

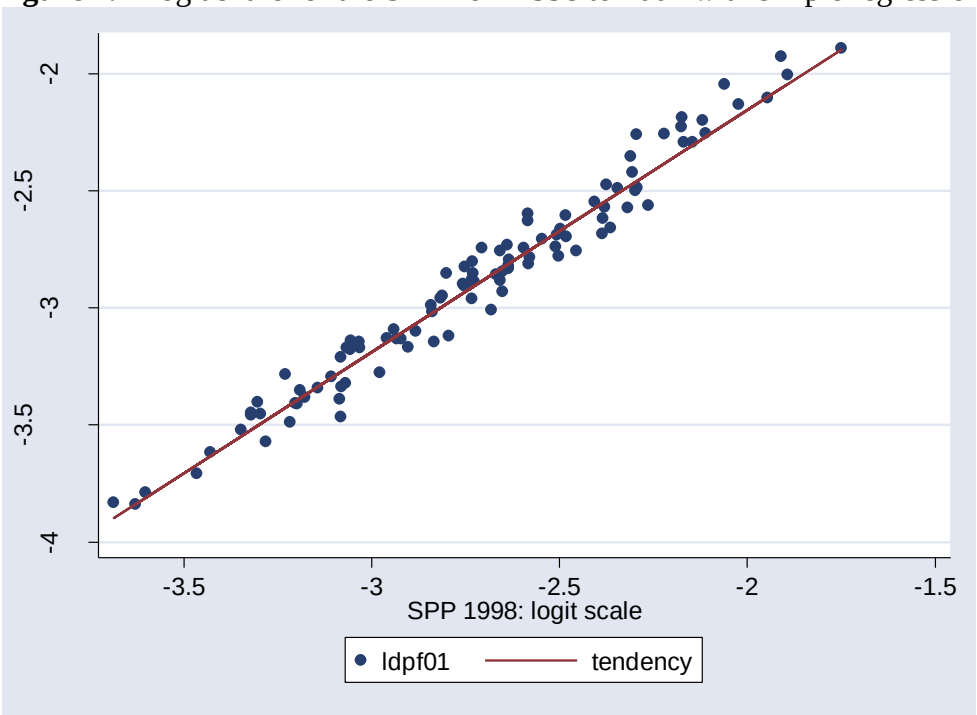
```
* Socialist People's Party (SPP) from 1998 to 2001

* Regression Analysis
**********************
pause on

*Scatter diagram
scatter ldpf01 ldpf98, xtitle(SPP 1998: logit scale) ytitle(SPP 2001:
logit scale)
pause

*Enter q to continue
```

Notice that the commands for the factor analysis are deactivated. After the text “Regression analysis” we write

pause on

The explanation is that we are going to display several scattergrams, and when pause is set on it is possible to pause temporary after each graph has been shown when running the whole file. The pause is initiated when ever the command pause is written in the do-file. The first time is just after the first scattergram. The execution of the commands continues again when the user enters q in the Stata Command window. Figure 7.1 shows the scattergram with explanatory text on each axis.

Figure 7.1 Logit share for the SPP from 1998 to 2001 with simple regression line



To draw the regression line in Figure 7.1 one must first do simple regression analysis. Table 7.2 shows the commands for doing the regression analysis and drawing the scattergram with the regression line.

Table 7.2 Commands for drawing scattergram with simple regression line

```
* Simple regression analysis weighted by unit size
regress ldpf01 ldpf98 [aweight=votdp01 ]
```

```
* Get coefficients
matrix coefs = e(b)
gen a = coefs[1,2]
gen b = coefs[1,1]
gen tendency = a + b*ldpf98

* Draw scatter diagram
scatter ldpf01 ldpf98, xtitle(SPP 1998: logit scale) ytitle(SPP 2001: logit scale)
/*
*/ || line tendency ldpf98
pause
drop a b tendency
```

The regression analysis is weighted by the number of voters in each district. The intercept (a) and slope (b) are read from coefficient matrix (e) in memory and used to compute the expected line or tendency. The line is added to the scattergram by adding

|| line tendency ldpf98

to the scatter command. Table 7.3 shows the results from the regression analysis.

Table 7.3 Results from simple regression analysis

```
. regress ldpf01 ldpf98 [aweight=votdp01 ]
(sum of wgt is   3.9990e+06)

      Source |       SS       df       MS              Number of obs =     103
-------------+------------------------------           F(  1,   101) = 2449.39
       Model |  15.0877923     1  15.0877923           Prob > F      =  0.0000
    Residual |  .622141166   101  .006159814           R-squared     =  0.9604
-------------+------------------------------           Adj R-squared =  0.9600
       Total |  15.7099335   102  .154018956           Root MSE      =  .07848

------------------------------------------------------------------------------
      ldpf01 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      ldpf98 |   1.033752   .0208875    49.49   0.000     .9923171    1.075188
       _cons |  -.0876413   .0575759    -1.52   0.131    -.2018563    .0265738
------------------------------------------------------------------------------
```

Notice that while the number of voters is used as weight, Stata keeps track of the number of observations (103) to make the test of significance realistic.

The problem is however, that one important assumption for doing the regression analysis is not fulfilled. It is the assumption that there must be no error or random variation in the independent variable, and according to equation (7.1) this is not the case.

Fortunately, Stata can do "errors-in-variables" regression, if one can estimate the amount of "noise variance" in the independent variable. Derived from equation (7.3) and (7.4) the noise variance is equal to σ^2 and the reliability defined as

$$\text{reliability} = 1 - \frac{\text{noise variance}}{\text{total variance}}$$

is approximately equal to the Pearson correlation ρ_j between x and y.[1]

[1] The derivation is not shown here. The reliability is exactly equal to the Pearson correlation if $b_1=a_1$, $b_2=a_2$, … , $b_K=a_K$.

$$\rho = \rho(x_{*j}, y_{*j})$$

Table 7.4 shows the commands for doing the errors-in-variables regression shown in Table 7.5 and drawing the scattergram in Figure 7.2 with the line from this regression.

Table 7.4 Commands for drawing scattergram with error-in-variables regression line

```
* Preparing errors-in-variables regression

* find correlation
correlate ldpf98 ldpf01 [aweight=votdp01 ]

* Error-in-variables regression analysis weighted by unit size
eivreg ldpf01 ldpf98 [aweight=votdp01], reliab(ldpf98 `r(rho)')

* Get coefficients
matrix coefs = e(b)
gen a = coefs[1,2]
gen b = coefs[1,1]
gen tendency = a + b*ldpf98

* Draw scatter diagram
scatter ldpf01 ldpf98, xtitle(SPP 1998: logit scale) ytitle(SPP 2001: logit scale)
/*
*/ || line tendency ldpf98
pause
```

Notice that the reliability for the aggregate choice of SPP in Table 7.5 is very high (0.980) and thus the slope of the regression is just slightly steeper (1.055) than with the simple regression in Table 7.3 (1.034).

Table 7.6 shows commands for additional "publication ready" scattergrams shown in Figure 7.3 and 7.4. An attractive property of Figure 7.3 is that the aspect ratio is about 1 and a grid is introduced so one can better see that the actual slope is close to 1.0. Further, in Figure 7.4 each point is indicated with the region number so one can see that for example Region 1 (the capital Copenhagen) is showing positive deviations from the linear tendency.

The slope obtained by error-in-variables regression is close to one when the coefficients with respect to the latent variables only change slightly from one election to the next, indicating that the issue positions of the parties are stable. A point we will return to later on.

Table 7.5 Results from errors-in-variables regression analysis

```
.
. * Preparing errors-in-variables regression
.
. * find correlation
. correlate ldpf98 ldpf01 [aweight=votdp01 ]
(sum of wgt is   3.9990e+06)
(obs=103)

             |   ldpf98   ldpf01
-------------+------------------
      ldpf98 |   1.0000
      ldpf01 |   0.9800   1.0000

.
. * Error-in-variables regression analysis weighted by unit size
. eivreg ldpf01 ldpf98 [aweight=votdp01], reliab(ldpf98 `r(rho)')

               assumed                              errors-in-variables regression
variable     reliability
------------------------                                Number of obs =     103
  ldpf98       0.9800                                   F(  1,   101) = 4849.79
       *       1.0000                                   Prob > F      =  0.0000
                                                        R-squared     =  0.9800
                                                        Root MSE      = .055777

------------------------------------------------------------------------------
      ldpf01 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      ldpf98 |    1.05485   .0151471    69.64   0.000     1.024803    1.084898
       _cons |  -.0300124   .0417376    -0.72   0.474    -.1128085    .0527838
------------------------------------------------------------------------------
```

Figure 7.2 Logit share for the SPP from 1998 to 2001 with errors-in-variables regression line

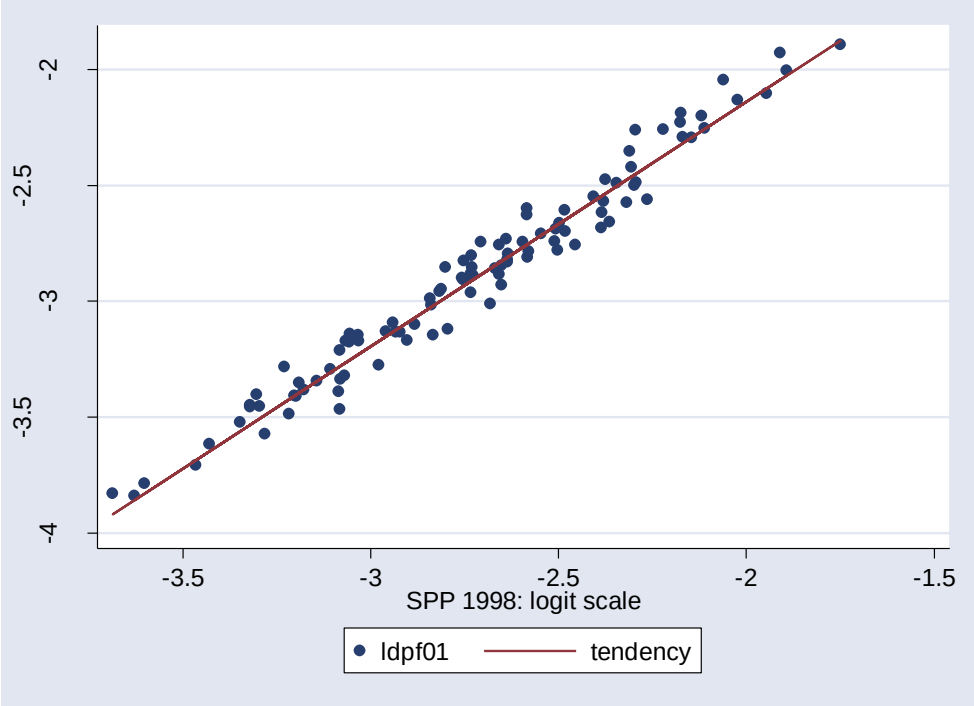


Table 7.6 Commands for drawing additional scattergrams with the errors-in-variables regression line

```
* Draw symmetric scattergram
twoway (scatter ldpf01 ldpf98, sort) (line tendency ldpf98, sort clpat(solid)),/*
*/ ytitle(SPP 2001: logit scale, margin(medsmall)) yscale(range(-4 -1.5))/*
*/ xtitle(SPP 1998: logit scale, margin(medsmall)) xscale(range(-4 -1.5))/*
*/ xlabel(-4(0.5) -1.5, grid) legend(off) ysize(4) xsize(5)/*
*/graphregion(fcolor(white) lcolor(black))
pause

gen region2 = region

* Draw symmetric scattergram with region code
twoway (scatter ldpf01 ldpf98, sort msymbol(none) mlabel(region2)/*
*/ mlabposition(0))(line tendency ldpf98, sort clpat(solid)),/*
*/ ytitle(SPP 2001: logit scale, margin(medsmall)) yscale(range(-4 -1.5))/*
*/ xtitle(SPP 1998: logit scale, margin(medsmall)) xscale(range(-4 -1.5))/*
*/ xlabel(-4(0.5)-1.5, grid) legend(off) ysize(4) xsize(5)/*
*/graphregion(fcolor(white) lcolor(black))
pause

drop a b tendency region2
```

Another example is the change from 1998 to 2001 for the small party CD (Centre Democrats) where the random variation seems to be much wider and errors-in-regression analysis is much more needed (see the Stata commands in the file DKdis02.do). Here the reliability is estimated to be 0.629 and the slope of the regression line changes from 0.872 to 1.385 when errors-in-variables regression analysis is used instead of simple regression analysis. The scattergram with indication of region numbers in Figure 7.5 shows that CD suffered a severe defeat in 2001 but fared much better in Region 1 (Copenhagen) and Region 8 (Northern Jutland) than in the rest of the country.

Figure 7.3 SPP 1998-2001 with errors-in-variables regression line and grid

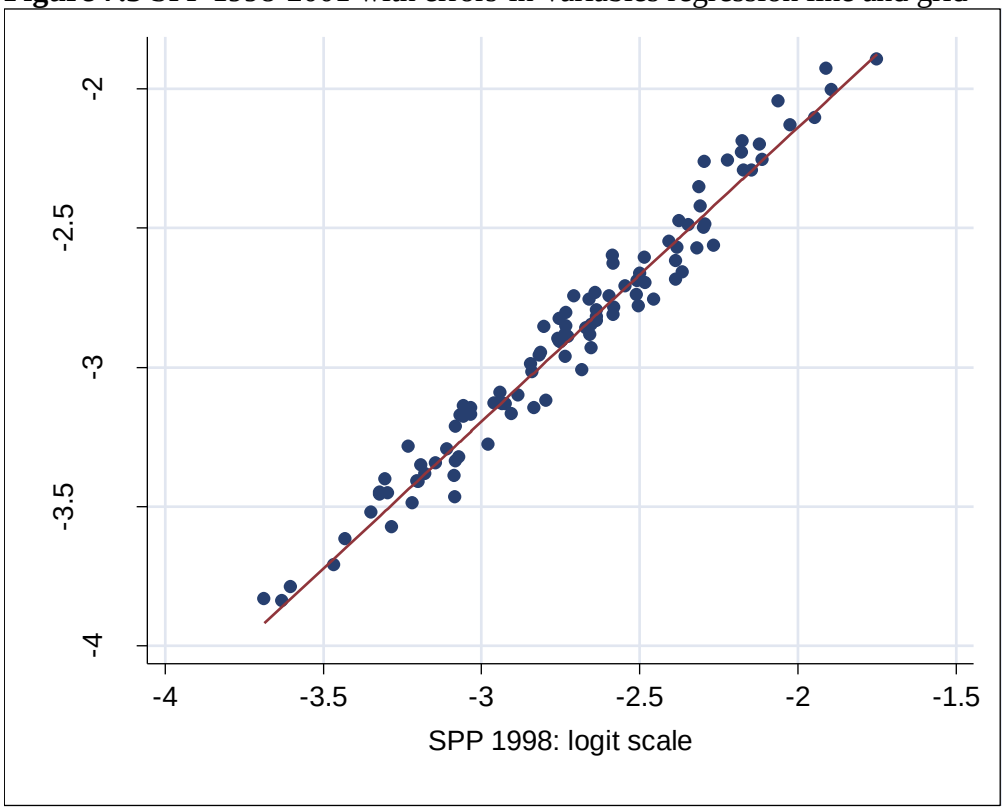


Figure 7.4 SPP 1998-2001 with errors-in-variables regression line, grid, and region numbers

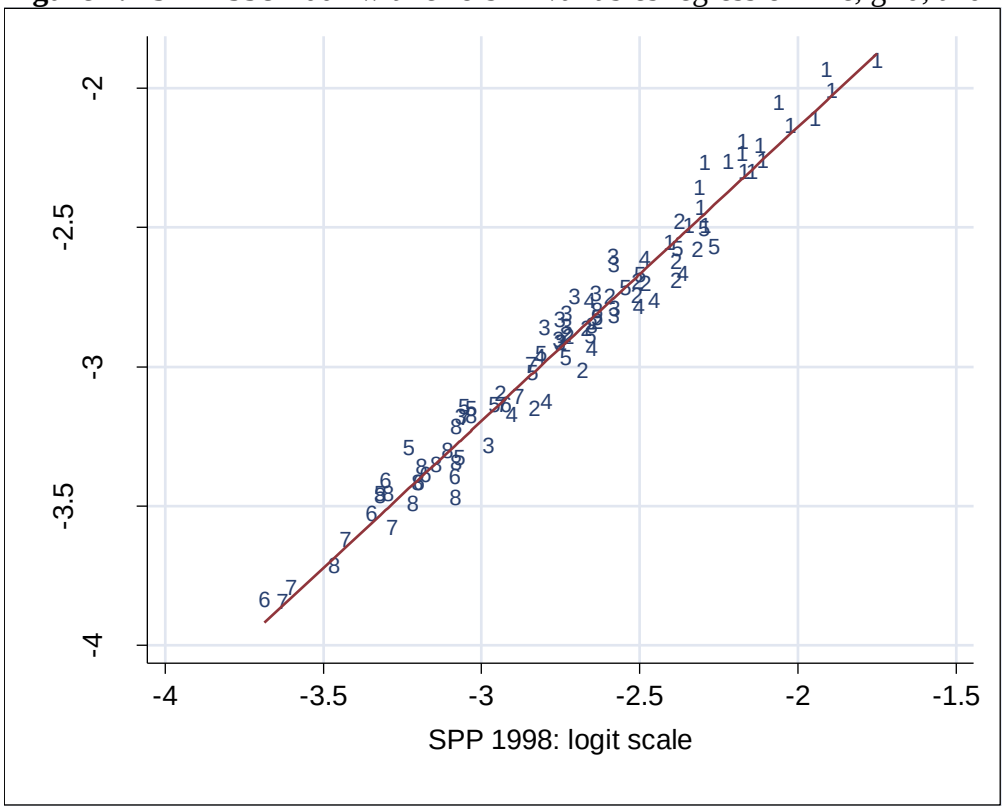
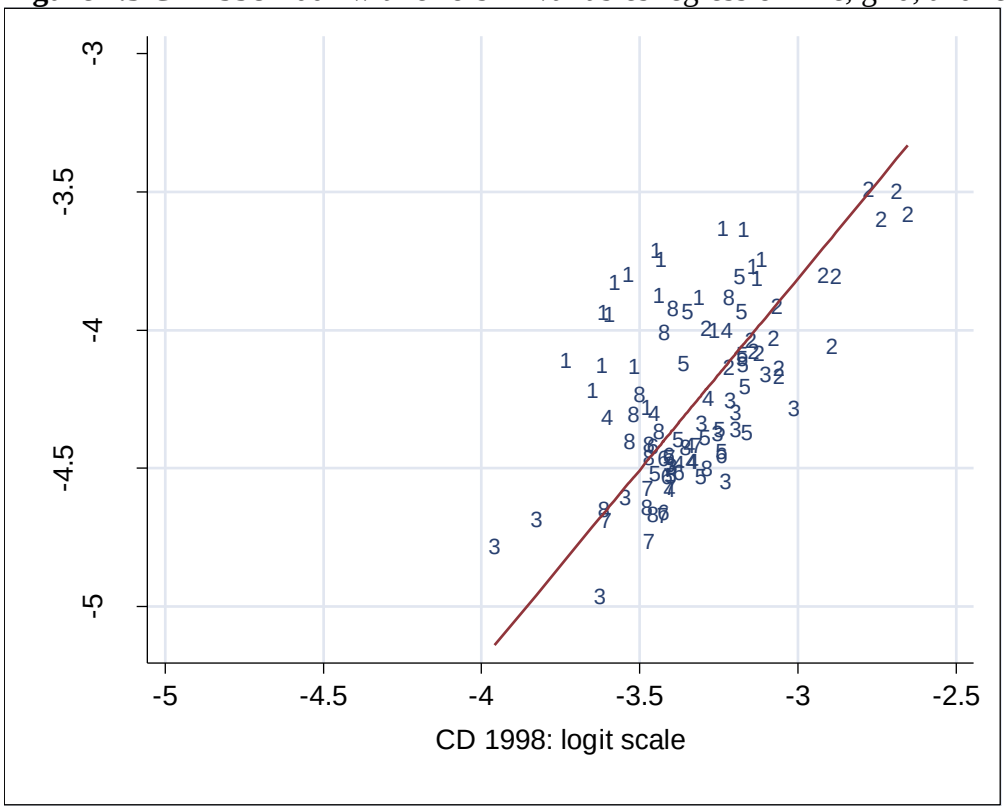


Figure 7.5 CD 1998-2001 with errors-in-variables regression line, grid, and region numbers



Finding homogenous regions with cluster analysis

As discussed in session 6 is it likely that the party constant a_{oj} for "general sympathy" is not the same for the whole country but differs between different political regions with different party strongholds and thus should be denoted a_{oj}^r, where *r* indicates the region (it is not an exponent). Since the party constant in the aggregate model is inferred (in a crude way) from the constant in the individual model one should also expect this party constant to differ between political regions. Thus the equations (7.1) and (7.2) should instead be written

$$x_{gj} = a_{oj}^r + a_{1j}\mu_{g1} + a_{2j}\mu_{g2} + \cdots + a_{Kj}\mu_{gK} + u_{gj} \tag{7.5}$$

$$y_{gj} = b_{oj}^r + b_{1j}\mu_{g1} + b_{2j}\mu_{g2} + \cdots + b_{Kj}\mu_{gK} + v_{gj}\,. \tag{7.6}$$

As mentioned in Thomsen (2000, p. 11) it is a common observation in many countries that the issue positions of the parties are very stable in time while the general sympathy might change dramatically from one election to the next. We further expect that the change in general sympathy might change differently in different regions depending on the regional political culture. Thus, an appropriate model for short time change could be

$$b_1 = a_1,\ \ b_2 = a_2,\ \ldots\ b_K = a_K \tag{7.7}$$

suggesting that the issue positions are constant from one election to the next. From this we can derive that the change z_{gj} in logit share for a party is

$$z_{gj} = y_{gj} - x_{gj} = (b_{oj}^r - a_{oj}^r) + (u_{gj} - v_{gj}) \tag{7.8}$$

i.e., constant across districts within each region apart from random deviations. If this is true one should be able to identify homogenous regions by finding subsets of adjoining districts with a tendency to similar change in party logit shares from election to election.

Cluster analysis is a technique for finding sets of units with similar values on a chosen set of variables. If for example the similarity between two units is measured by the squared difference we should construct clusters that minimize

$$\sum_{g=1}^{G_{r-1}} \sum_{h=g+1}^{G_r} (z_{gj} - z_{hj})^2 \ \ ;\ j = 1, \ldots, m;\ \ r = 1, \ldots, R \tag{7.9}$$

where *R* is the number of regions, *r* is the region number, and G_r is the number of districts in region r. There are many different techniques for constructing such clusters. Ward's technique that stresses the internal homogeneity by minimizing the sum of squares within clusters is recommended. It is also my experience that more homogenous regions are found if the variables in the cluster analysis are standardized (to have zero mean and unit standard deviation) before doing the cluster analysis. The reason is that dramatic changes for just a single party might influence the results too much. This phenomenon is diminished by standardizing the logit change for each party so that all parties have equal importance in the cluster analysis.

Another experience is that more stable regions are found if one includes not only the logit change for each party between two elections, but rather the logit change between several consecutive elections. The reason is that a single very popular or unpopular candidate might single out his or her region between two elections, while this phenomenon is less important when considering several consecutive elections. To illustrate this we will first consider only the change in the Danish elections from 1998 to 2001, and later on we will consider the change in a series of consecutive elections in a longer period. The commands for computing the standardized logit change from 1998 to 2001 for each party are shown in Table 7.7.

Table 7.7 Commands for computing standardized logit change from 1998 to 2001

```
*Cluster analysis
*****************

* Logit change for all parties (except u, x and spl)
egen ddpoe01 = std(ldpoe01 - ldpoe98)
egen ddpf01 = std(ldpf01 - ldpf98)
egen ddpa01 = std(ldpa01 - ldpa98)
egen ddpd01 = std(ldpd01 - ldpd98)
egen ddpq01 = std(ldpq01 - ldpq98)
egen ddpc01 = std(ldpc01 - ldpc98)
egen ddpv01 = std(ldpv01 - ldpv98)
egen ddpo01 = std(ldpo01 - ldpo98)
egen ddpz01 = std(ldpz01 - ldpz98)
egen ddpabs01 = std(labsdp01 - labsdp98)
sum ddpoe01-ddpabs01
```

We are excluding votes for candidates outside the parties and spoiled votes because these groups are of little importance. The logit change for at party is computed as the logit share at the new election minus the logit share at the old elections, and this difference is standardized with the std() function. Table 7.8 shows (unweighted) summary statistics for these variables.

Table 7.8 Summary statistics for standardized logit change 1998-2001

```
. sum ddpoe01-ddpabs01

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
     ddpoe01 |       103    2.42e-09           1   -1.78528     2.6982
      ddpf01 |       103    6.15e-10           1  -2.613741   2.451219
      ddpa01 |       103   -4.59e-09           1  -1.730974   3.462699
      ddpd01 |       103   -9.04e-10           1  -1.594725    2.43525
      ddpq01 |       103           0           1  -2.143264    2.10912
-------------+--------------------------------------------------------
      ddpc01 |       103    2.35e-09           1  -2.619339   2.836639
      ddpv01 |       103   -6.96e-09           1   -2.33616   2.143068
      ddpo01 |       103   -8.66e-10           1  -1.529547   3.157674
      ddpz01 |       103    8.68e-10           1  -3.343606   1.428707
    ddpabs01 |       103   -1.31e-09           1  -2.006603   3.810769
```

Per definition a standardized variable has zero mean and unit standard deviation as also shown in the table (2.42e-09 is the same as 0.00000000242 and thus for all practical means equal to zero). Standardized values greater than 3 or less than -3 indicates outliers, and we notice that Social Democrats (a), Danish People's Party (o), and abstainers (abs) have positive outliers while only the Progressive Party (z) has a negative outlier. To make the cluster analysis one must at least execute two commands. The first command is calculating the distances between the units and saving the necessary information for generating clusters in the file. Here the command

```
Cluster wardslinkage dd*
```

computes the distances between the units as the sum across all variables (with the first two characters equalt to dd) of the squared differences between each pairs of units to secure than one minimizes the within cluster sum of squares (Ward's method). And the command

cluster generate ward = groups(2/12)

creates a series of different solutions with 2, 3, etc. until 12 clusters in the variables ward2, ward3, etc. until ward12. The solutions are hierarchical so that a solution with a certain numbers of clusters can always be grouped to create a solution with a smaller number of clusters. To show the different solutions on a geographical map we use the NSDstat program, version 1.3. Unfortunately, this program can only import version 7 Stata files, so we must save the file including the cluster variables by the command saveold as shown in Table 7.9 (add ", replace" if the file already exist). You should save the file with the name Clusters.dta because the electronic map has the name Clusters.krt (or rename both files).

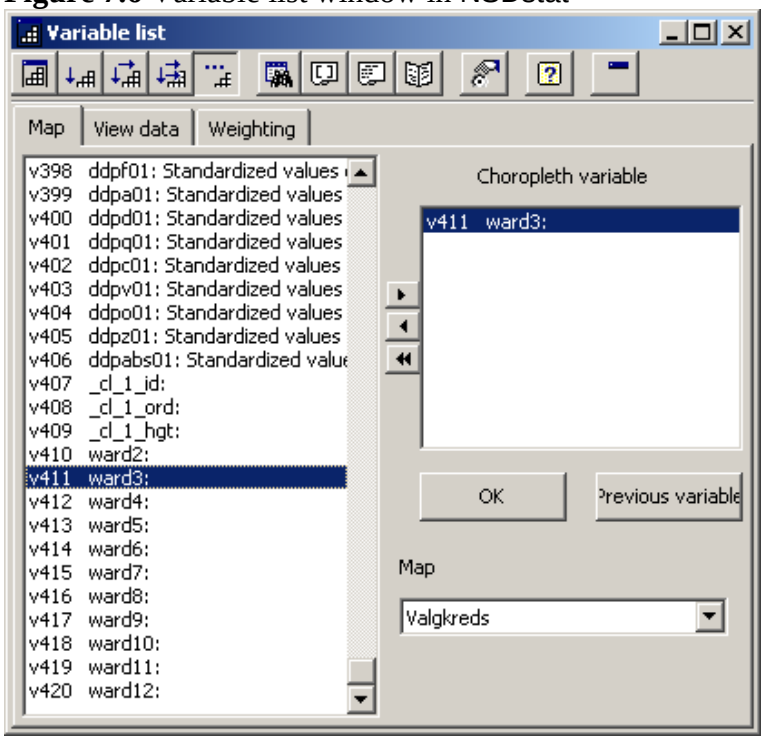
Table 7.9 Commands for making cluster analysis and saving the results in Stata version 7 format

```
cluster wardslinkage dd*
cluster generate ward = groups(2/12)

*saveold "C:\Stata\clusters.dta"
saveold "C:\Stata\clusters.dta", replace
```

When the file Clusters.dta is saved you can leave Stata and start the program NSDstat. The program has two different modules called DataBuilder and DataExplorer and which one of these programs that appear when you start NSDstat depends on what you have done before with the program. If some data (from a previous run) appear in the window of the program you remove the data with the command File| Close.

You can import the Stata file you just created from both the DataBuilder and the DataExplorer by choosing File| import. In the Open file dialog box you select Filetype: Stata (.dta) and opens Clusters.dta. In the Import Stata dialog box you select NSDstat variable labels: Symb. name + Label to keep both the variable names and the variable labels from the Stata file, and press OK. If you are not already in the DataExplorer you go from the DataBuilder to the DataExplorer by choosing File| DataExplorer. A window with the title Variable list similar to Figure 7.6 should now appear. From the list of icons just below the title you press the 5th one (says "other" when you point with the mouse on the icon) and you move the variable v411 ward3: to the field with the heading "Choropleth variable" as shown in Figure 7.6.

Figure 7.6 Variable list window in NSDstat

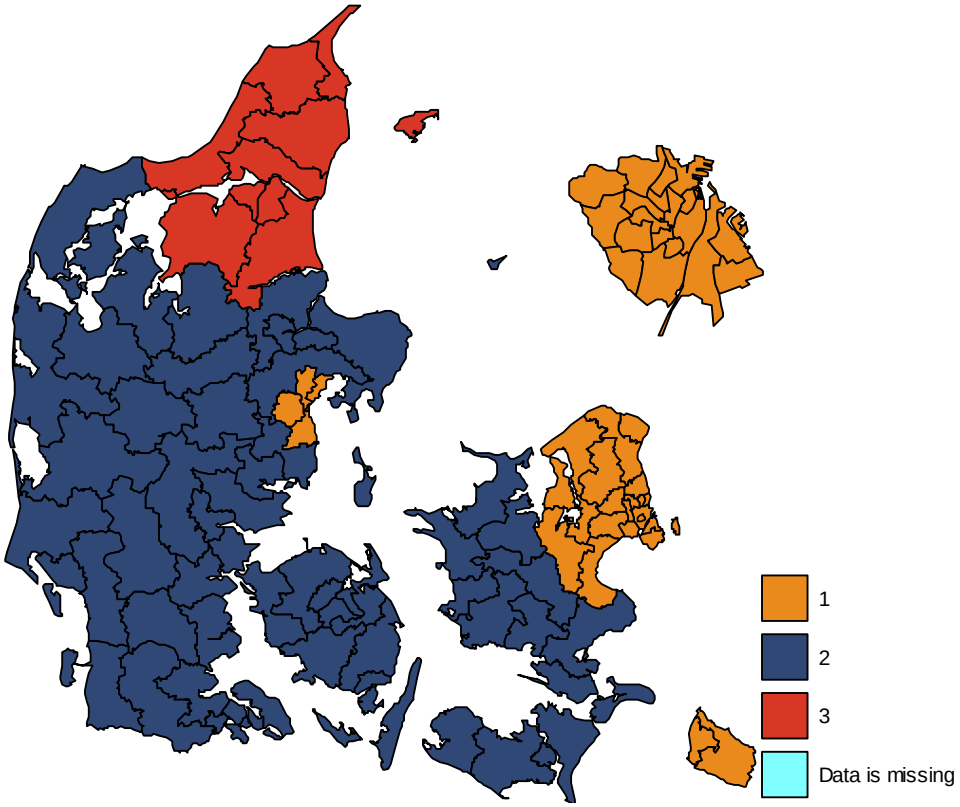


In the field with the heading Map you should also see the text "Valgkreds" which is the name in Danish of the kind of units in the map (nomination districts). If you do not see this text it is because the file Clusters.krt is not placed in the same directory as your data file. When you now press OK you should see the map similar two the one in Figure 7.7. If it is only remotely similar you should right-click on the map and select "contents of the table or graph." In the Setup dialog box you choose Hatching: Contrast (to get contrasting colors in the different clusters), and if it is not already selected you choose Number of groups equal to 3 (because there is only three cluster in the ward3 solution). Play around with the different possibilities in the left margin of the map window, and inspect the other ward solutions using the same procedure as just explained.

It is interesting that you in a map like Figure 7.7 get so homogenous regions where all districts within the same region have the same color. If you instead were looking at the geographical pattern of a single party at a certain election you would instead get a much more scattered picture because party support is strongly associated with structural characteristics like rural versus urban industry or class i.e., characteristics that are much more scattered on the map. The explanation is that change in party support is often more explained by change in "general sympathy" decided partly by a common conception of what in general is good or bad (also called culture) that is to a certain degree common for the whole country, but also at the same time a little bit different between geographical regions. As witnessed in Figure 7.7 a major cultural dividing line goes between the metropolitan area around the capital of Copenhagen (the orange area – the "big island" in the upper right corner of the map is not an island but the central Copenhagen area blown up, while the smaller island in the lower right corner of the map is the island Bornholm located far longer away in the Baltic Sea) and the rest of the country. The only districts in the "metropolitan" cluster outside the metropolitan area are the four districts on the east coast of Jutland from the city of Aarhus which is the second biggest city in Denmark after Copenhagen, also with a somewhat metropolitan culture, and the island of Bornholm (for no obvious reason).

The map in Figure 7.7 also shows that the Northern part of Jutland (in red) seems to be a homogenous culture. This is partly right, but the distinction of Northern Jutland already with three clusters is also caused by the fast that the leader of the progressive party in 1998 got a lot of votes in her own constituency, Northern Jutland, because of her own qualities in contrast to the general decline for the Progress Party in the rest of the country. In 2001 she had left active politics and the Progress Party lost heavily in Northern Jutland.

Figure 7.7 Map with three clusters generated from change in logit shares 1998-2001



The homogenous pattern is less obvious for solutions with more then three clusters. Figure 7.8 shows the solution with 12 clusters based on change in logit shares 1998-2001. Districts with the same change in party support are now more scattered, sometimes caused by somewhat random events.

To get a more stable map of political regions one should not only consider a single election period like 1998-2001 but rather several election periods. An example is the map in Figure 7.8 presenting 12 clusters based on all election periods from 1979 to 2001 also including European parliament elections and EU referendums. This map is much better in accordance with the expected pattern of regional political cultures in Denmark

Figure 7.8 Map with 12 clusters generated from change in logit shares 1998-2001

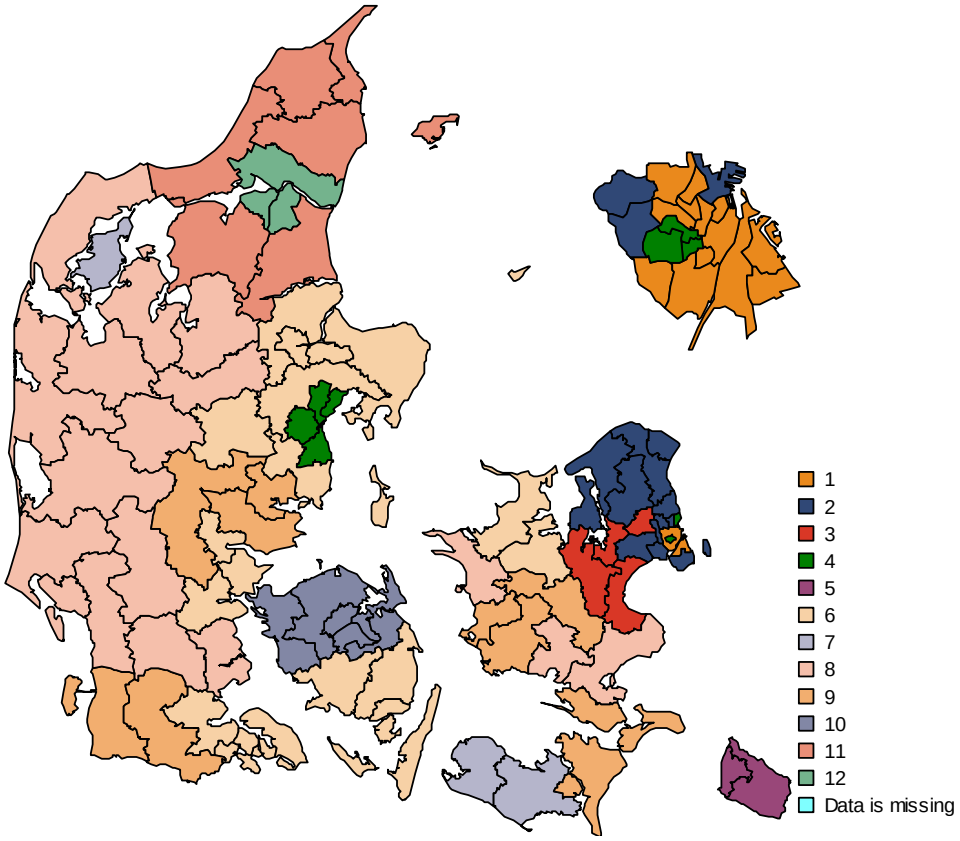


Figure 7.9 Map with 12 clusters generated from change in logit shares 1979-2001