STAD29 / STA 1007
Statistics for the Life and Social Sciences

Instructor: Ken Butler

butler@utsc.utoronto.ca

January 10, 2012

Course and instructor

- Lecture: Wednesday 14:00-16:00 in IC 328.
- Instructor: Ken Butler
- Office: IC 471.
- E-mail: `butler@utsc.utoronto.ca`
- Office hours: Mondays (until mid-afternoon), Wednesdays (from mid-morning) or by appointment. E-mail always good.
- Using Blackboard for grades only; using website for everything else.

Non-text

Prerequisites and exclusions

- **No official text** for this course. One that shares my philosophy:
  - "Using Multivariate Statistics" by Barbara G. Tabachnick and Linda S. Fidell, publ. Allyn and Bacon, ISBN 0205459382. There is a 5th edition around, but the 4th should be fine.
- Prerequisites: for undergrads STAB22 - STAB27 (or STA220-STA221). For grad students, a first course, and some training in regression and ANOVA.
- Exclusions: this course not for Statistics majors. For students in other fields who wish to learn some more advanced statistical methods. The exclusions in the Calendar reflect this.

Computing and assessment

- Computing: big part of the course, **not** optional. Demonstrate that you can use SAS to analyze data, and to critically interpret the output. *No* prior knowledge of SAS is assumed.
- Grading: (3 hour) final exam, but no midterm. There will be assignments most weeks. Graduate students also required to complete a project using one or more of the techniques learned in class, on a dataset from their field of study. Projects due at the last class of the semester.
- Assessment:

	STAD29	STA 1007
Assignments	70%	35%
Project	-	35%
Final exam	30%	30%

- Plagiarism: don't do it!

What we (might) cover

1. Review of inference; 2-sample t
2. Review of (multiple) regression
3. Logistic regression (ordinal/nominal response)
4. Survival analysis
5. Brief review of analysis of variance
6. Analysis of covariance
7. Multivariate ANOVA
8. Repeated measures by profile analysis
9. Multivariate regression
10. Discriminant analysis
11. Cluster analysis
12. Multidimensional scaling
13. Principal components
14. Exploratory factor analysis
15. Confirmatory factor analysis
16. Spatial statistics
17. Multiway frequency tables

Where we are going

1. Review of inference; 2-sample t
2. Review of (multiple) regression
3. Logistic regression (ordinal/nominal response)
4. Survival analysis
5. Brief review of analysis of variance
6. Analysis of covariance
7. Multivariate ANOVA
8. Repeated measures by profile analysis
9. Multivariate regression
10. Discriminant analysis
11. Cluster analysis
12. Multidimensional scaling
13. Principal components
14. Exploratory factor analysis
15. Confirmatory factor analysis
16. Spatial statistics
17. Multiway frequency tables

The statistical world

- Consists of:
  - objects or people of interest to us (*individuals*)
  - things measured or counted on those individuals (*variables*)
- About the individuals:
  - which ones do we care about? All of them, the *population*.
  - which ones do we know about? The ones we happened to look at, the *sample*.
- Sample is (or should be) randomly chosen from population, with no favoritism.

Sample to population: confidence interval

- Want to know about population (parameter), but don't. Only have sample (statistic). Eg. population mean, only have sample mean.
- Logic:
  - *If* we knew about population, could figure out kinds of samples that might appear (math).
  - In particular, can figure how far apart sample statistic and population parameter might be.
  - Use this to construct *confidence interval* for population parameter: says eg. "based on my sample, I think population mean between a and b".

Test of significance

- Or:
  - might have theory leading to *null hypothesis* (eg. population mean is 20) and *alternative hypothesis* (eg. population mean not 20).
  - This leads to *test of significance* (hypothesis test): "based on my sample, I think pop. mean is (is not) 20"
  - Done by choosing α (eg. 0.05), calculating *test statistic* and *P-value*. If P-value $< \alpha$, *reject null*: have evidence in favour of alternative.
- Math producing inference procedures can be difficult, but calculations (with software) and interpretations need not be.

Exploratory data analysis

- Sometimes don't have theory (yet), just want to see what data tell us.
- Use graphs, simple descriptive statistics, some of methods we learn.
- Idea: generate ideas ("hypotheses") for future study.
- Cannot make clear conclusions about populations.

The Degree of Reading Power data

- Have new method for teaching reading.
- Want to see if better than "standard" method ("research hypothesis").
- Design: randomly allocate available children to "treatment" (new method) or "control" (standard).
- Measure score for all children on standard reading test.
- Analysis: is observed difference between treatment/control score means big enough to be real not chance? Do 2-sample t-test.

Some of the data

```
t 43
t 53
t 57
t 49
t 56
t 33
c 42
c 33
c 46
c 37
c 43
```

- 1st column label ("t" for treatment, "c" for control).
- 2nd column response (score on reading test).
- Data in plain text file `drp.dat`.

Writing a SAS program

- 2 parts:
  - read data into SAS (DATA step)
  - tell SAS what to do with data (PROC step)
- DATA step, basic format 1 obs per row, each word/number a variable (separated by whitespace).
- This reads DRP data:

```
data drp;
  infile "drp.dat";
  input group $ score;
```

- Data in file `drp.dat`; 2 variables, "group" (text), score (number).

The SAS PROC step

- SAS has *many* procedures for doing things with data. We look at 3:
  - PROC PRINT simply lists data values
  - PROC MEANS computes means and SDs for variables given
  - PROC TTEST does 1- and 2-sample t tests
- Add PROC steps plus options to file containing data step. Here PROC MEANS and PROC TTEST have same options: "class" is variable splitting data into groups, "var" is (response) variable:

```
proc means;
  class group;
  var score;

proc ttest;
  class group;
  var score;
```


The whole thing

```
options linesize=80;

data drp;
  infile "drp.dat";
  input group $ score;

proc print;

proc means;
  class group;
  var score;

proc ttest;
  class group;
  var score;

run;
```

- I like to indent lines belonging to each step and leave blank line between steps.
- (Optionally) Save in file like "drp.sas".

To run SAS

Submit this program (Run, Submit).
To see if it worked, look at Log window. Any lines beginning `ERROR:` are things needing to be fixed. Go to Program Editor, select Run, Recall Last Submit. Fix errors, submit again.
Things like this indicate success:

```
NOTE: 44 records were read from the infile "drp.dat".
      The minimum record length was 6.
      The maximum record length was 6.
NOTE: The data set WORK.DRP has 44 observations and 2 variables.

NOTE: The PROCEDURE PRINT printed page 1.

NOTE: The PROCEDURE MEANS printed page 2.

NOTE: The PROCEDURE TTEST printed page 3.
```

The output, part 1

- In Output window.
- Page 1 just a listing of data.
- Page 2:

The MEANS Procedure
Analysis Variable : score

group	N Obs	N	Mean	Std Dev	Minimum
c	23	23	41.5217391	17.1487332	10.0000000
t	21	21	51.4761905	11.0073568	24.0000000

Analysis Variable : score

group	N Obs	Maximum
c	23	85.0000000
t	21	71.0000000

Treatment group has higher mean score, but control group scores more variable.

The t-test

. . . edited:

group	Method	Mean	95% CL Mean		Std Dev
c		41.5217	34.1061	48.9374	17.1487
t		51.4762	46.4657	56.4867	11.0074
Diff (1-2)	Pooled	-9.9545	-18.8176	-1.0913	14.5512
Diff (1-2)	Satterthwaite	-9.9545	-18.6759	-1.2330	

Method	Variances	DF	t Value	Pr > |t|
Pooled	Equal	42	-2.27	0.0286
Satterthwaite	Unequal	37.855	-2.31	0.0264

Equality of Variances

Method	Num DF	Den DF	F Value	Pr > F
Folded F	22	20	2.43	0.0507

Conclusions

- SAS does 2 t procedures:
  - Pooled: assumes 2 population variances/SDs are same
  - Satterthwaite: does not, but only approximation.
- Sample SDs quite different, suggests use of Satterthwaite.
- For test: look at P-value 0.0264. Less than 0.05, so have evidence of difference in mean test scores between reading methods.
- Satterthwaite CI for difference in means -18.7 to -1.2 (control minus treatment): treatment better.
- P-values for Satterthwaite vs. pooled very close (0.0286 and 0.0264), so conclusion not affected by choice of test.
- Last test for equality of variances/SDs between 2 groups. P-value 0.0507 very close to significance, supporting use of Satterthwaite.

Where we are going

Regression

- Use regression when one variable is an outcome (*response*, y).
- See if/how response depends on other variable(s), *explanatory*, $x_1, x_2, \ldots$.
- Can have *one* or *more than one* explanatory variable, but always one response.
- Assumes a *straight-line* relationship between response and explanatory.
- Ask:
  - *is there* a relationship between y and x's, and if so, which ones?
  - what does the relationship look like?

A regression with one x

13 children, measure average total sleep time (ATST, mins) and age (years) for each. See if ATST depends on age. Data in `sleep.dat`, ATST then age. Read in data:

```
data sleep;
  infile "sleep.dat";
  input atst age;
```

and make scatter plot of ATST (response) vs. age (explanatory) using this code:

```
proc plot;
  plot atst * age;
```

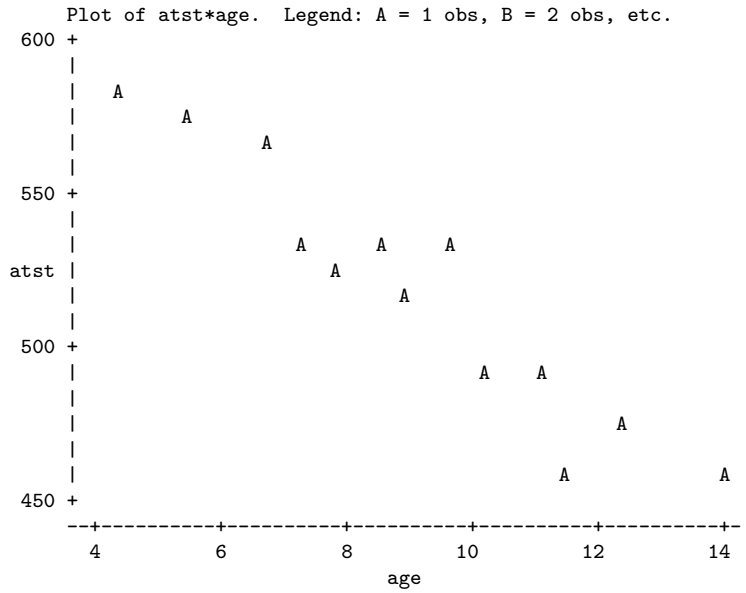

Code continued

Either run this first and see what the plot looks like, or be an optimist and add regression to end of this:

```
proc reg;
  model atst=age;
```

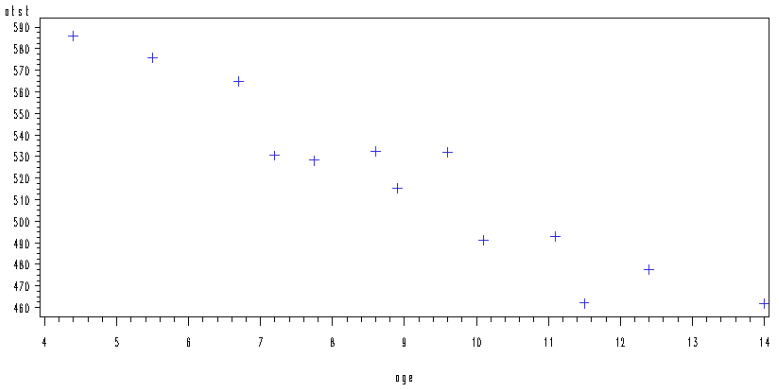
Assemble these commands in file `sleep.sas` and then run `sas sleep.sas`. Check `sleep.log` for any errors.

The scatterplot



A better scatterplot

Replace `plot` by `gplot` and re-submit:



The regression

Scatterplot shows no obvious curve, and a pretty clear downward trend. So we can run the regression:

```
                    The REG Procedure
                      Model: MODEL1
                Dependent Variable: atst

          Number of Observations Read          13
          Number of Observations Used          13

                    Analysis of Variance
                                 Sum of          Mean
Source                    DF     Squares        Square   F Value   Pr > F

Model                      1       18221         18221    105.33   <0.0001
Error                     11  1902.83505     172.98500
Corrected Total           12       20123
```


more. . . and conclusions

```
      Root MSE            13.15238    R-Square     0.9054
      Dependent Mean     519.30385    Adj R-Sq     0.8968
      Coeff Var            2.53269

                  Parameter Estimates

                  Parameter       Standard
Variable    DF     Estimate          Error   t Value   Pr > |t|

Intercept    1    646.48334       12.91773     50.05     <.0001
age          1    -14.04105        1.36812    -10.26     <.0001
```

- The relationship appears to be a straight line, with a downward trend.
- F-tests for model as a whole and t-test for slope (same) both confirm this.
- Slope is -14, so a 1-year increase in age goes with a 14-minute decrease in ATST on average.

CI for mean response and prediction intervals

Once useful regression exists, use it for prediction:

- To get a single number for prediction at a given x, substitute into regression equation, eg. age 10: predicted ATST is $646.48 - 14.04(10) = 506$ minutes.
- To express uncertainty of this prediction:
  - *CI for mean response* expresses uncertainty about mean ATST for all children aged 10, based on data.
  - *Prediction interval* expresses uncertainty about predicted ATST for a new child aged 10 whose ATST not known. More uncertain.
- Also do above for a child aged 3.

Intervals in SAS

- To get SAS to compute these:
  - add to end of data file line for each prediction, missing for response:

    ```
    . 10
    . 3
    ```

    (the dot is SAS's version of "missing")
  - modify SAS code to read

    ```
    proc reg;
      model atst=age / cli clm;
    ```

    The / is to distinguish variables from options.

The output

Includes all the stuff from before plus:

Dependent Variable: atst

Output Statistics

Obs	Dependent Variable	Predicted Value	Std Error Mean Predict	95% CL Mean	
1	586.0000	584.7027	7.3425	568.5420	600.8635
2	461.7500	449.9087	7.6829	432.9988	466.8185
	... snip				
13	530.5000	545.3878	4.4459	535.6024	555.1731
14	.	506.0729	3.8689	497.5574	514.5883
15	.	604.3602	9.0549	584.4305	624.2899

Obs. 14 is new "obs" with age 10, obs. 15 with age 3.

. . . continued

Output Statistics

Obs	95% CL Predict		Residual
1	551.5490	617.8564	1.2973
2	416.3834	483.4339	11.8413
	... snip		
13	514.8305	575.9451	-14.8878
14	475.8982	536.2475	.
15	569.2149	639.5055	.

- Age 10 closer to centre of data, so intervals are both narrower than those for age 3.
- Age 3 assumes that straight line continues to hold (don't have any data to support that)
- Prediction intervals bigger than CI for mean (additional uncertainty).

Diagnostics

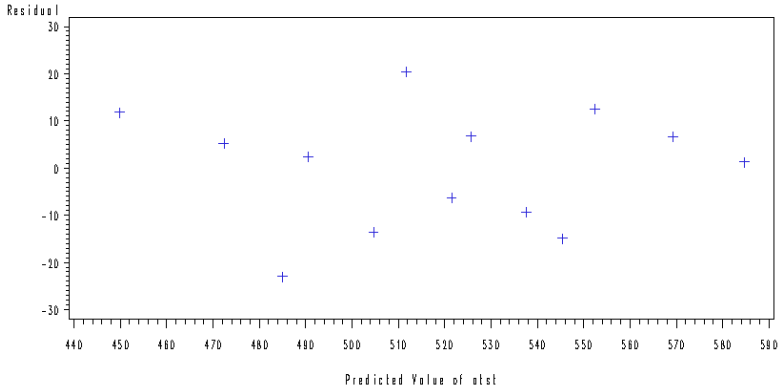
How do we tell whether a straight-line regression is appropriate?

- Before: check scatterplot for straight trend.
- After: plot *residuals* (observed minus predicted response) against predicted values. Aim: a plot with no pattern.

SAS approach: compute residuals and *save* them in a new data set, then plot using stuff in new data set. Code:

```
proc reg;
  model atst=age;
  output out=z p=predicted r=residual;

proc gplot
  plot residual * predicted;

run;
```

Output



Not much pattern here (is residual predictable from predicted? No). Good, indicating regression appropriate.

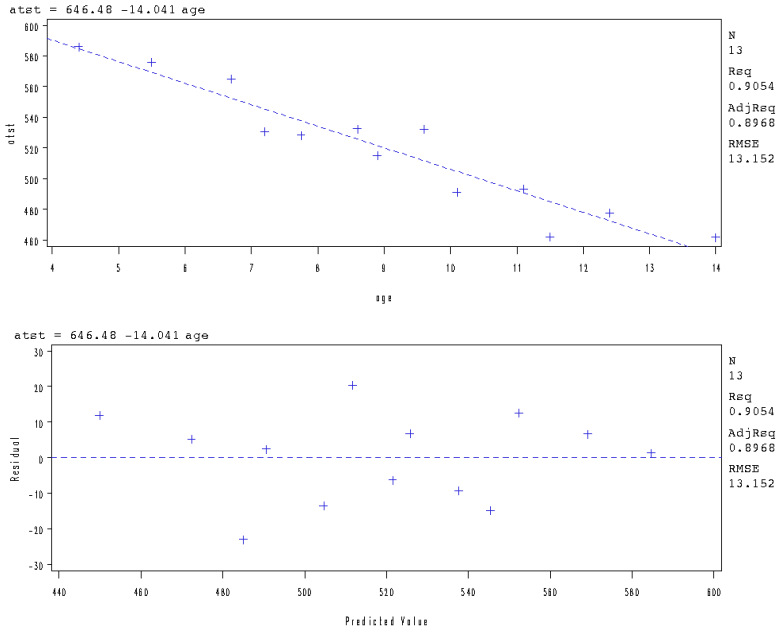
Another way of getting plots

- General principle: get output data set from `proc`, look at or plot.
- `proc reg` also allows embedding of `plot`, like this:

```
proc reg;
  model atst=age;
  plot atst*age;
  plot r.*p.;
```

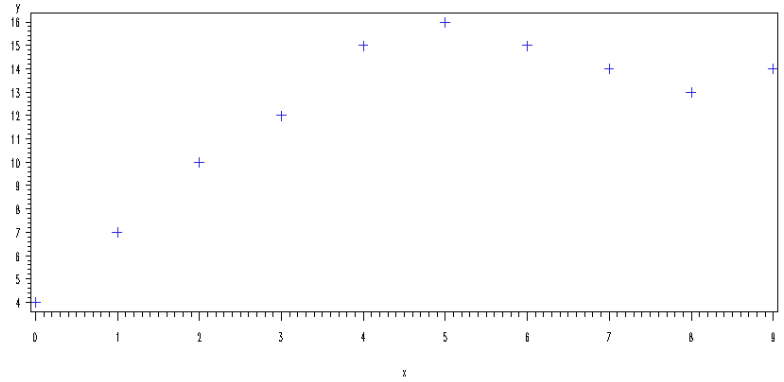
- Last line obtains plot of residuals vs predicted values.
- Scatterplot has regression line added; residual plot has 0 line added:

The plots



An inappropriate regression

Scatterplot of different data:



Trend goes up, then levels off, but a line would keep going up.

Regression line

Try fitting a regression line anyway, saving and plotting residuals using this code:

```
proc reg;
  model y=x;
  output out=z p=pred r=resid;

proc plot;
  plot resid * pred;

run;
```


Alternatively...

Get plot from `proc reg` and don't save output data set:

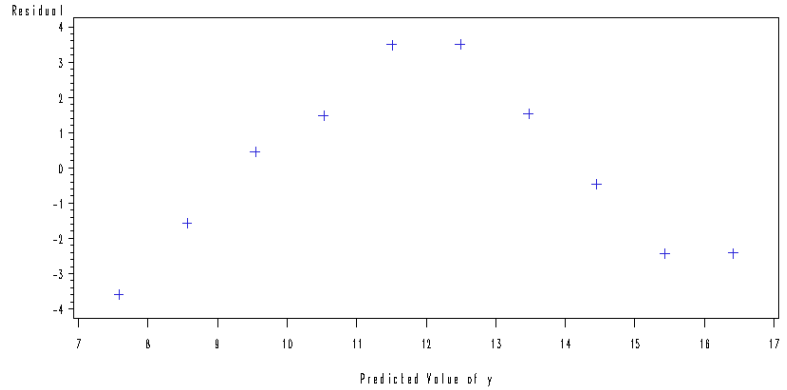
```
proc reg;
  model y=x;
  plot r.*p.;

run;
```


Output

```
                    The REG Procedure
                      Model: MODEL1
                  Dependent Variable: y

... snip
                      Parameter       Standard
Variable     DF        Estimate          Error    t Value    Pr > |t|

Intercept     1         7.58182        1.56160       4.86      0.0013
x             1         0.98182        0.29251       3.36      0.0100
```

Regression appears good: slope significantly different from zero. But ...

Residual plot



Residual plot has *curve*: middle residuals positive, high and low ones negative. Bad.

Fixing it up

Fitting a curve would be better. Try this:

```
data curve;
  infile "curvy.dat";
  input x y;
  xsq=x*x;

proc reg;
  model y=x xsq;
```

Define a new variable that is x-squared, and add this to the regression model (now *multiple* regression).

The output

The REG Procedure
Model: MODEL1
Dependent Variable: y

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value
Model	2	129.23182	64.61591	66.83
Error	7	6.76818	0.96688	
Corrected Total	9	136.00000		

Analysis of Variance

Source	Pr > F
Model	<.0001

Model as a whole fits well.

Continued

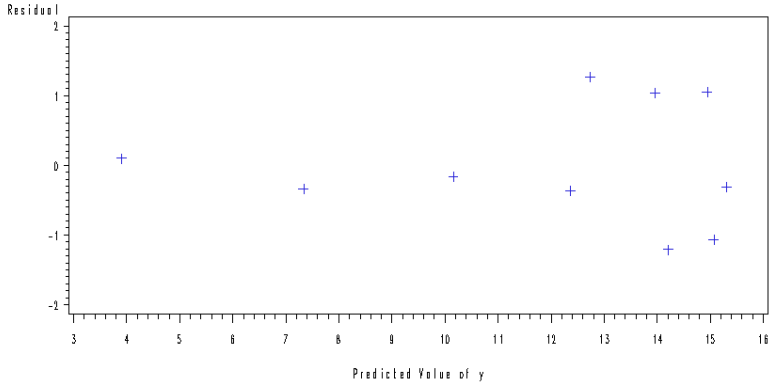
Root MSE	0.98330	R-Square	0.9502
Dependent Mean	12.00000	Adj R-Sq	0.9360
Coeff Var	8.19418		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	3.90000	0.77312	5.04	0.0015
x	1	3.74318	0.40006	9.36	<.0001
xsq	1	-0.30682	0.04279	-7.17	0.0002

R-squared is higher (better fit), and slope for new variable `xsq` is significantly nonzero — helps to predict y over and above x. Curve better than straight line. (When you have `xsq`, keep `x` in regardless of its significance because `x` "contained in" `xsq`.)

The residual plot now



No problems any more.

Multiple regression

- What if more than one x? Extra issues:
  - Now one intercept and a slope for each x: how to interpret?
  - Which x-variables actually help to predict y? Different interpretations of "global" F-test and individual t-tests.
- SAS code easy: on `model` line, add extra xs after =.
- Interpretation not so easy (and other problems that can occur).

Multiple regression example

Study of women and visits to health professionals, and how the number of visits might be related to other variables:

- timedrs: number of visits to health professionals (over course of study)
- phyheal: number of physical health problems
- menheal: number of mental health problems
- stress: result of questionnaire about number and type of life changes

`timedrs` response, others explanatory.

The SAS code

Ideas:

- read in data (first line is variable names so skip over)
- do regression predicting response from all explanatory
- save predicted values and residuals; plot later
- fit another regression model for comparison

```
data regr;
  infile "regressx.dat" firstobs=2;
  input subject timedrs phyheal menheal stress;
proc reg;
  model timedrs = phyheal menheal stress;
  output out=z1 p=pred1 r=res1;
  model timedrs = menheal;
proc gplot data=z1;
  plot res1 * pred1;
proc univariate plot;
  var res1;
```


Output part 1

```
                    The REG Procedure
                      Model: MODEL1
                Dependent Variable: timedrs
            Number of Observations Used         465

                    Analysis of Variance

                                Sum of         Mean
Source                   DF     Squares       Square  F Value  Pr > F

Model                     3       12168   4056.10512    43.03  <.0001
Error                   461       43451     94.25409
Corrected Total         464       55619

        Root MSE              9.70845    R-Square     0.2188
        Dependent Mean        7.90108    Adj R-Sq     0.2137
        Coeff Var           122.87510
```

Model as a whole strongly significant even though R-sq not very big (lots of data). At least one of the x's predicts `timedrs`.

The slopes

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-3.70485	1.12420	-3.30	0.0011
phyheal	1	1.78695	0.22107	8.08	<.0001
menheal	1	-0.00967	0.12903	-0.07	0.9403
stress	1	0.01361	0.00361	3.77	0.0002

The physical health and stress variables definitely help to predict the number of visits, but *with those in the model* we don't need `menheal`. However, look at prediction of `timedrs` from `menheal` by itself:

Just menheal (edited)

The REG Procedure
Model: MODEL2
Dependent Variable: timedrs
Number of Observations Used 465

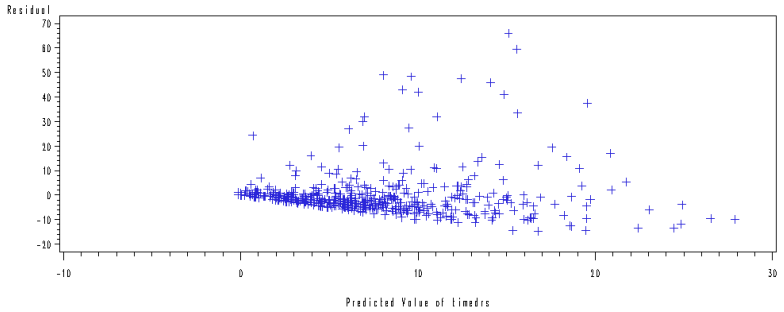
Root MSE 10.59632 R-Square 0.0653

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	3.81588	0.87022	4.38	<.0001
menheal	1	0.66723	0.11730	5.69	<.0001

`menheal` by itself *does* significantly help to predict `timedrs`. But the R-sq is much less (6.5% vs. 22%) so the other two variables do a better job of prediction.
Go back to regression of `timedrs` on all x's: predicts significantly, but is it appropriate? Look at plot of residuals vs. predicted values.

Residual plot



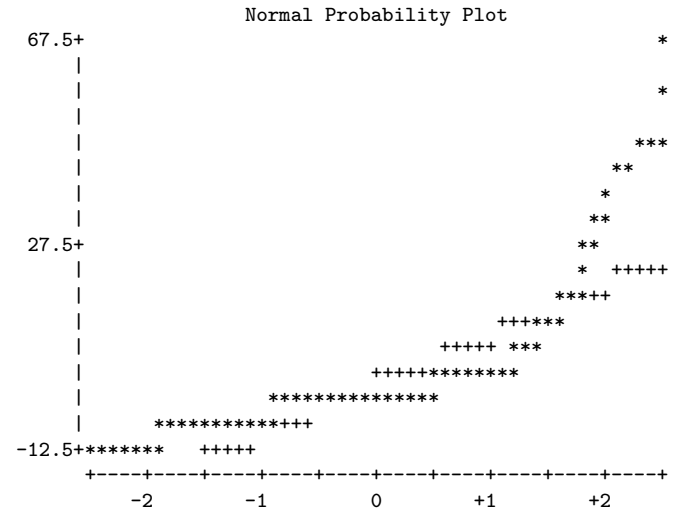
Residuals are not normal

- No pattern
- but some very positive residuals (compared to how negative).
- Distribution of residuals is *skewed*, not normal as it should be.
- See more clearly from a (sideways) histogram of residuals (output from PROC UNIVARIATE PLOT).

Univariate plots of residuals

```
                  Histogram                          #   Boxplot
 67.5+*                                              1      *
     .                                                
     .*                                              1      *
     .                                                
     .*                                              4      *
     .*                                              3      *
     .*                                              1      *
     .*                                              4      *
 27.5+*                                              2      *
     .*                                              2      *
     .**                                             7      0
     .***                                           13      0
     .****                                          16      0
     .******************                            92    +--+--+
     .*********************************************  229   *-----*
     .****************                              77      |
-12.5+***                                           13      0
      ----+----+----+----+----+----+----+----+----+-
      * may represent up to 5 counts
```


Normal probability plot of residuals



Fixing the problems

- Sometimes residuals are *very* positive: observed a *lot* larger than predicted.
- Try *transforming* response: use log or square root of response. (Note that response is *count*, often skewed to right.)
- Try regression again. Define transformed `timedrs` in data step, and use transformed variable as response. Check residual plot to see that it is OK now:

```
data reg2;
  infile "regressx.dat" firstobs=2;
  input subject timedrs phyheal menheal stress;
  lgtime=log(timedrs+1);
proc reg;
  model lgtime=phyheal menheal stress;
  output out=z2 p=pred2 r=res2;
proc gplot;
  plot res2*pred2;
```

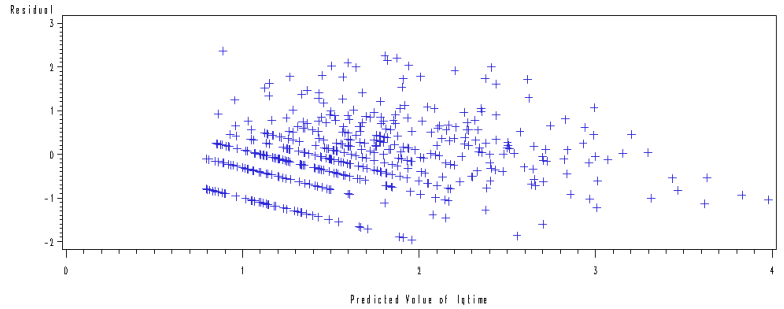

Output

```
                    The REG Procedure
                    Model: MODEL1
                Dependent Variable: lgtime

        Number of Observations Used                       465

... snip

        Root MSE              0.76247    R-Square     0.3682

                    Parameter Estimates

                     Parameter       Standard
Variable     DF       Estimate          Error    t Value    Pr > |t|

Intercept     1        0.39039        0.08829       4.42      <.0001
phyheal       1        0.20194        0.01736      11.63      <.0001
menheal       1        0.00714        0.01013       0.71      0.4812
stress        1        0.00132     0.00028369       4.64      <.0001
```

Comments

- Model as a whole strongly significant again (not shown)
- R-sq higher than before (37% vs. 22%) suggesting things more linear now
- Same conclusion re `menheal`: can take out of regression.
- Should look at residual plot (next page).

The residual plot



Much better. Residuals range from 2 to -2, and look symmetric in shape. Should be trustworthy now.

Box-Cox transformations

- Taking log of `timedrs` and having it work: lucky guess. How to find good transformation?
- Idea: Box-Cox: *estimate* the kind of transformation that would work: take power of response (1 = no change, 0.5 = square root, *0* = *log*).
- `proc transreg`.

PROC TRANSREG

- Some of `timedrs` values are 0, but Box-Cox expects all +. Define new variable `tp` in data step, then call `proc transreg` with that as response.

```
data regr;
    infile "regressx.dat" firstobs=2;
    input subject timedrs phyheal menheal stress;
    tp=timedrs+1;

proc transreg;
    model boxcox (tp) = identity(phyheal menheal stress);
```

- `tp` only necessary here because of zeros in `timedrs`; normally omit and use original response in `boxcox`.

The output (edited)

```
                The SAS System                        17
                      09:00 Wednesday, January 12, 2011

          The TRANSREG Procedure

Box-Cox Transformation Information for tp

    Lambda      R-Square    Log Like

     -3.00          0.10    -2053.25
     -2.75          0.11    -1892.35
       ...
     -0.50          0.33     -725.30
     -0.25          0.36     -677.02
      0.00 +        0.37     -667.59 <
      0.25          0.36     -703.08
      0.50          0.32     -783.77
      1.00          0.22    -1056.94
       ...
      3.00          0.05    -2841.16

  < - Best Lambda
  * - 95% Confidence Interval
  + - Convenient Lambda
```

- Best transformation uses $\lambda = 0$, corresponding to log.
- Square root transformation $\lambda = 0.5$ not so good.
- No transformation ($\lambda = 1$) worse still.

Testing more than one x at once

The t-tests test only whether one variable could be taken out of the regression you're looking at. To test significance of more than one variable at once, or to see whether certain values for the slopes consistent with data, use SAS `test` in PROC REG, eg.:

```
proc reg;
  model lgtime=phyheal menheal stress;
  test menheal=0, phyheal=0;
  test menheal=0.02, phyheal=0.2;
```

- 1st: take out both `menheal` and `phyheal`?
- 2nd: these values for slopes consistent with data?

Results of tests

```
      Test 1 Results for Dependent Variable lgtime

                              Mean
Source              DF        Square    F Value    Pr > F

Numerator            2      51.73210      88.98    <.0001
Denominator        461       0.58136

      Test 2 Results for Dependent Variable lgtime

                              Mean
Source              DF        Square    F Value    Pr > F

Numerator            2       0.54126       0.93    0.3949
Denominator        461       0.58136
```

- First test says "taking both variables out makes the fit worse, so don't do it".
- Second test says "yes, those values are consistent with the data" (we do not reject them).

The punting data

Data set `punting.dat` contains 4 variables for 13 right-footed football kickers (punters): left leg and right leg strength (lbs), distance punted (ft), another variable called "fred". Predict punting distance from other variables:

```
data punt;
  infile "punting.dat";
  input left right punt fred;

proc reg;
  model punt=left right fred;

proc corr;
  var punt left right fred;
```

PROC CORR finds correlations between variables.

Regression output (edited)

The REG Procedure
Dependent Variable: punt

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	6798.13109	2266.04370	10.52	0.0027

Root MSE	14.67520	R-Square	0.7781

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-4.68554	29.11722	-0.16	0.8757
left	1	0.26787	2.11110	0.13	0.9018
right	1	1.05241	2.14771	0.49	0.6358
fred	1	-0.26724	4.22661	-0.06	0.9510

Comments

- Regression strongly significant, R-sq high.
- None of the x's significant! Why?
- t-tests only say that you could take any one of the x's out without damaging the fit; doesn't matter which one.
- Explanation: look at *correlations*. (Reason for PROC CORR.)

The correlations

Pearson Correlation Coefficients, N = 13
Prob > |r| under H0: Rho=0

	punt	left	right	fred
punt	1.00000	0.81174	0.88055	0.86795
		0.0008	<.0001	0.0001
left	0.81174	1.00000	0.89572	0.97226
	0.0008		<.0001	<.0001
right	0.88055	0.89572	1.00000	0.97288
	<.0001	<.0001		<.0001
fred	0.86795	0.97226	0.97288	1.00000
	0.0001	<.0001	<.0001	

All correlations are high: x's with `punt` (good) and with each other (bad, at least confusing). How to detect? Use Variance Inflation Factor (next):

VIF code and output

```
proc reg;
  model punt=left right fred / vif;
```

Variable	DF	Variance Inflation
Intercept	1	0
left	1	130.53235
right	1	133.45186
fred	1	482.24616

Any VIF > 10 means trouble: here *all* the x's are highly correlated with each other, `fred` being worst.
Suggests: just pick one x. Kickers are right-footed, so try `right`:

```
proc reg;
  model punt=right / vif;
```

Output (edited)

```
      Root MSE            13.35704    R-Square     0.7754

                    Parameter Estimates

                   Parameter       Standard
Variable     DF     Estimate          Error    t Value    Pr > |t|

Intercept     1      -3.69304       25.26487      -0.15      0.8864
right         1       1.04267        0.16922       6.16      <.0001

                    Parameter Estimates

                                    Variance
                Variable     DF     Inflation

                Intercept     1             0
                right         1       1.00000
```

R-sq almost as high as before, no problems with VIF. Punting distance definitely predicted by right-leg strength

Where we are going

Logistic regression

- When response variable is measured/counted, regression can work well.
- But what if response is yes/no, lived/died, success/failure?
- Model *probability* of success.
- Probability must be between 0 and 1; need method that ensures this.
- *Logistic regression* does this; PROC LOGISTIC in SAS.
- Begin with simplest case.

The rats, part 1

Rats given dose of some poison; either live or die:

```
0 lived
1 died
2 lived
3 lived
4 died
5 died
```

Basic logistic regression analysis:

```
options linesize=80;

data rat;
  infile "rat.dat";
  input dose survival $;

proc logistic;
  class survival;
  model survival = dose;
  output out=rat2 pred=pred;

proc print data=rat2;
```

Output

The LOGISTIC Procedure

Model Information

Data Set	WORK.RAT
Response Variable	survival
Number of Response Levels	2
Model	binary logit
Optimization Technique	Fisher's scoring

Number of Observations Read	6
Number of Observations Used	6

Response Profile

Ordered Value	survival	Total Frequency
1	died	3
2	lived	3

Probability modeled is survival='died'.

Output part 2 (edited)

Model Convergence Status

Convergence criterion (GCONV=1E-8) satisfied.

... snip

Testing Global Null Hypothesis: BETA=0

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	1.5449	1	0.2139
Score	1.4286	1	0.2320
Wald	1.2037	1	0.2726

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-1.6841	1.7978	0.8774	0.3489
dose	1	0.6736	0.6140	1.2037	0.2726

Interpreting the output

- Like (multiple) regression, get:
  - overall test of model ("global null hypothesis")
  - tests of significance of individual x's ("analysis of maximum likelihood estimates").
- Here none of them significant (only 6 observations).
- These tests all agree for regression, but don't for logistic regression. Look for consistent picture (Wald often different from others).
- Look at event "modeled", here "died".
- "Slope" for dose is positive, meaning that as dose increases, probability of event modelled (death) increases.
- Output data set contains predicted probabilities (next slide):

Predicted probabilities

Obs	dose	survival	_LEVEL_	pred
1	0	lived	died	0.15656
2	1	died	died	0.26690
3	2	lived	died	0.41658
4	3	lived	died	0.58342
5	4	died	died	0.73310
6	5	died	died	0.84344

"Pred" is predicted probability of event named by _LEVEL_ (death). Goes up as dose increases.

The rats, part 2

- More realistic: more rats at each dose (say 10).
- Listing each rat on one line makes a big data file.
- Use format below: dose, number of deaths, number of trials (rats):

```
0 0 10
1 3 10
2 4 10
3 6 10
4 8 10
5 9 10
```

- Alter model line for PROC LOGISTIC to say:

```
model deaths/trials = dose;
```


SAS code for this logistic regression

```
options linesize=80;

data rat;
  infile "rat2.dat";
  input dose deaths trials;

proc logistic;
  model deaths/trials = dose;
  output out=rat2 pred=pred lower=lcl upper=ucl;

proc print data=rat2;
```

This time, have output data set also contain lower and upper limits of a 95% CI for each death probability.

Output part 1 (edited)

```
Number of Observations Read           6
Number of Observations Used           6
Sum of Frequencies Read              60
Sum of Frequencies Used              60


            Response Profile

  Ordered      Binary          Total
    Value      Outcome     Frequency

        1      Event              30
        2      Nonevent           30


         Model Convergence Status

Convergence criterion (GCONV=1E-8) satisfied.
```

The 6 lines of data correspond to 60 actual rats.

Output part 2 (edited)

```
        Testing Global Null Hypothesis: BETA=0

Test                 Chi-Square       DF     Pr > ChiSq

Likelihood Ratio        25.0562        1         <.0001
Score                   21.9657        1         <.0001
Wald                    16.1449        1         <.0001

       Analysis of Maximum Likelihood Estimates

                                 Standard          Wald
Parameter    DF    Estimate       Error    Chi-Square    Pr > ChiSq

Intercept     1     -2.3619      0.6719       12.3585        0.0004
dose          1      0.9448      0.2351       16.1449        <.0001
```

- All 4 tests agree: significant effect of dose.
- Effect of larger dose is to increase death probability ("slope" positive).

Predicted probabilities

Just run PROC PRINT on output data set:

Obs	dose	deaths	trials	pred	lcl	ucl
1	0	0	10	0.08612	0.02463	0.26017
2	1	3	10	0.19511	0.08646	0.38304
3	2	4	10	0.38405	0.24041	0.55124
4	3	6	10	0.61595	0.44876	0.75959
5	4	8	10	0.80489	0.61696	0.91354
6	5	9	10	0.91388	0.73983	0.97537

- Predicted death probs increase with dose.
- Last 2 columns are 95% CI for prob of death at each dose (eg. dose 2, from 0.24 to 0.55).
- Intervals still quite wide even with $n = 60$ rats.
- Each rat doesn't contribute much information (just lived/died) so need n in hundreds to get precise intervals.

Multiple logistic regression

- With more than one x, works much like multiple regression.
- Example: study of patients with blood poisoning severe enough to warrant surgery. Relate survival to other potential risk factors.
- Variables, 1=present, 0=absent:
  - survival (death from sepsis=1), response
  - shock
  - malnutrition
  - alcoholism
  - age (as numerical variable)
  - bowel infarction
- See what relates to death.

Some SAS code

```
data x;
  infile "sepsis.dat";
  input death shock malnut alcohol age bowelinf;

proc logistic;
  model death=shock malnut alcohol age bowelinf;
  test malnut=0, bowelinf=0;

proc logistic;
  model death=shock alcohol age bowelinf;
  output out=z pred=p;

proc print data=z;
```

Use of PROC LOGISTIC resembles use of PROC REG, including "test".

Output part 1

```
Number of Observations Used         106

              Response Profile

Ordered                          Total
  Value        death     Frequency
      1            0            85
      2            1            21

Probability modeled is death=0.

Testing Global Null Hypothesis: BETA=0

Test                 Chi-Square       DF     Pr > ChiSq

Likelihood Ratio        52.4060        5         <.0001
Score                   43.8921        5         <.0001
Wald                    16.2433        5         0.0062
```

Model as a whole is significant: at least one of the x's helps predict death (actually modelling P(survival)).

Finding significant x's

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	9.7539	2.5417	14.7267	0.0001
shock	1	-3.6739	1.1648	9.9479	0.0016
malnut	1	-1.2166	0.7282	2.7909	0.0948
alcohol	1	-3.3549	0.9821	11.6691	0.0006
age	1	-0.0922	0.0303	9.2353	0.0024
bowelinf	1	-2.7976	1.1640	5.7767	0.0162

- Only marginal one is `malnut`.
- Test that both `malnut` and `bowelinf` can be removed (suspect not):

Label	Wald Chi-Square	DF	Pr > ChiSq
Test 1	6.8302	2	0.0329

- Indeed, not.

Predictions from model without "malnut"

- So fit model without `malnut` and obtain predictions.
- A few chosen at random:

Obs	death	shock	malnut	alcohol	age	bowelinf	_LEVEL_	p
4	0	0	0	0	26	0	0	0.99858
1	0	0	0	0	56	0	0	0.97945
2	0	0	0	0	80	0	0	0.84658
11	1	0	0	1	66	1	0	0.06871
32	1	0	0	1	49	0	0	0.78700

- Survival chances pretty good if no risk factors, though decreasing with age.
- Having more than one risk factor reduces survival chances dramatically.
- Usually model does a good job of predicting survival, but occasionally someone dies who was predicted to survive.

Changing the response category

- In first rats example, got prob of death but maybe wanted prob of living.
- Change `model` line to this:

```
model survival(event='lived') = dose;
```

- Output now includes:

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	1.6841	1.7978	0.8774	0.3489
dose	1	-0.6736	0.6140	1.2037	0.2726

Obs	dose	survival	_LEVEL_	pred
1	0	lived	lived	0.84344
2	1	died	lived	0.73310
3	2	lived	lived	0.58342
4	3	lived	lived	0.41658
5	4	died	lived	0.26690
6	5	died	lived	0.15656

Testing fit: seroconversion example

- Seroconversion: body develops specific antibodies to microorganisms in blood (as when person gets certain disease).
- Seropositive: still have antibodies in blood after recovery from the disease.
- Malaria survey: ages plus seropositiveness recorded. Data, with variables: age group number, middle of age group, #individuals, #seropositive:

1	1.5	123	8
2	4.0	132	6
3	7.5	182	18
4	12.5	140	14
5	17.5	138	20
6	25.0	161	39
7	35.0	133	19
8	47.0	92	25
9	60.0	74	44

Does seropositiveness depend on age?

Calculate observed proportion seropositive for each age group in DATA step:

```
data sero;
  infile "sero.dat";
  input group age n r;
  obspos=r/n;
proc print;
```

Obs	group	age	n	r	obspos
1	1	1.5	123	8	0.06504
2	2	4.0	132	6	0.04545
3	3	7.5	182	18	0.09890
4	4	12.5	140	14	0.10000
5	5	17.5	138	20	0.14493
6	6	25.0	161	39	0.24224
7	7	35.0	133	19	0.14286
8	8	47.0	92	25	0.27174
9	9	60.0	74	44	0.59459

Does a logistic regression fit?

- Prob of being seropositive generally increases with age, but age group 6 has too many seropositives and age group 7 too few.
- Fit logistic model anyway, and test for fit.
- Hosmer-Lemeshow test:
  - null: logistic regression is appropriate
  - alternative: it is not.
- Code (note "events/trials" syntax and "lackfit"):

```
proc logistic;
  model r/n = age / lackfit;
```


Hosmer-Lemeshow test output

Partition for the Hosmer and Lemeshow Test

		Event		Nonevent	
Group	Total	Observed	Expected	Observed	Expected
1	123	8	8.14	115	114.86
2	132	6	9.69	126	122.31
3	182	18	15.43	164	166.57
4	140	14	14.53	126	125.47
5	138	20	17.46	118	120.54
6	161	39	27.11	122	133.89
7	133	19	31.97	114	101.03
8	92	25	32.30	67	59.70
9	74	44	36.38	30	37.62

Hosmer and Lemeshow Goodness-of-Fit Test

Chi-Square	DF	Pr > ChiSq
21.3185	7	0.0033

Interpretation

- Actually a chi-squared test based on division of x (age) into groups (here, 9 age groups).
- P-value 0.0033 small, so logistic regression not appropriate.
- Maybe age groups 6 and 7 are wrong way around. Assume this (in practice wouldn't, of course)
- Fit same model again and re-do Hosmer-Lemeshow.

Output from this analysis

The LOGISTIC Procedure

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-2.8107	0.1565	322.5387	<.0001
age	1	0.0476	0.00457	108.4657	<.0001

Hosmer and Lemeshow Goodness-of-Fit Test

Chi-Square	DF	Pr > ChiSq
8.4427	7	0.2952

- No problems with logistic model now.
- Probability of being seropositive definitely increases with age.

Predicted probabilities

Obs	age	n	r	pobs	pred	lcl	ucl
1	1.5	123	8	0.06504	0.06069	0.04588	0.07989
2	4.0	132	6	0.04545	0.06783	0.05227	0.08759
3	7.5	182	18	0.09890	0.07914	0.06258	0.09961
4	12.5	140	14	0.10000	0.09830	0.08042	0.11963
5	17.5	138	20	0.14493	0.12147	0.10230	0.14366
6	25.0	133	19	0.14286	0.16494	0.14313	0.18934
7	35.0	161	39	0.24224	0.24115	0.21102	0.27409
8	47.0	92	25	0.27174	0.35991	0.30883	0.41437
9	60.0	74	44	0.59459	0.51061	0.43049	0.59018

Plenty of data, so CIs are mostly short. Note clear upward trend in probabilities.

More than 2 response categories

- With 2 response categories, model the probability of one, and prob of other is one minus that. So doesn't matter which category you model.
- With more than 2 categories, have to think more carefully about the categories: are they
  - *ordered*: you can put them in a natural order (like low, medium, high)
  - *nominal*: ordering the categories doesn't make sense (like red, green, blue).
- SAS handles both kinds of response; learn how.

Ordinal response: the miners

- Model probability of being in given category *or lower*.
- Example: coal-miners often suffer disease pneumoconiosis. Likelihood of disease believed to be greater among miners who have worked longer.
- Severity of disease measured on categorical scale: 1 = none, 2 = moderate, 3 = severe.
- Data are frequencies:

Exposure	None	Moderate	Severe
5.8	98	0	0
15.0	51	2	1
21.5	34	6	3
27.5	35	5	8
33.5	32	10	9
39.5	23	7	8
46.0	12	6	10
51.5	4	2	5

Data setup

- Set up data file with one frequency on each line, like this: exposure, response category, frequency.

```
5.8  1 98
15   1 51
15   2 2
15   3 1
21.5 1 34
```

- Don't need to enter zero frequencies.
- Multiple response categories treated as ordered by default.
- Make sure ordering in data is the right one! (I use numbers to keep ordering straight.)

Code

```
data miners;
  infile "miners.dat";
  input exposure severity frequency;

proc logistic;
  class severity;
  freq frequency;
  model severity = exposure;
  output out=miners2 pred=pred;

proc print data=miners2;
```

Note:

- `class` statement turns numbers into ordered response
- `freq` statement ensures frequencies are read as such.

Output part 1

Model Information

Number of Observations Read	22
Number of Observations Used	22
Sum of Frequencies Read	371
Sum of Frequencies Used	371

Response Profile

Ordered Value	severity	Total Frequency
1	1	289
2	2	38
3	3	44

Probabilities modeled are cumulated over the lower Ordered Values.

22 lines in data file; frequencies indicate 371 miners total.

Response profile shows number in each severity category in total.

Output part 2

Testing Global Null Hypothesis: BETA=0

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	88.2432	1	<.0001
Score	80.7246	1	<.0001
Wald	64.5206	1	<.0001

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept 1	1	3.9559	0.4096	93.2527	<.0001
Intercept 2	1	4.8691	0.4437	120.4349	<.0001
exposure	1	-0.0959	0.0119	64.5206	<.0001

Severity of disease definitely depends on exposure. To see how:

Predicted severity probs (edited)

as they depend on exposure:

Obs	exposure	severity	frequency	_LEVEL_	pred
1	5.8	1	98	1	0.96769
2	5.8	1	98	2	0.98678
3	15.0	1	51	1	0.92535
4	15.0	1	51	2	0.96865
9	21.5	1	34	1	0.86920
10	21.5	1	34	2	0.94306
15	27.5	1	35	1	0.78893
16	27.5	1	35	2	0.90306
21	33.5	1	32	1	0.67766
22	33.5	1	32	2	0.83974
27	39.5	1	23	1	0.54181
28	39.5	1	23	2	0.74666
33	46.0	1	12	1	0.38799
34	46.0	1	12	2	0.61241
39	51.5	1	4	1	0.27225
40	51.5	1	4	2	0.48251

Understanding the predicted probs

- Miner with 5.8 years exposure has prob 0.968 of no disease, and prob 0.987 of moderate disease or lower (and prob 1 of severe disease or lower).
- Subtracting: prob of no disease 0.968, moderate disease $0.987 - 0.968 = 0.019$, severe disease $1 - 0.987 = 0.013$.
- Compare with miner with 51.5 years exposure: prob 0.272 of no disease, prob $0.483 - 0.272 = 0.211$ of moderate disease, prob $1 - 0.483 = 0.517$ of severe disease.
- Summary:

Exposure	P(none)	P(moderate)	P(severe)
5.8	0.968	0.019	0.013
27.5	0.789	0.115	0.097
51.5	0.272	0.211	0.517

- Miner with more exposure has higher prob of having worse disease.

Unordered responses

- With unordered (nominal) responses, can use *generalized logit*.
- Example: 735 people, record age and sex (male 0, female 1), which of 3 brands of some product preferred.
- Data in `mlogit.dat` separated by commas.
- Tell SAS that sex and brand numbers only distinguish categories.
- For predictions, get output data set and inspect.

The code

```
data prefs;
  infile "mlogit.dat" delimiter=",";
  input brand sex age;

proc logistic;
  class brand;
  class sex;
  model brand=sex age / link=glogit;
  output out=mlogit2 pred=pred;

proc print data=mlogit2;
```

Output part 1

Model Information

Response Variable	brand
Number of Response Levels	3
Model	generalized logit
Number of Observations Used	735

Response Profile

Ordered Value	brand	Total Frequency
1	1	207
2	2	307
3	3	221

Logits modeled use brand=3 as the reference category.

Output part 2

Testing Global Null Hypothesis: BETA=0

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	185.8502	4	<.0001
Score	163.9538	4	<.0001
Wald	129.7966	4	<.0001

Type 3 Analysis of Effects

Effect	DF	Wald Chi-Square	Pr > ChiSq
sex	2	7.6704	0.0216
age	2	123.3880	<.0001

At least one of sex and age makes a difference to the predicted probs; the bottom table says they both do.

Predicted probabilities (a few)

Obs	brand	sex	age	_LEVEL_	pred
4	1	0	26	1	0.89429
5	1	0	26	2	0.09896
6	1	0	26	3	0.00674
10	1	1	27	1	0.77288
11	1	1	27	2	0.20869
12	1	1	27	3	0.01843
2149	3	0	38	1	0.02598
2150	3	0	38	2	0.23855
2151	3	0	38	3	0.73547
2152	2	1	38	1	0.01623
2153	2	1	38	2	0.25162
2154	2	1	38	3	0.73215

Understanding them

- Many combinations of age, sex and brand-preferred.
- Obs 4, 5 and 6 are for males (sex=0) age 26; prob of preferring brand 1 is 0.894, brand 2 is 0.099, brand 3 is 0.007.
- Summarize whole table from previous page:

Sex	Age	P(prefer 1)	P(prefer 2)	P(prefer 3)
Male	26	0.894	0.099	0.007
Female	27	0.773	0.209	0.018
Male	38	0.026	0.239	0.735
Female	38	0.016	0.252	0.732

- Younger people prefer brand 1, older prefer brand 3.
- Females (a little) less likely to prefer brand 1 and more likely to prefer brand 2. (Sex difference *is* significant.)

Alternative data format

Summarize all people of same brand preference, same sex, same age on one line of data file with frequency on end:

```
1 0 24 1
1 0 26 2
1 0 27 4
1 0 28 4
1 0 29 7
1 0 30 3
...
```

Whole data set in 65 lines not 735!

Code for alternative data format

```
data prefs;
  infile "mlogit2.dat";
  input brand sex age frequency;

proc logistic;
  class brand;
  class sex;
  freq frequency;
  model brand=sex age / link=glogit;
  output out=mlogit2 pred=pred;
```

Add `freq` line in analysis. Output same as before.

Where we are going

Survival analysis

- So far, have seen:
  - response variable counted or measured (regression)
  - response variable categorized (logistic regression)

  and have predicted response from explanatory variables.
- But what if response is time until event (eg. time of survival after surgery)?
- Additional complication: event might not have happened at end of study (eg. patient still alive). But knowing that patient has "not died yet" presumably informative. Such data called *censored*.
- Enter *survival analysis*, in particular the "Cox proportional hazards model".
- Explanatory variables in this context often called *covariates*.

Example: still dancing?

- 12 women who have just started taking dancing lessons are followed for up to a year, to see whether they are still taking dancing lessons (or have quit).
- This might depend on:
  - a treatment (visit to a dance competition)
  - woman's age (at start of study).
- Data:

Months	Dancing	Treatment	Age
1	1	0	16
2	1	0	24
2	1	0	18
3	0	0	27
4	1	0	25
5	1	0	21
11	1	0	55
7	1	1	26
8	1	1	36
10	1	1	38
10	0	1	45
12	1	1	47

About the data

- `months` and `dancing` are kind of combined response:
  - `Months` is number of months a woman was actually observed dancing
  - `dancing` is 1 if woman quit, 0 if still dancing at end of study.
- Treatment is 1 if woman went to dance competition, 0 otherwise.
- Want to do predictions for probabilities of still dancing after 3, 6, 9, 12 months for treatment group and control group, for women of ages 25 and 45.

Doing predictions

Add to data file:

```
3 . 0 25
6 . 0 25
9 . 0 25
12 . 0 25
...
3 . 1 45
6 . 1 45
9 . 1 45
12 . 1 45
```

Gives predicted survival probabilities for 3, 6, 9 and 12 months for (a) woman aged 25 in control group, (b) women aged 45 in treatment group (do other age/treatment combos also).

Censoring variable missing for these: won't affect analysis.

The code

```
data dancers;
  infile "survival1.dat";
  input months dancing treatment age;

proc phreg;
  model months*dancing(0) = age treatment;
  output out=fred survival=s;

proc print data=fred;
```

- Nothing new in reading data.
- Note specification of model: includes both survival time and censoring variable in response, and indication of what value means "censored".
- As ever, predictions saved in output data set, then printed.

The output, edited

Model Information

Data Set	WORK.DANCERS
Dependent Variable	months
Censoring Variable	dancing
Censoring Value(s)	0
Ties Handling	BRESLOW

Number of Observations Read	28
Number of Observations Used	12

Summary of the Number of Event and Censored Values

Total	Event	Censored	Percent Censored
12	10	2	16.67

Output part 2

Testing Global Null Hypothesis: BETA=0

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	21.0016	2	<.0001
Score	14.2093	2	0.0008
Wald	5.5556	2	0.0622

Analysis of Maximum Likelihood Estimates

Parameter	DF	Parameter Estimate	Standard Error	Chi-Square	Pr > ChiSq	Hazard Ratio
age	1	-0.35284	0.14973	5.5532	0.0184	0.703
treatment	1	-4.28283	2.54084	2.8412	0.0919	0.014

- Overall model seems significant.
- Survival depends on age but not apparently on treatment (could be small size of data set or confounding of treatment with age).

Predicted survival probs

Obs	months	dancing	treatment	age	s
13	3	.	0	25	0.87856
14	6	.	0	25	0.56647
15	9	.	0	25	0.00000
16	12	.	0	25	0.00000
17	3	.	1	25	0.99821
18	6	.	1	25	0.99219
19	9	.	1	25	0.00000
20	12	.	1	25	0.00000
21	3	.	0	45	0.99989
22	6	.	0	45	0.99951
23	9	.	0	45	0.14589
24	12	.	0	45	0.00000
25	3	.	1	45	1.00000
26	6	.	1	45	0.99999
27	9	.	1	45	0.97378
28	12	.	1	45	0.08223

Conclusions from predicted probs

- Older women more likely to be still dancing than younger women (compare "profiles" for same treatment group).
- Effect of treatment seems to be to increase prob of still dancing (compare "profiles" for same age for treatment group vs. not)
- Would be nice to see this on a graph.

Another way of doing predictions

Instead of adding lines to data file and creating an output data set, use `baseline` command like this:

```
data dancers;
  infile "survival1.dat";
  input months dancing treatment age;

data mypred;
  input treatment age;
  datalines;
  0 25
  0 45
  1 25
  1 45
;

proc phreg data=dancers;
  model months*dancing(0) = age treatment;
  baseline out=fred covariates=mypred survival=s lower=lcl upper=ucl /
nomean;

proc print data=fred;
```


Results, including CIs

Obs	age	treatment	months	s	lcl	ucl
1	25	0	0	1.00000	.	.
2	25	0	1	0.96633	0.90266	1.00000
3	25	0	2	0.79225	0.60826	1.00000
4	25	0	4	0.63726	0.35919	1.00000
5	25	0	5	0.14748	0.05834	0.37282
6	25	0	7	0.00000	0.00000	1.00000
7	25	0	8	0.00000	0.00000	1.00000
8	25	0	10	0.00000	0.00000	1.00000
9	25	0	11	0.00000	0.00000	1.00000
10	25	0	12	0.00000	.	.
11	45	0	0	1.00000	.	.
12	45	0	1	0.99997	0.99980	1.00000
13	45	0	2	0.99980	0.99895	1.00000
14	45	0	4	0.99961	0.99760	1.00000
15	45	0	5	0.99835	0.99486	1.00000
16	45	0	7	0.75954	0.52629	1.00000
17	45	0	8	0.04468	0.00002	1.00000
18	45	0	10	0.00001	0.00000	1.00000
19	45	0	11	0.00000	0.00000	1.00000
20	45	0	12	0.00000	.	.

The rest

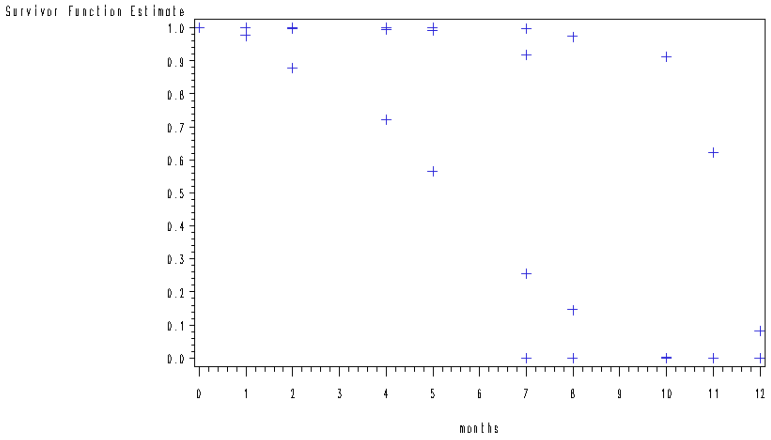
21	25	1	0	1.00000	.	.
22	25	1	1	0.99953	0.99727	1.00000
23	25	1	2	0.99679	0.98545	1.00000
24	25	1	4	0.99380	0.96712	1.00000
25	25	1	5	0.97393	0.92908	1.00000
26	25	1	7	0.01220	0.00080	0.18538
27	25	1	8	0.00000	0.00000	1.00000
28	25	1	10	0.00000	0.00000	1.00000
29	25	1	11	0.00000	0.00000	1.00000
30	25	1	12	0.00000	.	.
31	45	1	0	1.00000	.	.
32	45	1	1	1.00000	1.00000	1.00000
33	45	1	2	1.00000	0.99998	1.00000
34	45	1	4	0.99999	0.99995	1.00000
35	45	1	5	0.99998	0.99990	1.00000
36	45	1	7	0.99621	0.98945	1.00000
37	45	1	8	0.95800	0.88352	1.00000
38	45	1	10	0.84737	0.67929	1.00000
39	45	1	11	0.38657	0.09793	1.00000
40	45	1	12	0.00000	.	.

Plotting survival probabilities

- Each age - treatment combination has string of estimated survival probabilities.
- Would like to plot them against time (month), labelled by which age - treatment combo they are for.
- This *almost* works:

```
proc gplot;
  plot s*months;
```

Graph, attempt 1

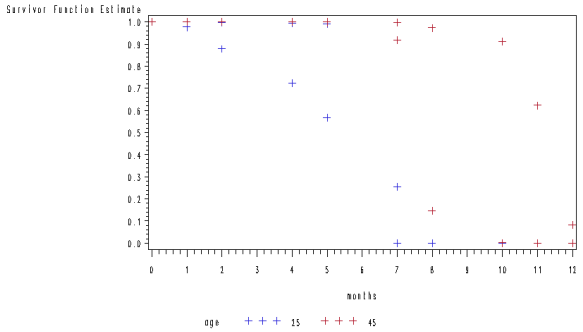


- Can't distinguish treatment - age groups.
- Would like points joined by line, so can see trends.

Graph, attempt 2

If we just had one variable to distinguish, eg age, could do this:

```
proc gplot;
    plot s*months=age;
```



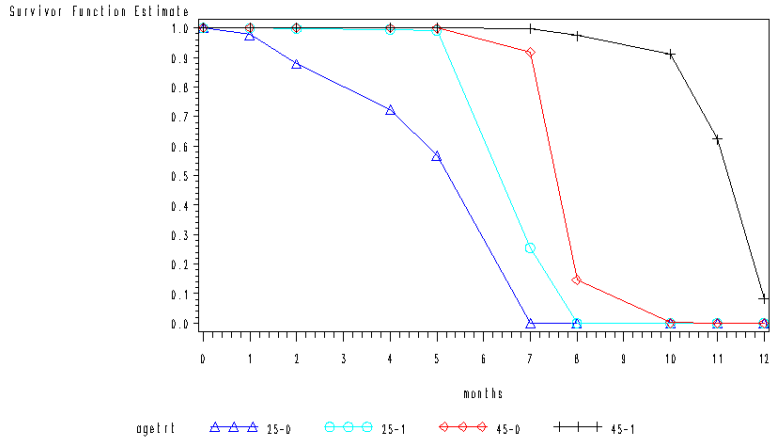
but can't tell treatments apart.

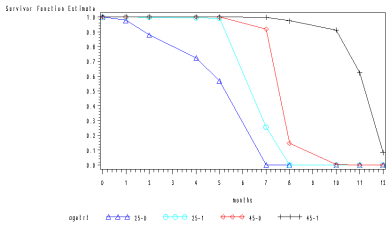
Doing it the right way

```
data fred2;
  set fred;
  agetrt=cat(age,"-",treatment);

goptions reset=all;
symbol1 c=blue v=triangle i=l;
symbol2 c=cyan v=circle i=l;
symbol3 c=red v=diamond i=l;
symbol4 c=black v=plus i=l;

proc gplot;
  plot s*months=agetrt;
```

- Create a new data set that is everything in `fred`, plus new variable: age followed by dash followed by `treatment`.
- Reset SAS's preferred options for `gplot` in favour of stuff we define. 4 age-treatment combos, so define 4 `symbols`:
  - `c=` gives colour.
  - `v=` gives symbol.
  - `i=l` says join plotted points by line.
- When SAS needs symbols for a plot, takes from this list.
- Then make plot, distinguishing age-treatment combos.

The resulting plot





- Survivor curve farther to the right is better (better chance of surviving longer).
- Best is age 45 with treatment, worst age 25 without.
- Appears to be:
  - age effect (45 better than 25)
  - treatment effect (treatment better than not)
- In analysis, treatment effect only marginally significant.

Where we are going

Analysis of variance

- Analysis of variance used with:
  - counted/measured response
  - categorical explanatory variable(s)
  - that is, data divided into groups, and see if response significantly different among groups
  - or, see whether knowing group membership helps to predict response.
- Typically two stages:
  - F-test to detect *any* differences among/due to groups
  - if F-test significant, do *multiple comparisons* to see which groups significantly different from which.
  - Need special multiple comparisons method because just doing (say) two-sample t-tests on each pair of groups gives too big a chance of finding "significant" differences by accident.

Example: jumping rats

- Link between exercise and healthy bones: exercise stresses bones and helps them grow stronger.
- Study assessed effect of jumping on bone density of rats. Rats randomly assigned to one of 3 treatment groups:
  - no jumping (control)
  - low-jump (30 cm)
  - high-jump (60 cm)
- 8 jumps/day, 5 days/week, measure bone density (response) at end.
- PROC GLM to analyze (or PROC ANOVA, only works for balanced designs).

The data

- Some of the data (10 rats in each group). Data separated by tabs.

```
Control     1           603
Control     1           569
...
Lowjump     2           635
Lowjump     2           605
...
Highjump    3           643
Highjump    3           650
```


Code

- Code below. Note format for reading tab-separated data.

```
options linesize=70;

data jumping;
  infile "jumping.dat" delimiter='09'x;
  input group $ g density;

proc means;
  var density;
  class group;

proc glm;
  class group;
  model density=group;
  lsmeans group / adjust=tukey lines;
  lsmeans group / adjust=bon lines;
```


Comments

- "Straightforward" one-way ANOVA.
- Get table of group means and SDs. Assumption: population SD in each group the same, so sample SDs should be "not too different".
- Tukey's method asks: "how far apart might lowest and highest sample group means be, if population means all same?". Anything larger than that declared significantly different.
- Bonferroni's method allows for number of paired comparisons, in general for n groups is $n(n-1)/2$, here 3: divide α by 3 for each test (eg. $0.05/3 = 0.0167$). More "conservative" than Tukey.

Output part 1

Analysis Variable : density

group	N Obs	N	Mean	Std Dev	Minimum	Maximum
Control	10	10	601.1000000	27.3636011	554.0000000	653.0000000
Highjump	10	10	638.7000000	16.5935061	622.0000000	674.0000000
Lowjump	10	10	612.5000000	19.3290225	588.0000000	638.0000000

Dependent Variable: density

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	2	7433.86667	3716.93333	7.98	0.0019
Error	27	12579.50000	465.90741		
Corrected Total	29	20013.36667			

Source	DF	Type I SS	Mean Square	F Value	Pr > F
group	2	7433.866667	3716.933333	7.98	0.0019

Source	DF	Type III SS	Mean Square	F Value	Pr > F
group	2	7433.866667	3716.933333	7.98	0.0019

Notes

- Sample SDs not too different. (Argue that rats were randomly assigned to groups, so population SDs necessarily same.)
- F-tests for model as a whole and for groups (same) significant: there is effect of jumping on bone density. Use multiple comparisons to see what: Tukey then Bonferroni.

Tukey

Tukey Comparison Lines for Least Squares Means of group

LS-means with the same letter are not significantly different.

	density LSMEAN	group	LSMEAN Number
A	638.7	Highjump	2
B	612.5	Lowjump	3
B			
B	601.1	Control	1

High jumping has a significantly different (better) effect on bone density; no significant difference between low jumping and control.

Bonferroni

Bonferroni Comparison Lines for Least Squares Means of group

LS-means with the same letter are not significantly different.

	density LSMEAN	group	LSMEAN Number
A	638.7	Highjump	2
B	612.5	Lowjump	3
B			
B	601.1	Control	1

- Here, same conclusions as before. But...

More of the `lsmeans` output (edited)

group	density LSMEAN	LSMEAN Number
Control	601.100000	1
Highjump	638.700000	2
Lowjump	612.500000	3

Tukey

Pr > |t| for H0: LSMean(i)=LSMean(j)

i/j	1	2	3
1		0.0016	0.4744
2	0.0016		0.0298
3	0.4744	0.0298	

Bonferroni

i/j	1	2	3
1		0.0018	0.7437
2	0.0018		0.0343
3	0.7437	0.0343	

- But P-values for Bonferroni all higher than corresponding ones for Tukey.
- Bonferroni has harder job finding significant differences if they exist.

Another example: scaffolds

- Repair serious wounds by inserting material as "scaffold" for body's repair cells to use as template for new tissue.
- Scaffolds made from extracellular material (ECMs) promising (made from biological material).
- Study: use mice to compare effects of 6 types of material.
- Response: % glucose phosphated isomerase (GPI) cells in region of wound: higher better.
- GPI measured 2, 4, 8 weeks after tissue repair.
- 3 mice for each combo of material (6) and weeks (3): 54 total.
- Data: material, weeks, GPI.
- See whether GPI depends on either/both of material and weeks or their interaction.

Data

```
ecm1  2   70
ecm1  2   75
ecm1  2   65
ecm1  4   55
ecm1  4   70
ecm1  4   70
ecm1  8   60
ecm1  8   65
ecm1  8   65
ecm2  2   60
...
mat3  8   5
mat3  8   15
mat3  8   10
```


Code

```
options linesize=75;

data scaffold;
  infile "scaffold.dat";
  input material $ weeks gpi;

proc glm;
  class material weeks;
  model gpi=material|weeks;
```

- Declare "weeks" as a categorical variable too (look for any differences among weeks), then fit model saying GPI depends on both and interaction too.
- The | between material and weeks means "fit interaction as well as main effects".
- (Looking to see whether interaction significant first, then decide what to do next.)

ANOVA output

The GLM Procedure

Dependent Variable: gpi

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	17	37609.25926	2212.30937	86.88	<.0001
Error	36	916.66667	25.46296		
Corrected Total	53	38525.92593			

...

Source	DF	Type I SS	Mean Square	F Value	Pr > F
material	5	35659.25926	7131.85185	280.09	<.0001
weeks	2	867.59259	433.79630	17.04	<.0001
material*weeks	10	1082.40741	108.24074	4.25	0.0006

Source	DF	Type III SS	Mean Square	F Value	Pr > F
material	5	35659.25926	7131.85185	280.09	<.0001
weeks	2	867.59259	433.79630	17.04	<.0001
material*weeks	10	1082.40741	108.24074	4.25	0.0006

Look at interaction test (bottom line) first: significant, so don't do any other tests. GPI depends on weeks in different way according to materials.

Doing Tukey for interactions

Using `means` in `proc glm`, difficult. But easy enough using `lsmeans`:

```
proc glm;
  model gpi=material|weeks;
  lsmeans material*weeks / adjust=tukey lines;
```

Or `lsmeans material|weeks` appears to do the same thing.

Tukey output

Adjustment for Multiple Comparisons: Tukey

Tukey Comparison Lines for Least Squares Means of material*weeks

LS-means with the same letter are not significantly different.

			gpi LSMEAN	material	weeks	LSMEAN Number
	A		73.333333	ecm3	8	9
	A		73.333333	ecm3	4	8
	A		71.666667	ecm3	2	7
	A		70.000000	ecm1	2	1
	A		65.000000	ecm2	2	4
	A		65.000000	ecm1	4	2
B	A		63.333333	ecm2	8	6
B	A		63.333333	ecm2	4	5
B	A		63.333333	ecm1	8	3
B			48.333333	mat1	2	10
	C		26.666667	mat3	2	16
D	C		23.333333	mat1	4	11
D	C	E	21.666667	mat1	8	12
D	C	E	11.666667	mat3	4	17
D		E	10.000000	mat3	8	18
D		E	10.000000	mat2	2	13
		E	6.666667	mat2	8	15
		E	6.666667	mat2	4	14

Interpretation

- Complicated, because of overlapping lines.
- No sig. differences among ECMs.
- ECMs all better than MATs except mat1 at 2 weeks.
- Other MATs worse, with complicated pattern of significant differences.
- No consistent pattern of which #weeks best for each material (explains significant interaction).
- Next step should be: MAT materials no good, so do another experiment on just ECMs.
- We cheat — extract data for just ECMs!

Just the ECMs: code

First do the same analysis again, checking for significant interaction:

```
data scaffold;
  infile "scaffold2.dat";
  input material $ weeks gpi;

proc glm;
  class material weeks;
  model gpi=weeks|material;
```

Interaction test

The GLM Procedure

Dependent Variable: gpi

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	8	468.518519	58.564815	1.62	0.1874
Error	18	650.000000	36.111111		
Corrected Total	26	1118.518519			

Source	DF	Type I SS	Mean Square	F Value	Pr > F
weeks	2	24.0740741	12.0370370	0.33	0.7209
material	2	385.1851852	192.5925926	5.33	0.0152
material*weeks	4	59.2592593	14.8148148	0.41	0.7989

Source	DF	Type III SS	Mean Square	F Value	Pr > F
weeks	2	24.0740741	12.0370370	0.33	0.7209
material	2	385.1851852	192.5925926	5.33	0.0152
material*weeks	4	59.2592593	14.8148148	0.41	0.7989

No significant interaction (very bottom line), so re-run analysis without (and do Tukey accordingly).

Revised code

Read data as before, and then this:

```
proc glm;
  class material weeks;
  model gpi=weeks material;
  lsmeans material weeks / adjust=tukey lines;
```

- Note lack of | in `model` line, no interaction in `lsmeans` line.
- No interaction means effect of weeks on GPI same for each material, and effect of material on GPI same for each number of weeks.
- So get separate Tukeys to see which materials best, which #weeks best.

The ANOVA

The GLM Procedure

Dependent Variable: gpi

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	4	409.259259	102.314815	3.17	0.0335
Error	22	709.259259	32.239057		
Corrected Total	26	1118.518519			

Source	DF	Type I SS	Mean Square	F Value	Pr > F
weeks	2	24.0740741	12.0370370	0.37	0.6927
material	2	385.1851852	192.5925926	5.97	0.0085

Source	DF	Type III SS	Mean Square	F Value	Pr > F
weeks	2	24.0740741	12.0370370	0.37	0.6927
material	2	385.1851852	192.5925926	5.97	0.0085

Significant effect of materials, but not of #weeks.

Tukey

Tukey Comparison Lines for Least Squares Means of material

LS-means with the same letter are not significantly different.

		gpi LSMEAN	material	LSMEAN Number
	A	72.77778	ecm3	3
	A			
B	A	66.11111	ecm1	1
B				
B		63.88889	ecm2	2

- ecm3 better than ecm2.
- ecm1 in curious middle ground: not sig. worse than ecm3, not sig. better than ecm2.
- Not enough data to resolve this (ecm1 and ecm3 "almost" sig. different: P-value 0.0523).

Tukey for weeks

No sig. difference due to weeks, so shouldn't really even look at Tukey, but results not surprising:

```
    Tukey Comparison Lines for Least Squares Means of weeks

LS-means with the same letter are not significantly different.

                      gpi              LSMEAN
                    LSMEAN    weeks    Number

               A    68.88889  2             1
               A
               A    67.22222  4             2
               A
               A    66.66667  8             3
```


Where we are going

Analysis of covariance

- ANOVA: explanatory variables categorical (divide data into groups)
- traditionally, analysis of covariance has categorical x's plus one numerical x ("covariate") to be adjusted for.
- PROC GLM handles this too.
- Simple example: two treatments (drugs) (a and b), with before and after scores.
  - Does knowing before score and/or treatment help to predict after score?
  - Is after score different by treatment/before score?

Data

Treatment, before, after:

```
a 5 20
a 10 23
a 12 30
a 9 25
a 23 34
a 21 40
a 14 27
a 18 38
a 6 24
a 13 31
b 7 19
b 12 26
b 27 33
b 24 35
b 18 30
b 22 31
b 26 34
b 21 28
b 14 23
b 9 22
```

SAS code

```
data drugs;
  infile "ancova.dat";
  input drug $ before after;

proc means;
  class drug;

proc glm;
  class drug;
  model after = drug before drug*before;
```

- Get means of before and after scores for each treatment.
- Make sure drug treated as categorical ("class")
- Before score treated as numeric by default
- Interaction means "effect of before score on after score is different for each treatment". Fit this first.

The means

The MEANS Procedure

drug	N Obs	Variable	N	Mean	Std Dev
a	10	before	10	13.1000000	6.0452001
		after	10	29.2000000	6.6131183
b	10	before	10	18.0000000	7.1492035
		after	10	28.1000000	5.4660569

- Mean "after" score slightly higher for treatment A.
- Mean "before" score much higher for treatment B.
- Greater *improvement* on treatment A.

Testing for interaction

The GLM Procedure

Dependent Variable: after

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	558.5668744	186.1889581	27.09	<.0001
Error	16	109.9831256	6.8739453		
Corrected Total	19	668.5500000			

Source	DF	Type I SS	Mean Square	F Value	Pr > F
drug	1	6.0500000	6.0500000	0.88	0.3621
before	1	540.1797947	540.1797947	78.58	<.0001
before*drug	1	12.3370798	12.3370798	1.79	0.1991

Source	DF	Type III SS	Mean Square	F Value	Pr > F
drug	1	1.2105592	1.2105592	0.18	0.6803
before	1	552.3578682	552.3578682	80.36	<.0001
before*drug	1	12.3370798	12.3370798	1.79	0.1991

Taking out interaction

- Take out non-significant interaction.
- Assuming linear dependence of after score on before score has *same slope* for both treatments (though possibly different intercept).
- Get predicted means for "after" score depending on drug and before.
- Also get means for treatments "adjusted" for before score.
- Code:

```
proc glm;
  class drug;
  model after = drug before;
  output out=z predict=pred;
  lsmeans drug;

proc print data=z;
```

Results

```
Dependent Variable: after
                                 Sum of
 Source                    DF       Squares   Mean Square  F Value  Pr > F
 Model                      2   546.2297947   273.1148973    37.96  <.0001
 Error                     17   122.3202053     7.1953062
 Corrected Total           19   668.5500000

 Source                    DF     Type I SS   Mean Square  F Value  Pr > F

 drug                       1     6.0500000     6.0500000     0.84  0.3720
 before                     1   540.1797947   540.1797947    75.07  <.0001

 Source                    DF   Type III SS   Mean Square  F Value  Pr > F

 drug                       1   115.3059567   115.3059567    16.03  0.0009
 before                     1   540.1797947   540.1797947    75.07  <.0001
```


Interpreting the output

- Requires care!
- Model as a whole is significant.
- Type I SS says "is each variable significant when added *in order*: that is:
  - `drug` added to a model containing nothing (not sig)
  - `before` added to model containing only `drug` (sig)
- Not really what we want to know.
- Type III SS: "can I take this variable out of a model containing everything?" Answer in both cases no. Interpretation: once you allow for before score, there is a significant difference between treatments. (But if you don't allow for before score, there isn't.)

LS-means

Sample means for each treatment close:

```
                N
drug       Obs    Variable     N           Mean        Std Dev
-------------------------------------------------------------
a           10    after       10     29.2000000      6.6131183
b           10    after       10     28.1000000      5.4660569
-------------------------------------------------------------
```

"Least squares means": mean score for each treatment, after allowing for difference in before scores:

```
                    The GLM Procedure
                   Least Squares Means

drug    after LSMEAN
a         31.2273292
b         26.0726708
```

Treatment A noticeably (significantly) better than B, *once you allow for before score*.

Looking at the predictions

Some of them, arranged in before score order:

Obs	drug	before	pred
4	a	9	25.8073
3	a	12	28.2898
7	a	14	29.9447
8	a	18	33.2547
6	a	21	35.7371
20	b	9	20.6527
12	b	12	23.1351
19	b	14	24.7901
15	b	18	28.1000
18	b	21	30.5824

- Prediction for treatment A about 5 units higher than for treatment B at the same before score — same difference as between LSMEANS.
- Consistent because no interaction.
- If interaction had been included, A might be higher for some before scores and B higher for others: clouds interpretation.

Where we are going

Multivariate analysis of variance

- Standard ANOVA has just one response variable.
- What if you have more than one response?
- Try an ANOVA on each response separately.
- But might miss some kinds of interesting dependence between the responses that distinguish the groups.
- SAS can run MANOVA using an option on PROC GLM.

Small example

- Measure yield and seed weight of plants grown under 2 conditions: low and high amounts of fertilizer.
- Data (fertilizer, yield, seed weight):

```
low 34 10
low 29 14
low 35 11
low 32 13
high 33 14
high 38 12
high 34 13
high 35 14
```

- 2 responses, yield and seed weight.
- First get means by fertilizer amount.
- Then run 1-way ANOVA for each of yield and seed weight, using fertilizer type as explanatory.

Code

```
data manova1;
  infile "manova1.dat";
  input fertilizer $ yield weight;

proc means;
  var yield weight;
  class fertilizer;

proc glm;
  class fertilizer;
  model yield=fertilizer;

proc glm;
  class fertilizer;
  model weight=fertilizer;
```

The means

The MEANS Procedure

fertilizer	N Obs	Variable	N	Mean	Std Dev	Minimum
high	4	yield	4	35.0000000	2.1602469	33.0000000
		weight	4	13.2500000	0.9574271	12.0000000
low	4	yield	4	32.5000000	2.6457513	29.0000000
		weight	4	12.0000000	1.8257419	10.0000000

Means on both variables are slightly higher for high fertilizer. Are those differences significant? Look at ANOVAs (2-sample t-tests would also have worked.)

The ANOVAs

Only one x (fertilizer amount) so look at "model" line.

Dependent Variable: yield

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	12.50000000	12.50000000	2.14	0.1936
Error	6	35.00000000	5.83333333		
Corrected Total	7	47.50000000			

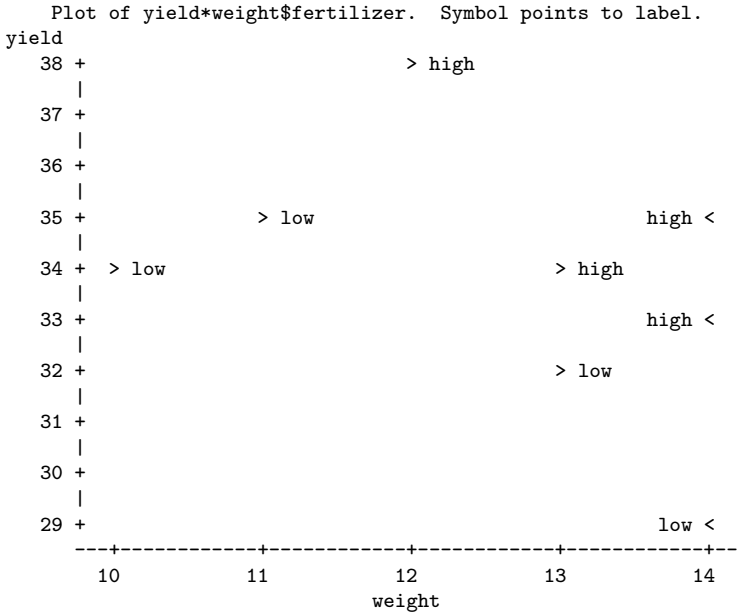
Dependent Variable: weight

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	3.12500000	3.12500000	1.47	0.2708
Error	6	12.75000000	2.12500000		
Corrected Total	7	15.87500000			

Neither mean yield nor mean weight depends on the amount of fertilizer. But: look at plot of yield vs. weight labelled by fertilizer, using this code:

```
proc plot;
  plot yield*weight $ fertilizer;
```


Plot of yield vs. weight



MANOVA code

- High-fertilizer plants have both yield and weight high.
- True even though no sig difference in yield or weight individually.
- Could draw a line separating highs from lows on graph.
- Is *that* significant? MANOVA finds out.
- Code:

```
proc glm;
  class fertilizer;
  model yield weight=fertilizer;
  manova h=_all_;
```

Output

Includes this:

The GLM Procedure
Multivariate Analysis of Variance

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.19845779	10.10	2	5	0.0175
Pillai's Trace	0.80154221	10.10	2	5	0.0175
Hotelling-Lawley Trace	4.03885481	10.10	2	5	0.0175
Roy's Greatest Root	4.03885481	10.10	2	5	0.0175

- Four versions of ANOVA F-test, here all agree: the multivariate difference seen on graph *is* significant.
- With more than 2 responses, cannot draw graph. What then?
- Use *discriminant analysis* (of which more later).

A discriminant analysis

Treat this as "magic" for now, but: obtain output data set and look at it.

```
proc discrim can out=fred;
  class fertilizer;
  var yield weight;

proc print data=fred;
```

Ignore output from PROC DISCRIM, look at output data set.

Output data set

Obs	fertilizer	yield	weight	Can1	Can2	high	low	_INTO_
1	low	34	10	-3.09314	.	0.00002	0.99998	low
2	low	29	14	-1.92110	.	0.00125	0.99875	low
3	low	35	11	-1.07511	.	0.02315	0.97685	low
4	low	32	13	-0.87242	.	0.04579	0.95421	low
5	high	33	14	1.14561	.	0.98180	0.01820	high
6	high	38	12	2.47628	.	0.99982	0.00018	high
7	high	34	13	0.66093	.	0.90893	0.09107	high
8	high	35	14	2.67896	.	0.99991	0.00009	high

- In `Can1`, low value suggests low fertilizer, high suggests high.
- "high" and "low" are estimated probabilities that observation with that yield and weight was high or low fertilizer.
- Last column is SAS's guess at which group it comes from (higher est prob). Got them all right.
- Distinction between high and low quite clear when looked at the right way.
- Procedure works no matter what combination of responses best divides data into groups by x.

Where we are going

Repeated measures by profile analysis

- More than one response *measurement* for each subject. Might be
  - measurements of the same thing at different times
  - measurements of different (but related) things
- Variation: each subject does several different treatments at different times (called *crossover design*).
- Expect measurements on same subject to be correlated, so assumptions of independence will fail.
- Called *repeated measures*. Different approaches, but *profile analysis* uses PROC GLM and looks like MANOVA.

Some fake data

```
a 10 10 9 10
a 11 9 10 11
a 10 11 10 9
b 9 10 12 10
b 11 10 10 8
b 11 10 8 9
```

- 6 subjects; 2 treatments A and B, 4 (repeated) measurements of some response (at 4 different times).
- Nothing much happening:
  - no difference between the treatments (no treatment effect)
  - no trend over time (values just "jumping about randomly" for each subject).
- Expect to see no significant test results.
- Imagine plotting mean response (y-axis) vs. time (x-axis), labelling response by treatment — "profile".

Doing a repeated measures analysis

```
data rm;
  infile "rm1.dat";
  input trt $ y1 y2 y3 y4;

proc glm;
  class trt;
  model y1 y2 y3 y4 = trt / nouni;
  repeated time;
```

- In "model", put the multiple responses to left of =, like MANOVA.
- `nouni` suppresses univariate ANOVAs (not valid/helpful anyway).
- specify that the 4 responses are measurements at different times.
- Output contains 2 MANOVAs and a univariate ANOVA.

Output for the first analysis

Repeated Measures Level Information

Dependent Variable	y1	y2	y3	y4
Level of time	1	2	3	4

MANOVA Test Criteria and Exact F Statistics for the Hypothesis of no time Effect

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.60922541	0.43	3	2	0.7557
Pillai's Trace	0.39077459	0.43	3	2	0.7557
Hotelling-Lawley Trace	0.64142857	0.43	3	2	0.7557
Roy's Greatest Root	0.64142857	0.43	3	2	0.7557

- No trend over time for either treatment. (No evidence that mean responses at different times are different.)
- Next test time by treatment interaction, also non-significant: no overall difference in response over times, so that non-pattern must be same for both treatment groups.

Last ANOVA for first data set

The GLM Procedure
Repeated Measures Analysis of Variance
Tests of Hypotheses for Between Subjects Effects

Source	DF	Type III SS	Mean Square	F Value	Pr > F
trt	1	0.16666667	0.16666667	0.40	0.5614
Error	4	1.66666667	0.41666667		

This tests whether there is a treatment effect, by comparing mean of the 4 response variables for the treatment groups (so is ordinary ANOVA). Not significant either.
Next, change the data to produce a treatment effect but still no time trend:

Data set 2

```
a 10 10 9 10
a 11 9 10 11
a 10 11 10 9
b 11 10 13 11
b 14 12 12 11
b 15 13 9 11
```

- Now treatment B looks to have a slightly higher mean, so we might find a significant treatment effect.
- Still no apparent differences between times, same for each treatment.
- Run same code on this data set (changing only name of data file).

MANOVAs for data set 2

MANOVA Test Criteria and Exact F Statistics
for the Hypothesis of no time Effect

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.17789982	3.08	3	2	0.2546
Pillai's Trace	0.82210018	3.08	3	2	0.2546
Hotelling-Lawley Trace	4.62114125	3.08	3	2	0.2546
Roy's Greatest Root	4.62114125	3.08	3	2	0.2546

MANOVA Test Criteria and Exact F Statistics
for the Hypothesis of no time*trt Effect

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.23153563	2.21	3	2	0.3263
Pillai's Trace	0.76846437	2.21	3	2	0.3263
Hotelling-Lawley Trace	3.31898971	2.21	3	2	0.3263
Roy's Greatest Root	3.31898971	2.21	3	2	0.3263

No significant difference between times (or difference in pattern of responses over time for the treatments. As we guessed.

Between-subjects analysis for data set 2

The GLM Procedure
Repeated Measures Analysis of Variance
Tests of Hypotheses for Between Subjects Effects

Source	DF	Type III SS	Mean Square	F Value	Pr > F
trt	1	20.16666667	20.16666667	30.25	0.0053
Error	4	2.66666667	0.66666667		

Treatment effect we introduced is indeed significant.

Introducing a time effect

Now make another change to data:

```
a 10 10 11 13
a 11 9 12 14
a 10 11 12 12
b 11 10 15 15
b 10 12 14 14
b 12 13 13 15
```

This time responses at times 3 and 4 seem higher, so expect a time effect now. But pattern of responses over time still same for both treatments, so don't expect a treatment-by-time interaction.
Run the same code again.

MANOVAs for data set 3

MANOVA Test Criteria and Exact F Statistics
for the Hypothesis of no time Effect

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.01516477	43.29	3	2	0.0227
Pillai's Trace	0.98483523	43.29	3	2	0.0227
Hotelling-Lawley Trace	64.94230769	43.29	3	2	0.0227
Roy's Greatest Root	64.94230769	43.29	3	2	0.0227

MANOVA Test Criteria and Exact F Statistics
for the Hypothesis of no time*trt Effect

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.31515152	1.45	3	2	0.4332
Pillai's Trace	0.68484848	1.45	3	2	0.4332
Hotelling-Lawley Trace	2.17307692	1.45	3	2	0.4332
Roy's Greatest Root	2.17307692	1.45	3	2	0.4332

- Now a significant time effect.
- Time by treatment interaction still not significant because pattern of change over time same for each treatment.

Still a significant treatment effect

The GLM Procedure
Repeated Measures Analysis of Variance
Tests of Hypotheses for Between Subjects Effects

Source	DF	Type III SS	Mean Square	F Value	Pr > F
trt	1	15.04166667	15.04166667	36.10	0.0039
Error	4	1.66666667	0.41666667		

because Treatment B numbers still bigger than Treatment A.

Finally...

Make one more change to data:

```
a 10 10 14 13
a 11 9 12 14
a 10 11 13 13
b 15 15 11 10
b 14 14 10 12
b 13 15 10 11
```

- Now the time 3 and 4 numbers are bigger for treatment A and smaller for treatment B.
- Effect of time, but different for each treatment.
- So now time by treatment interaction should be significant.

MANOVAs for data set 4

MANOVA Test Criteria and Exact F Statistics
for the Hypothesis of no time Effect

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.44926108	0.82	3	2	0.5913
Pillai's Trace	0.55073892	0.82	3	2	0.5913
Hotelling-Lawley Trace	1.22587719	0.82	3	2	0.5913
Roy's Greatest Root	1.22587719	0.82	3	2	0.5913

MANOVA Test Criteria and Exact F Statistics
for the Hypothesis of no time*trt Effect

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.01797044	36.43	3	2	0.0268
Pillai's Trace	0.98202956	36.43	3	2	0.0268
Hotelling-Lawley Trace	54.64692982	36.43	3	2	0.0268
Roy's Greatest Root	54.64692982	36.43	3	2	0.0268

- Interaction indeed significant: pattern of change over time depends on treatment.
- Main effect not significant because mean scores for each time (over all the data) aren't very different.

There is still a treatment effect

The GLM Procedure
Repeated Measures Analysis of Variance
Tests of Hypotheses for Between Subjects Effects

Source	DF	Type III SS	Mean Square	F Value	Pr > F
trt	1	4.16666667	4.16666667	25.00	0.0075
Error	4	0.66666667	0.16666667		

In summary

- Hard to understand what all the tests are showing, so manipulated data to produce results we could guess (for easier understanding).
- Test of time effect called test for "flatness" of profiles.
- Test of time by treatment(s) interaction called test of "parallelism" of profiles.
- Test of treatment effects called test of "levels".

A more realistic example

- Do subjects from different professions differ in what they think about different leisure activities?
- 3 occupational groups, bellydancers, politicians and administrators; 5 subjects from each group.
- Each subject participates in 4 activities, reading, dancing, TV-watching, skiing; rates satisfaction with each on 10-point scale.
- Data like this. (Scores on activities as listed.)

```
bellydancer 7 10 6 5
bellydancer 8  9 5 7
bellydancer 5 10 5 8
politician  4  4 4 4
politician  6  4 5 3
politician  5  5 5 6
admin       3  1 1 2
admin       5  3 1 5
admin       4  2 2 5
```

- Profession group plays role of treatment, activity plays role of time.

Some means

Group	Reading	Dancing	TV	Skiing	Activities
Bellydancers	6.6	9.4	5.8	7.4	7.3
Politicians	5.0	4.8	5.2	5.3	5.0
Administrators	5.0	2.0	1.8	3.8	3.2
Groups	5.3	5.4	4.3	5.4	5.2

- Mean scores for each activity overall quite similar.
- Mean scores for each profession group very different.
- Bellydancers like dancing; administrators hate everything but reading.
- Are any of these differences significant?

Repeated measures code

- Code:

```
options linesize=75;
data profile;
  infile "profile.dat";
  input group $ read dance tv ski;
proc glm;
  class group;
  model read dance tv ski = group / nouni;
  repeated activity;
```

- group is profession group.
- "repeated" line says that the responses are all "activities".
- "Nouni": omit separate 1-way analyses by activity.

Output (edited)

MANOVA Test Criteria and Exact F Statistics
for the Hypothesis of no activity Effect

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.27913735	8.61	3	10	0.0040
Pillai's Trace	0.72086265	8.61	3	10	0.0040
Hotelling-Lawley Trace	2.58246571	8.61	3	10	0.0040
Roy's Greatest Root	2.58246571	8.61	3	10	0.0040

MANOVA Test Criteria and F Approximations for
the Hypothesis of no activity*group Effect

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.07627855	8.74	6	20	<.0001
Pillai's Trace	1.43341443	9.28	6	22	<.0001
Hotelling-Lawley Trace	5.42784967	8.73	6	11.714	0.0009
Roy's Greatest Root	3.54059987	12.98	3	11	0.0006

NOTE: F Statistic for Roy's Greatest Root is an upper bound.
NOTE: F Statistic for Wilks' Lambda is exact.

Output part 2

- Significant difference in mean scores (for all the subjects) over activities, even though overall means were not that different.
- The pattern of scores over activities is definitely different for each profession group.

Repeated Measures Analysis of Variance
Tests of Hypotheses for Between Subjects Effects

Source	DF	Type III SS	Mean Square	F Value	Pr > F
group	2	172.9000000	86.4500000	44.14	<.0001
Error	12	23.5000000	1.9583333		

- Those different mean scores (over activities) for each profession are very clearly significantly different.

Another example: histamine in dogs

- 8 dogs take part in experiment.
- Dogs randomized to one of 2 different drugs.
- Response: log of blood concentration of histamine 0, 1, 3 and 5 minutes after taking drug. (Repeated measures.)
- Data in dogs2.dat.

The code

```
options linesize=75;

data dogs;
  infile "dogs2.dat";
  input Drug $ x $ lh1 lh2 lh3 lh4;
  avg=(lh1+lh2+lh3+lh4)/4;

proc glm;
  class Drug;
  model lh1 lh2 lh3 lh4 = Drug / nouni;
  repeated Time;
  lsmeans Drug;

proc glm;
  class Drug;
  model avg=Drug;
  lsmeans Drug;
```


Comments on code

- Calculate mean of 4 responses (`avg`).
- Do repeated measures analysis.
- `lsmeans` convenient way to get means on 4 variables for each Drug.
- Also do ordinary ANOVA using average log-histamine level as response, and obtain means.

Output part 1

MANOVA Test Criteria and Exact F Statistics
for the Hypothesis of no Time Effect

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.05012095	25.27	3	4	0.0046
Pillai's Trace	0.94987905	25.27	3	4	0.0046
Hotelling-Lawley Trace	18.95173763	25.27	3	4	0.0046
Roy's Greatest Root	18.95173763	25.27	3	4	0.0046

MANOVA Test Criteria and Exact F Statistics
for the Hypothesis of no Time*Drug Effect

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.10523944	11.34	3	4	0.0200
Pillai's Trace	0.89476056	11.34	3	4	0.0200
Hotelling-Lawley Trace	8.50214058	11.34	3	4	0.0200
Roy's Greatest Root	8.50214058	11.34	3	4	0.0200

Comments and drug-effect analysis

- The histamine levels do change over time, and the pattern of change differs for the 2 drugs (though latter P-value not *very* small).
- Analysis of drug effect:

The GLM Procedure
Repeated Measures Analysis of Variance
Tests of Hypotheses for Between Subjects Effects

Source	DF	Type III SS	Mean Square	F Value	Pr > F
Drug	1	11.52000000	11.52000000	3.13	0.1274
Error	6	22.10263750	3.68377292		

Averaging over time, no significant difference between drugs.

LSMEANS

The GLM Procedure
Least Squares Means

Drug	lh1 LSMEAN	lh2 LSMEAN	lh3 LSMEAN	lh4 LSMEAN
Morphine	-2.89000000	-1.16000000	-1.99750000	-2.32500000
Trimetha	-3.02250000	0.13000000	-0.17250000	-0.50750000

Both drugs show increase (to time 2) then decrease. (Time effect.) Rate of decrease smaller for Trimetha (time-drug interaction effect).

The second PROC GLM, edited

Source	DF	Squares	Mean Square	F Value	Pr > F
Drug	1	2.88000000	2.88000000	3.13	0.1274
Error	6	5.52565938	0.92094323		
Corrected Total	7	8.40565938			

The GLM Procedure
Least Squares Means

Drug	avg LSMEAN
Morphine	-2.09312500
Trimetha	-0.89312500

- P-value identical to last part of repeated measures analysis.
- Drug means look different, but not different enough to be significant.

Where we are going

Multivariate regression

- Ordinary regression has *one* response variable and one or more explanatory.
- Multivariate regression has *more than one* response variable and one or more explanatory.
- Can do regressions of each response separately for all explanatory,
- but ignores interdependence among responses.
- Strategy:
  - use multivariate regression tests to determine what (if anything) happening
  - use individual regressions to understand results of multivariate tests.

Example

- Psychologist wanted to see whether performance on a set of "paired-associate" tests predicted scores on achievement/aptitude tests.
- Paired associate test: students learn to associate two unrelated words and recall the other when one is given, like "cat" and "ladder".
- 5 PA tests, called `n`, `s`, `ns`, `na`, `ss`.
- 3 responses, SAT (Student Achievement Test), PPVT (picture vocabulary test), Raven (progressive matrices test).
- Also recorded: socio-economic status (SES), Lo/Hi, only look at Lo.
- Data in `Rohwer.dat`, first line variable names.

SAS code

Select only `SES='Lo'`, and skip first line. Run multivariate regression, test whether any of the PA tests predict any of responses:

```
data rohwer;
    infile "Rohwer.dat" firstobs=2;
    input group SES $ SAT PPVT Raven n s ns na ss;
    if SES='Lo';

proc reg;
    model SAT PPVT Raven = n s ns na ss;
    mtest;
```

Output includes univariate regressions of each response on all explanatory; only PPVT appears predictable from any PA test scores.

Multivariate test of any association

```
                    The REG Procedure
                      Model: MODEL1
                  Multivariate Test 1

          Multivariate Statistics and F Approximations

                        S=3    M=0.5    N=13.5

Statistic                    Value    F Value    Num DF    Den DF    Pr > F

Wilks' Lambda            0.34316907       2.54        15    80.458    0.0039
Pillai's Trace           0.82528864       2.35        15        93    0.0066
Hotelling-Lawley Trace   1.44875712       2.72        15    49.769    0.0042
Roy's Greatest Root      1.05511542       6.54         5        31    0.0003

        NOTE: F Statistic for Roy's Greatest Root is an upper bound.
```

These strongly significant, more than would guess from individual regressions.

Are x's associated with any y's?

Add more mtest lines; can label to make easier to find in output:

```
proc reg;
    model SAT PPVT Raven = n s ns na ss;
    mtest;
    n: mtest n;
    s: mtest s;
    ns: mtest ns;
    na: mtest na;
    ss: mtest ss;
```

Each test asks whether the x tested is associated with *any* of the y's.

Output (selected)

Multivariate Test: n

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.96164244	0.39	3	29	0.7642
Pillai's Trace	0.03835756	0.39	3	29	0.7642
Hotelling-Lawley Trace	0.03988755	0.39	3	29	0.7642
Roy's Greatest Root	0.03988755	0.39	3	29	0.7642

Multivariate Test: ns

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.77477885	2.81	3	29	0.0570
Pillai's Trace	0.22522115	2.81	3	29	0.0570
Hotelling-Lawley Trace	0.29069088	2.81	3	29	0.0570
Roy's Greatest Root	0.29069088	2.81	3	29	0.0570

Multivariate Test: na

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.73254211	3.53	3	29	0.0271
Pillai's Trace	0.26745789	3.53	3	29	0.0271
Hotelling-Lawley Trace	0.36510923	3.53	3	29	0.0271
Roy's Greatest Root	0.36510923	3.53	3	29	0.0271

s and ss not significant either.

Leave only ns and na

and test them individually:

```
proc reg;
    model SAT PPVT Raven = ns na;
    mtest;
    ns2: mtest ns;
    na2: mtest na;
```

Overall mtest strongly significant, and:

. . . and

Multivariate Test: ns2

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.86310909	1.69	3	32	0.1884
Pillai's Trace	0.13689091	1.69	3	32	0.1884
Hotelling-Lawley Trace	0.15860209	1.69	3	32	0.1884
Roy's Greatest Root	0.15860209	1.69	3	32	0.1884

Multivariate Test: na2

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.68623559	4.88	3	32	0.0066
Pillai's Trace	0.31376441	4.88	3	32	0.0066
Hotelling-Lawley Trace	0.45722550	4.88	3	32	0.0066
Roy's Greatest Root	0.45722550	4.88	3	32	0.0066

So ns not worth keeping after all. Use only na:

The last stage

```
proc reg;
    model SAT PPVT Raven = na;
    na3: mtest;
```

Since only one x, mtest tests its significance with any y.

mtest output

Multivariate Test 1

Multivariate Statistics and Exact F Statistics

S=1 M=0.5 N=15.5

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.53681650	9.49	3	33	0.0001
Pillai's Trace	0.46318350	9.49	3	33	0.0001
Hotelling-Lawley Trace	0.86283396	9.49	3	33	0.0001
Roy's Greatest Root	0.86283396	9.49	3	33	0.0001

Which y's are predicted by na? Look now at individual regressions:

The regressions (edited)

Dependent Variable: SAT

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	1204.38021	1204.38021	2.57	0.1176
Error	35	16379	467.96906		
Corrected Total	36	17583			

Dependent Variable: PPVT

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	2550.78211	2550.78211	28.82	<.0001
Error	35	3097.65032	88.50429		
Corrected Total	36	5648.43243			

Raven

Dependent Variable: Raven

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	39.63250	39.63250	4.55	0.0401
Error	35	305.17831	8.71938		
Corrected Total	36	344.81081			

SAT cannot be predicted from na, but PPVT and Raven *both* can. Might have missed na-Raven relationship otherwise.

Where we are going

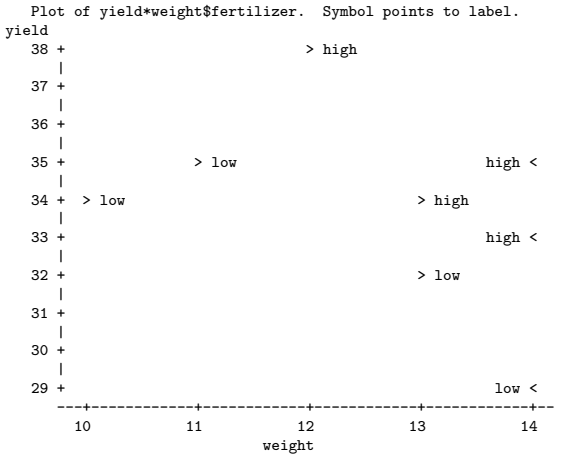
bg

Discriminant analysis

- ANOVA and MANOVA: predict a (counted/measured) response from group membership.
- Discriminant analysis: predict group membership based on counted/measured variables.
- Covers same ground as logistic regression (and its variations), but emphasis on classifying observed data into correct groups.
- Does so by searching for linear combination of original variables that best separates data into groups (canonical variables).
- Assumption here that groups are known (for data we have). If trying to "best separate" data into unknown groups, see *cluster analysis*.
- Examples: revisit seed yield and weight data, professions/activities data; remote-sensing data.

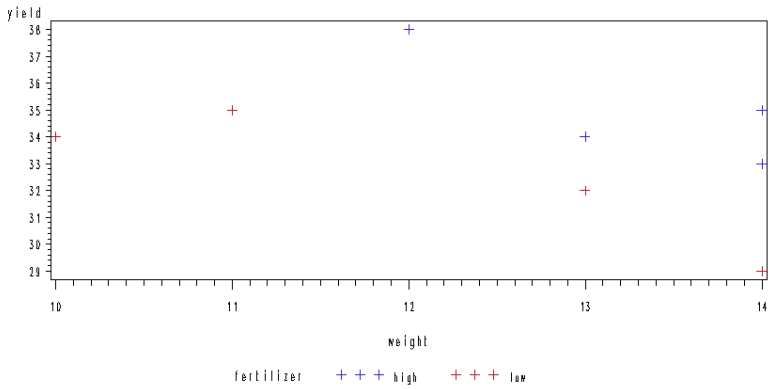
Example 1: seed yields and weights

Recall data from MANOVA: needed a multivariate analysis to find difference in seed yield and weight based on whether they were high or low fertilizer.



Plot variations

Above plot produced with `plot yield * weight $ fertilizer`. Compare `gplot yield * weight = fertilizer`:



Basic PROC DISCRIM

We found it was a *combination* of weight and yield that distinguished high from low fertilizer.

```
data manova1;
  infile "manova1.dat";
  input fertilizer $ yield weight;
proc discrim can list out=x;
  class fertilizer;
  var yield weight;
```

In PROC DISCRIM:

- `can` gets "canonical variables analysis"
- `list` lists observations and summarizes classification
- output data set gives "canonical variable scores" for each observation

Don't need both `list` and output data set; choose according to needs.

Output

The DISCRIM Procedure

Observations	8	DF Total	7
Variables	2	DF Within Classes	6
Classes	2	DF Between Classes	1

Class Level Information

fertilizer	Variable Name	Frequency	Weight	Proportion	Prior Probability
high	high	4	4.0000	0.500000	0.500000
low	low	4	4.0000	0.500000	0.500000

Summarizes input: 8 observations, 2 classes (high and low), 4 observations in each class.

More output

Test of H0: The canonical correlations in the current row and all that follow are zero

	Likelihood Ratio	Approximate F Value	Num DF	Den DF	Pr > F
1	0.19845779	10.10	2	5	0.0175

NOTE: The F statistic is exact.

That is, we really do have $1 + 1 = 2$ groups (the "highs" and "lows" are not all mixed up).

Canonical coefficients

Raw Canonical Coefficients

Variable	Can1
yield	0.766676064
weight	1.251356335

Class Means on Canonical Variables

fertilizer	Can1
high	1.740442790
low	-1.740442790

The combination $0.77\mathit{yield} + 1.25\mathit{weight}$ best separates the highs from the lows. When you do this (and standardize the results: see below) a positive value of Can1 goes with "high" and a negative goes with "low".

Output from "list"

Posterior Probability of Membership in fertilizer

Obs	From fertilizer	Classified into fertilizer	high	low
1	low	low	0.0000	1.0000
2	low	low	0.0012	0.9988
3	low	low	0.0232	0.9768
4	low	low	0.0458	0.9542
5	high	high	0.9818	0.0182
6	high	high	0.9998	0.0002
7	high	high	0.9089	0.0911
8	high	high	0.9999	0.0001

Summary of estimated probabilities that observation with those values of seed yield and seed weight would be classified into each fertilizer category. See that each classification was correct, emphasized below:

Classification summary

Number of Observations and Percent Classified into fertilizer

From fertilizer	high	low	Total
high	4	0	4
	100.00	0.00	100.00
low	0	4	4
	0.00	100.00	100.00
Total	4	4	8
	50.00	50.00	100.00
Priors	0.5	0.5	

Error Count Estimates for fertilizer

	high	low	Total
Rate	0.0000	0.0000	0.0000
Priors	0.5000	0.5000	

Output data set

Finally, the output data set, like the output from `list`, but with more detail:

Obs	fertilizer	yield	weight	Can1	Can2	high	low	_INTO_
1	low	34	10	-3.09314	.	0.00002	0.99998	low
2	low	29	14	-1.92110	.	0.00125	0.99875	low
3	low	35	11	-1.07511	.	0.02315	0.97685	low
4	low	32	13	-0.87242	.	0.04579	0.95421	low
5	high	33	14	1.14561	.	0.98180	0.01820	high
6	high	38	12	2.47628	.	0.99982	0.00018	high
7	high	34	13	0.66093	.	0.90893	0.09107	high
8	high	35	14	2.67896	.	0.99991	0.00009	high

Shows original variable values plus scores on first canonical variable (the one that best separates observations into correct categories). Here `Can1` scaled to have mean 0 (overall) and SD 1 for each group.

Example 2: professions and leisure activities

- Same data we used for profile analysis (some):

```
bellydancer 7 10 6 5
bellydancer 8 9 5 7
bellydancer 5 10 5 8
politician 5 5 5 6
politician 4 5 6 5
admin 4 2 2 5
admin 7 1 2 4
admin 6 3 3 3
```

- How can we best use the scores on the activities to predict a person's profession?
- Or, what combination(s) of scores best separate data into profession groups?

Some SAS code

```
data profile;
  infile "profile.dat";
  input group $ read dance tv ski;

proc discrim can list out=fred;
  class group;

proc print data=fred;

proc gplot data=fred;
  plot Can1 * Can2 = group;
```

Can also specify `read`, `dance`, `tv` and `ski` on a `var` line in PROC DISCRIM; by default all other variables used. (Same idea as PROC MEANS.)
Obtain output data set and plot 1st 2 canonical variables.

Some output

The DISCRIM Procedure

Total Sample Size	15	DF Total	14
Variables	4	DF Within Classes	12
Classes	3	DF Between Classes	2

Number of Observations Read	15
Number of Observations Used	15

Class Level Information

group	Variable Name	Frequency	Weight	Proportion	Prior Probability
admin	admin	5	5.0000	0.333333	0.333333
bellydan	bellydan	5	5.0000	0.333333	0.333333
politici	politici	5	5.0000	0.333333	0.333333

Distances between groups

Generalized Squared Distance to group

From group	admin	bellydan	politici
admin	0	77.68532	25.14460
bellydan	77.68532	0	27.90946
politici	25.14460	27.90946	0

Bellydancers are very different overall from administrators.

Eigenvalues of Inv(E)*H = CanRsq/(1-CanRsq)

	Eigenvalue	Difference	Proportion	Cumulative
1	16.1922	14.2262	0.8917	0.8917
2	1.9660		0.1083	1.0000

2 eigenvalues (it takes 2 lines to divide data into 3 groups), but 1st much bigger than 2nd, so data close to 1-dimensional (see on graph later).

How many canonical variables do I need?

Next table shows this:

Test of H0: The canonical correlations in the current row and all that follow are zero

	Likelihood Ratio	Approximate F Value	Num DF	Den DF	Pr > F
1	0.01961069	13.82	8	18	<.0001
2	0.33715124	6.55	3	10	0.0100

- 1st row says "need at least 1"; 2nd row says "need at least 2".
- Max number of canonical variables is smaller of:
  - number of variables used to assess grouping (4 here)
  - number of groups minus 1 ($3 - 1 = 2$).
- Why: with g groups, $g - 1$ variables separate into that many groups.

What separates the groups

Look at "raw canonical coefficients":

Raw Canonical Coefficients

Variable	Can1	Can2
read	0.012974652	-0.474808056
dance	0.952123961	-0.461497594
tv	0.474172636	1.244632708
ski	-0.041536839	-0.203312237

- 1st canonical variable is mostly attitudes towards dance, with a small amount of attitudes towards TV.
- 2nd is attitudes towards TV-watching contrasted with everything else.
- Bellydancers loved dancing, so Can1 distinguishes them.
- Administrators and bellydancers both hated TV compared to everything else, while politicians indifferent. (Can2 distinguishes politicians.)

Class means on canonical variables

Shows more clearly how the groups differ in terms of Can1 and Can2:

Class Means on Canonical Variables

group	Can1	Can2
admin	-4.347308175	-0.922471653
bellydan	4.466326504	-0.850639955
politici	-0.119018329	1.773111608

- Can1 distinguishes all 3 groups (Can1 close to 0 suggests politician).
- Can2 provides further confirmation that individual is politician. Combo: close to 0 on Can1 *and* positive on Can2 strongly indicates politician.

Output from "list"

. . . shows that groups are pretty separate:

Posterior Probability of Membership in group

Obs	From group	Classified into group	admin	bellydan	politici
1	bellydan	bellydan	0.0000	1.0000	0.0000
2	bellydan	bellydan	0.0000	1.0000	0.0000
3	bellydan	bellydan	0.0000	1.0000	0.0000
4	bellydan	bellydan	0.0000	1.0000	0.0000
5	bellydan	bellydan	0.0000	0.9973	0.0027
6	politici	politici	0.0028	0.0000	0.9972
7	politici	politici	0.0001	0.0000	0.9999
8	politici	politici	0.0000	0.0000	1.0000
9	politici	politici	0.0000	0.0021	0.9979
10	politici	politici	0.0000	0.0000	1.0000
11	admin	admin	1.0000	0.0000	0.0000
12	admin	admin	1.0000	0.0000	0.0000
13	admin	admin	1.0000	0.0000	0.0000
14	admin	admin	1.0000	0.0000	0.0000
15	admin	admin	0.9821	0.0000	0.0179

Classification summary

shows that everyone got classified into the right job:

Number of Observations and Percent Classified into group

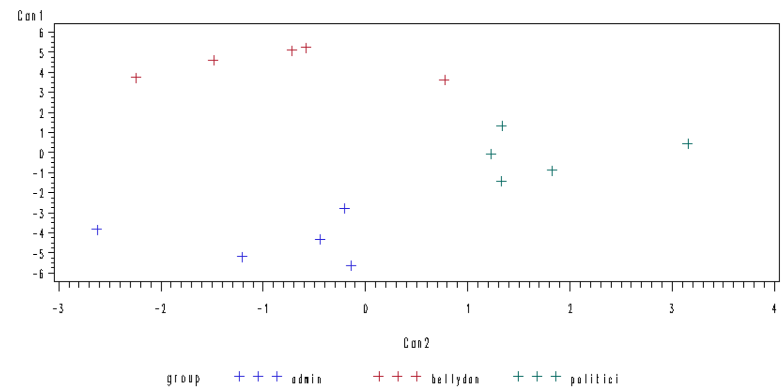
From group	admin	bellydan	politici	Total
admin	5	0	0	5
	100.00	0.00	0.00	100.00
bellydan	0	5	0	5
	0.00	100.00	0.00	100.00
politici	0	0	5	5
	0.00	0.00	100.00	100.00
Total	5	5	5	15
	33.33	33.33	33.33	100.00
Priors	0.33333	0.33333	0.33333	

Output data set

contains a bit more detail (note column names *vertical*):

Obs	group	read	dance	tv	ski	Can1	Can2	Can3	Can4	admin	bellydan	politici	_INTO_
1	bellydan	7	10	6	5	5.23731	-0.58059	.	.	0.00000	1.00000	0.00000	bellydan
2	bellydan	8	9	5	7	3.74092	-2.24515	.	.	0.00000	1.00000	0.00000	bellydan
3	bellydan	5	10	5	8	4.61258	-1.48554	.	.	0.00000	1.00000	0.00000	bellydan
4	bellydan	6	10	6	8	5.09973	-0.71571	.	.	0.00000	1.00000	0.00000	bellydan
5	bellydan	7	8	7	9	3.64109	0.77379	.	.	0.00000	0.99729	0.00271	bellydan
6	politici	4	4	4	4	-1.42116	1.32687	.	.	0.00283	0.00000	0.99717	politici
7	politici	6	4	5	3	-0.87950	1.82520	.	.	0.00008	0.00000	0.99992	politici
8	politici	5	5	5	6	-0.06496	1.22857	.	.	0.00001	0.00000	0.99998	politici
9	politici	6	6	6	7	1.33277	1.33359	.	.	0.00000	0.00214	0.99786	politici
10	politici	4	5	6	5	0.43777	3.15133	.	.	0.00000	0.00000	1.00000	politici
11	admin	3	1	1	2	-5.62995	-0.14110	.	.	1.00000	0.00000	0.00000	admin
12	admin	5	3	1	5	-3.82437	-2.62365	.	.	1.00000	0.00000	0.00000	admin
13	admin	4	2	2	5	-4.31529	-0.44271	.	.	0.99999	0.00000	0.00001	admin
14	admin	7	1	2	4	-5.18696	-1.20233	.	.	1.00000	0.00000	0.00000	admin
15	admin	6	3	3	3	-2.77997	-0.20257	.	.	0.98209	0.00000	0.01791	admin

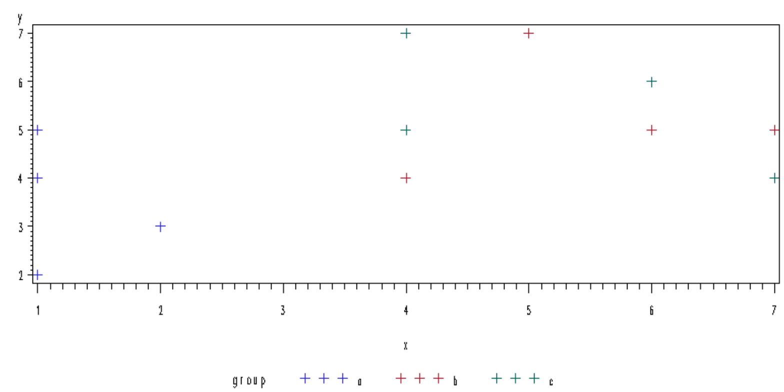
Plotting 1st 2 canonical variables



Comments

- Even though had 4 variables, can plot 1st 2 canonical variables to "see" data. True regardless of number of original variables (though won't see everything if more canonical variables useful).
- See that Can1 separates bellydancers (b) from administrators (a); Can2 separates politicians (p) from rest, and clarifies the position of politicians relative to others.

What if groups aren't all distinct?



SAS code

```
data mix;
  infile "mixup.dat";
  input group $ x y;
proc discrim can list out=xx;
  class group;
  var x y;
proc print;
proc plot;
  plot Can1 * Can2 = group;
```

Original data has 2 variables (x and y), so can be plotted. Perform discriminant analysis with output data set, and plot 1st 2 canonical variables.

Distances

Generalized Squared Distance to group

From group	a	b	c
a	0	18.65441	17.88235
b	18.65441	0	0.06618
c	17.88235	0.06618	0

Groups b and c could be hard to tell apart.

Just one useful canonical variable

Eigenvalues of Inv(E)*H
= CanRsq/(1-CanRsq)

	Eigenvalue	Difference	Proportion	Cumulative
1	5.4098	5.3969	0.9976	0.9976
2	0.0129		0.0024	1.0000

Test of H0: The canonical correlations in the current row and all that follow are zero

	Likelihood Ratio	Approximate F Value	Num DF	Den DF	Pr > F
1	0.15402685	6.19	4	16	0.0033
2	0.98727677	0.12	1	9	0.7412

With 2 variables, can only be max 2, but smallness of eigenvalue and non-significance of test tell us 2nd is not useful.

One variable *might* separate all 3 groups, however.

Canonical variables

Raw Canonical Coefficients

Variable	Can1	Can2
x	0.8252532609	-.3003312927
y	0.4629576531	0.6627706863

1st one is combination of x and y, x weighted more heavily.

Class Means on Canonical Variables

group	Can1	Can2
a	-2.848143534	-0.002552303
b	1.469358718	-0.119111596
c	1.378784816	0.121663899

Can1 separates group a from rest, Can2 doesn't do much of anything. Neither distinguishes groups b and c.

Classification

Posterior Probability of Membership in group

Obs	From group	Classified into group		a	b	c
1	a	a		1.0000	0.0000	0.0000
2	a	a		0.9982	0.0006	0.0012
3	a	a		0.9989	0.0005	0.0006
4	a	a		0.9998	0.0001	0.0002
5	b	c	*	0.0000	0.4387	0.5613
6	b	c	*	0.0961	0.4428	0.4611
7	b	b		0.0000	0.5703	0.4297
8	b	b		0.0000	0.5339	0.4660
9	c	b	*	0.0000	0.5046	0.4954
10	c	b	*	0.0000	0.5989	0.4011
11	c	c		0.0003	0.4028	0.5969
12	c	c		0.0144	0.4539	0.5317

* Misclassified observation

The a's are very clear, but even when b's and c's are correctly classified, it's a very close call.

Classification summary

Doesn't look *so* bad, but overall a third of the 12 observations wrongly classified (and doesn't show how close a call it was).

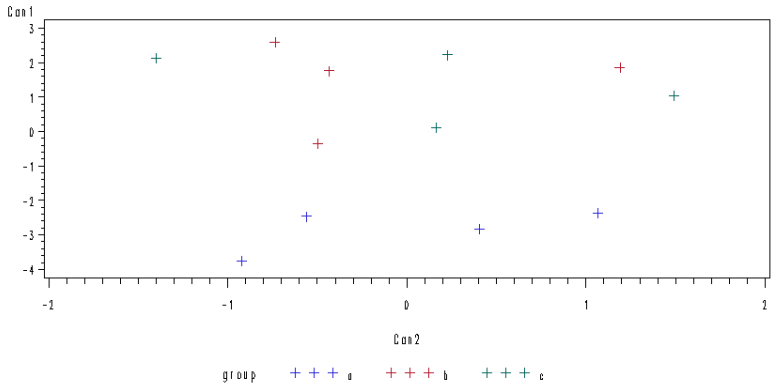
Number of Observations and Percent Classified into group

From group	a	b	c	Total
a	4	0	0	4
	100.00	0.00	0.00	100.00
b	0	2	2	4
	0.00	50.00	50.00	100.00
c	0	2	2	4
	0.00	50.00	50.00	100.00
Total	4	4	4	12
	33.33	33.33	33.33	100.00

Error Count Estimates for group

	a	b	c	Total
Rate	0.0000	0.5000	0.5000	0.3333

Canonical variable plot



Much like original plot, but shows that Can1 distinguishes a's and Can2 doesn't do anything.

Example 3: remote-sensing data

- View 38 crops from air, measure 4 variables `x1-x4`.
- Go back and record what each crop was.
- Can we use the 4 variables to distinguish crops?
- Two new things:
  - (Linear) discriminant analysis assumes "equal covariance matrices", loosely each group has same spread and correlations between all variables. Assumed so far. Can be tested, and if fails, can do *quadratic discriminant analysis*.
  - Using same data to develop discrimination *and* to test performance is optimistic; may not generalize to other data. *Cross-validation* more honest: sees how each observation's group predicted from discriminant analysis based on *rest* of data.
  - SAS can do these. "pooled=yes" means "do linear", "pooled=no" means "do quadratic", "pooled=test" means "do test and do appropriate one". "Crosslist" option means produce classification by cross-validation.

The resulting SAS code, part 1

```
options linesize=75;

data crops;
   infile "remote-sensing.dat";
   input Crop $ x1-x4 label $;

proc discrim can list pool=test out=zz crosslist;
   class Crop;
   var x1-x4;
```


Part 2

```
goptions reset=all;
symbol1 c=blue v=triangle;
symbol2 c=cyan v=circle;
symbol3 c=red v=diamond;
symbol4 c=black v=plus;
symbol5 c=green v=x;

proc gplot;
  plot Can1 * Can2 = Crop;
```

Use different symbols as well as different colours for plotting.

Summary of data

The DISCRIM Procedure

Observations	36	DF Total	35
Variables	4	DF Within Classes	31
Classes	5	DF Between Classes	4

Class Level Information

Crop	Variable Name	Frequency	Weight	Proportion	Prior Probability
Clover	Clover	11	11.0000	0.305556	0.200000
Corn	Corn	7	7.0000	0.194444	0.200000
Cotton	Cotton	6	6.0000	0.166667	0.200000
Soybeans	Soybeans	6	6.0000	0.166667	0.200000
Sugarbee	Sugarbee	6	6.0000	0.166667	0.200000

36 crops, of which 11 (31%) are clover.

Assessing equality of covariance matrices

Within Covariance Matrix Information

Crop	Covariance Matrix Rank	Natural Log of the Determinant of the Covariance Matrix
Clover	4	23.64618
Corn	4	11.13472
Cotton	4	13.23569
Soybeans	4	12.45263
Sugarbee	4	17.76293
Pooled	4	21.30189

If (population) covariance matrices equal, last column should be roughly constant: not plausible here. Formal test:

Chi-Square	DF	Pr > ChiSq
98.022966	40	<.0001

Covariance matrices not equal. So use separate covariance matrices for each crop. (SAS decides with $\alpha = 0.10$).

How many canonical variables?

	Eigenvalue	Difference	Proportion	Cumulative
1	0.6742	0.4925	0.7364	0.7364
2	0.1817	0.1289	0.1985	0.9349
3	0.0528	0.0459	0.0576	0.9925
4	0.0068		0.0075	1.0000

Test of H0: The canonical correlations in the current row and all that follow are zero

	Likelihood Ratio	Approximate F Value	Num DF	Den DF	Pr > F
1	0.47687044	1.48	16	86.179	0.1271
2	0.79837318	0.76	9	70.729	0.6515
3	0.94343017	0.44	4	60	0.7769
4	0.99319917	0.21	1	31	0.6482

4th one has very small eigenvalue: contributes nothing. Indeed, not even first significant. (Look nonetheless at plot of first two.)

Crop means on canonical variables

Class Means on Canonical Variables

Crop	Can1	Can2	Can3	Can4
Clover	0.897881914	0.171142956	-0.159468473	-0.028427125
Corn	-1.154423506	0.297279119	-0.011822020	-0.086854272
Cotton	0.155788168	0.379410840	0.348614473	0.089639679
Soybeans	-0.629213609	-0.299565534	-0.248541709	0.118577501
Sugarbee	0.174136022	-0.740433032	0.206078461	-0.054770800

Can1 distinguishes clover from corn and maybe soybeans. Can2, if anything, picks out sugarbeet.

Classification

Posterior Probability of Membership in Crop

Obs	From Crop	Classified into Crop	Clover	Corn	Cotton	Soybeans	Sugarbee
1	Corn	Corn	0.0097	0.9810	0.0000	0.0000	0.0093
2	Corn	Corn	0.0010	0.9946	0.0000	0.0000	0.0045
3	Corn	Corn	0.0015	0.9809	0.0000	0.0000	0.0177
4	Corn	Corn	0.0068	0.9815	0.0000	0.0024	0.0093
5	Corn	Corn	0.0039	0.9835	0.0000	0.0000	0.0126
6	Corn	Corn	0.0044	0.9424	0.0000	0.0000	0.0532
7	Corn	Corn	0.0008	0.9992	0.0000	0.0000	0.0000
8	Soybeans	Soybeans	0.0053	0.0033	0.0000	0.9821	0.0092
9	Soybeans	Soybeans	0.0143	0.0000	0.0014	0.7647	0.2196
10	Soybeans	Soybeans	0.0034	0.0000	0.0002	0.9896	0.0068
11	Soybeans	Soybeans	0.0058	0.0000	0.0000	0.9854	0.0088
12	Soybeans	Soybeans	0.0072	0.0000	0.0000	0.9921	0.0007
13	Soybeans	Soybeans	0.0149	0.0000	0.0000	0.9850	0.0001
14	Cotton	Cotton	0.0157	0.0000	0.9718	0.0032	0.0093
15	Cotton	Cotton	0.0198	0.0000	0.7925	0.0004	0.1873
16	Cotton	Cotton	0.0290	0.0000	0.9590	0.0000	0.0120
17	Cotton	Cotton	0.0067	0.0000	0.9407	0.0446	0.0080
18	Cotton	Cotton	0.0051	0.0000	0.9949	0.0000	0.0000
19	Cotton	Cotton	0.0024	0.0000	0.9976	0.0000	0.0000

The rest

Obs	From Crop	Classified into Crop	Clover	Corn	Cotton	Soybeans	Sugarbee
20	Sugarbee	Soybeans *	0.0255	0.0000	0.0000	0.8227	0.1518
21	Sugarbee	Cotton *	0.0112	0.0000	0.5014	0.4366	0.0507
22	Sugarbee	Sugarbee	0.0422	0.0000	0.0000	0.0000	0.9578
23	Sugarbee	Sugarbee	0.1705	0.0000	0.0000	0.0000	0.8295
24	Sugarbee	Sugarbee	0.1207	0.0000	0.0000	0.0131	0.8663
25	Sugarbee	Sugarbee	0.0052	0.0000	0.0000	0.0000	0.9948
26	Clover	Clover	1.0000	0.0000	0.0000	0.0000	0.0000
27	Clover	Clover	0.9470	0.0000	0.0000	0.0001	0.0529
28	Clover	Clover	1.0000	0.0000	0.0000	0.0000	0.0000
29	Clover	Clover	0.9790	0.0000	0.0000	0.0000	0.0210
30	Clover	Clover	1.0000	0.0000	0.0000	0.0000	0.0000
31	Clover	Clover	1.0000	0.0000	0.0000	0.0000	0.0000
32	Clover	Sugarbee *	0.1612	0.0000	0.0000	0.0000	0.8388
33	Clover	Sugarbee *	0.1885	0.0000	0.0000	0.0000	0.8115
34	Clover	Clover	1.0000	0.0000	0.0000	0.0000	0.0000
35	Clover	Clover	1.0000	0.0000	0.0000	0.0000	0.0000
36	Clover	Clover	1.0000	0.0000	0.0000	0.0000	0.0000

Only 4 crops misclassified.

Misclassification summary

Number of Observations and Percent Classified into Crop

From Crop	Clover	Corn	Cotton	Soybeans	Sugarbee	Total
Clover	9	0	0	0	2	11
	81.82	0.00	0.00	0.00	18.18	100.00
Corn	0	7	0	0	0	7
	0.00	100.00	0.00	0.00	0.00	100.00
Cotton	0	0	6	0	0	6
	0.00	0.00	100.00	0.00	0.00	100.00
Soybeans	0	0	0	6	0	6
	0.00	0.00	0.00	100.00	0.00	100.00
Sugarbee	0	0	1	1	4	6
	0.00	0.00	16.67	16.67	66.67	100.00
Total	9	7	7	7	6	36
	25.00	19.44	19.44	19.44	16.67	100.00

Error Count Estimates for Crop

	Clover	Corn	Cotton	Soybeans	Sugarbee	Total
Rate	0.1818	0.0000	0.0000	0.0000	0.3333	0.1030

2 clover were classified as sugarbeet; 2 sugarbeet were classified as something else.

Cross-validation results are quite different

Posterior Probability of Membership in Crop

Obs	From Crop	Classified into Crop		Clover	Corn	Cotton	Soybeans	Sugarbee
1	Corn	Clover	*	0.5114	0.0000	0.0000	0.0000	0.4886
2	Corn	Corn		0.0014	0.9921	0.0000	0.0000	0.0065
3	Corn	Corn		0.0023	0.9699	0.0000	0.0000	0.0277
4	Corn	Sugarbee	*	0.3692	0.0000	0.0000	0.1291	0.5017
5	Corn	Sugarbee	*	0.2362	0.0004	0.0000	0.0000	0.7634
6	Corn	Sugarbee	*	0.0753	0.0190	0.0000	0.0000	0.9057
7	Corn	Clover	*	0.9998	0.0000	0.0000	0.0000	0.0002
8	Soybeans	Soybeans		0.0257	0.0161	0.0000	0.9136	0.0446
9	Soybeans	Sugarbee	*	0.0606	0.0000	0.0059	0.0000	0.9334
10	Soybeans	Soybeans		0.0065	0.0000	0.0003	0.9803	0.0129
11	Soybeans	Sugarbee	*	0.3965	0.0000	0.0000	0.0000	0.6035
12	Soybeans	Clover	*	0.9171	0.0000	0.0000	0.0000	0.0829
13	Soybeans	Clover	*	0.9944	0.0000	0.0000	0.0000	0.0056
14	Cotton	Cotton		0.1428	0.0000	0.7439	0.0291	0.0842
15	Cotton	Sugarbee	*	0.0954	0.0000	0.0000	0.0021	0.9025
16	Cotton	Clover	*	0.7066	0.0000	0.0000	0.0000	0.2934
17	Cotton	Cotton		0.0159	0.0000	0.8595	0.1056	0.0190
18	Cotton	Clover	*	1.0000	0.0000	0.0000	0.0000	0.0000
19	Cotton	Clover	*	1.0000	0.0000	0.0000	0.0000	0.0000

The rest

Obs	From Crop	Classified into Crop		Clover	Corn	Cotton	Soybeans	Sugarbee
20	Sugarbee	Soybeans	*	0.0300	0.0000	0.0000	0.9700	0.0000
21	Sugarbee	Cotton	*	0.0118	0.0000	0.5282	0.4599	0.0000
22	Sugarbee	Sugarbee		0.0694	0.0000	0.0000	0.0000	0.9306
23	Sugarbee	Clover	*	1.0000	0.0000	0.0000	0.0000	0.0000
24	Sugarbee	Clover	*	0.9023	0.0000	0.0000	0.0977	0.0000
25	Sugarbee	Clover	*	1.0000	0.0000	0.0000	0.0000	0.0000
26	Clover	Clover		1.0000	0.0000	0.0000	0.0000	0.0000
27	Clover	Clover		0.5477	0.0000	0.0000	0.0008	0.4514
28	Clover	Clover		1.0000	0.0000	0.0000	0.0000	0.0000
29	Clover	Clover		0.9694	0.0000	0.0000	0.0000	0.0306
30	Clover	Clover		1.0000	0.0000	0.0000	0.0000	0.0000
31	Clover	Clover		1.0000	0.0000	0.0000	0.0000	0.0000
32	Clover	Sugarbee	*	0.0441	0.0000	0.0000	0.0000	0.9559
33	Clover	Sugarbee	*	0.1352	0.0000	0.0000	0.0000	0.8648
34	Clover	Clover		1.0000	0.0000	0.0000	0.0000	0.0000
35	Clover	Clover		1.0000	0.0000	0.0000	0.0000	0.0000
36	Clover	Clover		1.0000	0.0000	0.0000	0.0000	0.0000

A lot of misclassifications, and in some cases the estimated probabilities are quite low.

Cross-validation misclassification error summary

Number of Observations and Percent Classified into Crop

From Crop	Clover	Corn	Cotton	Soybeans	Sugarbee	Total
Clover	9	0	0	0	2	11
	81.82	0.00	0.00	0.00	18.18	100.00
Corn	2	2	0	0	3	7
	28.57	28.57	0.00	0.00	42.86	100.00
Cotton	3	0	2	0	1	6
	50.00	0.00	33.33	0.00	16.67	100.00
Soybeans	2	0	0	2	2	6
	33.33	0.00	0.00	33.33	33.33	100.00
Sugarbee	3	0	1	1	1	6
	50.00	0.00	16.67	16.67	16.67	100.00
Total	19	2	3	3	9	36
	52.78	5.56	8.33	8.33	25.00	100.00

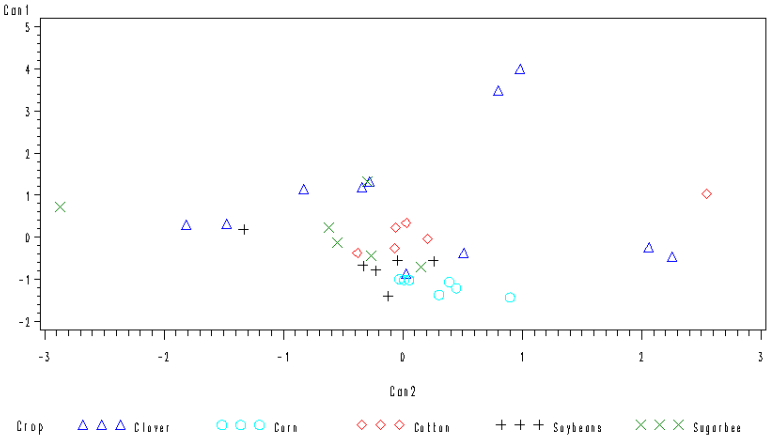
Error Count Estimates for Crop

	Clover	Corn	Cotton	Soybeans	Sugarbee	Total
Rate	0.1818	0.7143	0.6667	0.6667	0.8333	0.6126

61% of the crops are misclassified this more honest way! Sugarbeet especially hard to get right.

Plot of 1st 2 canonical variables

Perhaps surprising that *any* method got much right!



Can1 distinguishes Corn and sometimes Clover.

Where we are going

Cluster Analysis

- One side-effect of discriminant analysis: could draw picture of data (if 1st 2 canonical variables told most of story) and see which individuals "close" to each other.
- Discriminant analysis requires knowledge of groups.
- Without knowledge of groups, use *cluster analysis*: see which individuals close, which groups suggested by data.
- Idea: see how individuals group into "clusters" of nearby individuals.
- Base on "dissimilarities" between individuals.
- Or base on standard deviations and correlations between variables (assesses dissimilarity behind scenes).

One to ten in 11 languages

	English	Norwegian	Danish	Dutch	German
1	one	en	en	een	eins
2	two	to	to	twee	zwei
3	three	tre	tre	drie	drei
4	four	fire	fire	vier	vier
5	five	fem	fem	vijf	funf
6	six	seks	seks	zes	sechs
7	seven	sju	syv	zeven	sieben
8	eight	atte	otte	acht	acht
9	nine	ni	ni	negen	neun
10	ten	ti	ti	tien	zehn

One to ten

	French	Spanish	Italian	Polish	Hungarian	Finnish
1	un	uno	uno	jeden	egy	yksi
2	deux	dos	due	dwa	ketto	kaksi
3	trois	tres	tre	trzy	harom	kolme
4	quatre	cuatro	quattro	cztery	negy	nelja
5	cinq	cinco	cinque	piec	ot	viisi
6	six	seis	sei	szesc	hat	kuusi
7	sept	siete	sette	siedem	het	seitseman
8	huit	ocho	otto	osiem	nyolc	kahdeksan
9	neuf	nueve	nove	dziewiec	kilenc	yhdeksan
10	dix	diez	dieci	dziesiec	tiz	kymmenen

Dissimilarities and languages example

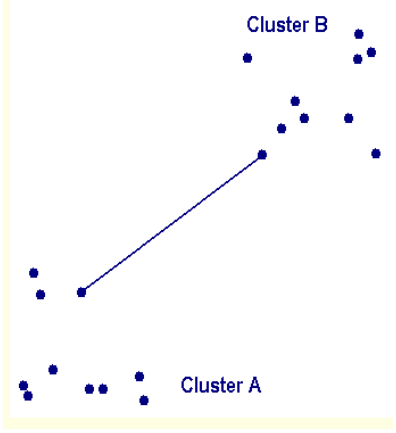
- Can define dissimilarities how you like (whatever makes sense in application).
- Sometimes defining "similarity" makes more sense; can turn this into dissimilarity by subtracting from some maximum.
- Example: numbers 1–10 in various European languages. Define similarity between two languages by counting how often the same number has a name starting with the same letter (and dissimilarity by how often number has names starting with different letter).
- Crude (doesn't even look at most of the words), but see how effective.

Two kinds of cluster analysis

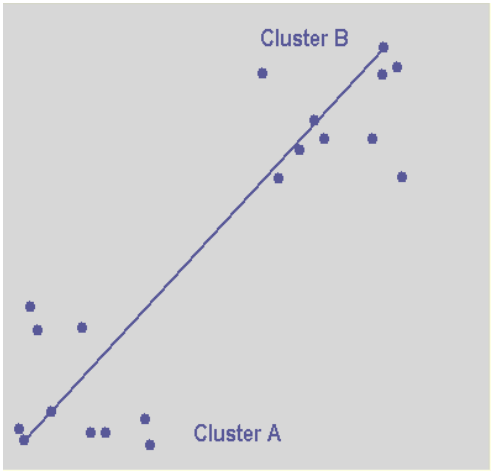
- Looking at process of forming clusters (of similar languages): PROC CLUSTER, hierarchical cluster analysis.
  - Start with each individual in cluster by itself.
  - Join "closest" clusters one by one until all individuals in one cluster.
  - Rule to join clusters: single-linkage, complete linkage, Ward's method, etc.
- Know how many clusters: which division into that many clusters is "best" for individuals? PROC FASTCLUS, K-means clustering.

Hierarchical cluster analysis: joining rules

Join the two clusters that are "closest", but how to define?
Single-linkage (from `http://www.resample.com`)



Complete linkage



Also average linkage (obvious?)

Ward's method example

- Easiest to illustrate how Ward's method works by example.
- Data (one variable): 1, 2, 3, 7, 8, 9, 11, 12, 13. Suppose currently have 3 clusters 1,2,3; 7,8,9; 11,12,13. Measure dissimilarity by absolute difference (throw away minus sign).
- Which 2 of these 3 clusters to join together?
- Single-linkage distances: $7-3=4$, $11-3=8$, $11-9=2$; join 2nd and 3rd.
- Complete-linkage distances: $9-1=8$, $13-1=12$, $13-7=6$; also join 2nd and 3rd.

. . . continued

- Suppose join 1st 2 clusters. Joined cluster has mean $(1+2+3+7+8+9)/6=5$; new sum of squared distances from mean $(1-5)^2+(2-5)^2+(3-5)^2+(7-5)^2+(8-5)^2+(9-5)^2=58$.
- Join 1st and 3rd (obviously bad idea): mean now 7, sum of squared distances $(1-7)^2+(2-7)^2+(3-7)^2+(11-7)^2+(12-7)^2+(13-7)^2=154$.
- Join 2nd and 3rd: mean now 10, sum of squared distances $(7-10)^2+(8-10)^2+(9-10)^2+(11-10)^2+(12-10)^2+(13-10)^2=28$.
- Smallest of these three sums is 28, so join 2nd and 3rd clusters.
- Much computation, especially early with many clusters. But we don't care!

Ward's method in general

- Work out sum of squared distances/dissimilarities from each observation to centre of its current cluster. Like error SS in ANOVA. Call it ESS.
- At start, each point in own cluster, so ESS 0.
- At each stage, join the two clusters that make resulting ESS smallest.
- Favours joining small clusters.
- Like linkage methods, joins "similar" clusters.

Dissimilarity data in SAS

Dissimilarities for language data (first line for reference, not in data file):

```
   en no dk nl de fr es it pl hu sf
en 0  2  2  7  6  6  6  6  7  9  9
no 2  0  1  5  4  6  6  6  7  8  9
dk 2  1  0  6  5  6  5  5  6  8  9
nl 7  5  6  0  5  9  9  9 10  8  9
de 6  4  5  5  0  7  7  7  8  9  9
fr 6  6  6  9  7  0  2  1  5 10  9
es 6  6  5  9  7  2  0  1  3 10  9
it 6  6  5  9  7  1  1  0  4 10  8
pl 7  7  6 10  8  5  3  4  0 10  9
hu 9  8  8  8  9 10 10 10 10  0  8
sf 9  9  9  9  9  9  9  8  9  8  0
```

SAS has special `type=distance` for data like these:

```
data lang(type=distance);
infile "one-ten.dat";
input lang $ en no dk nl de fr es it pl hu sf;
```

Variable `lang` has names of languages; variable names given on `input` line must match.

Doing a hierarchical cluster analysis

- Here, interested in clustering *process* more than final result, so hierarchical analysis appropriate: PROC CLUSTER.
- Choose single-linkage method for combining clusters (that is, combine clusters whose closest members are closest).
- Draw clustering "tree" from output data set. Trees by default vertical.

```
proc cluster method=single outtree=tree;
  id lang;

proc tree horizontal;
  id lang;
```


Output: cluster history

```
          The CLUSTER Procedure
       Single Linkage Cluster Analysis

  Mean Distance Between Observations    6.672727
```

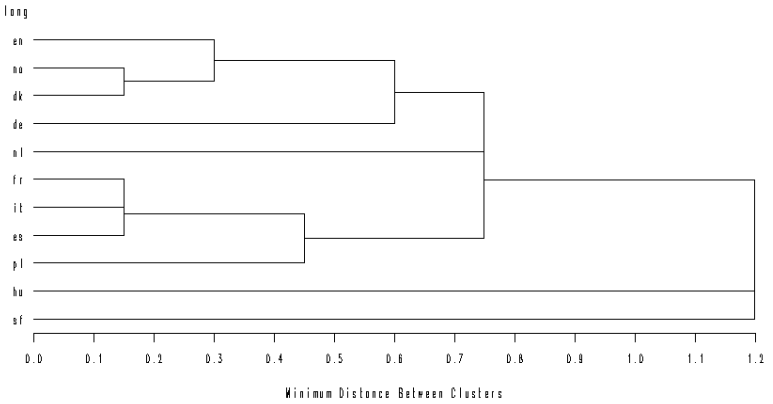
Cluster History

NCL	--Clusters Joined---		FREQ	Norm Min Dist	T i e
10	no	dk	2	0.1499	T
9	fr	it	2	0.1499	T
8	CL9	es	3	0.1499	
7	en	CL10	3	0.2997	
6	CL8	pl	4	0.4496	
5	CL7	de	4	0.5995	
4	CL5	nl	5	0.7493	T
3	CL4	CL6	9	0.7493	
2	CL3	hu	10	1.1989	T
1	CL2	sf	11	1.1989	

Summary of clustering history

- Join Norwegian and Danish.
- Join French and Italian.
- Join Spanish to the French-Italian cluster.
- Join English to the Norwegian-Danish cluster.
- Then: German and Dutch joined to Germanic languages cluster, Polish to Romance language cluster (!)
- Then join these two clusters together, and join Hungarian and Finnish to them.

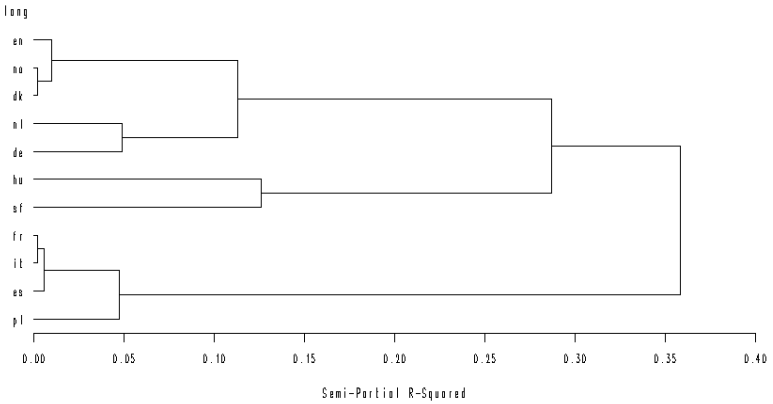
Output from PROC TREE



Checking our intuition about languages

- Have a Germanic cluster (English, Norwegian, Danish, German, Dutch)
- Have a Romance cluster (French, Italian, Spanish, maybe Polish)
- Have two odd languages (Hungarian, Finnish).
- Corresponds to linguistics/geography pretty well (for such a crude measure).
- Maybe Dutch joins Germanic cluster late. Dutch number words much like German, but often happen not to start with same letter.
- Clustering method: single linkage may join languages that happen to have words starting with same letter, but not otherwise similar. Ward's method joins clusters that are more "alike". Change "method=" on PROC DISCRIM line.

Tree from Ward's method



Comparing single-linkage and Ward

- In Ward, Dutch and German get joined earlier (before joining to Germanic cluster).
- Also Hungarian and Finnish get combined earlier.
- Consider which clustering method makes sense for data like these.

Another example

Birth, death and infant mortality rates for 96 countries (variables not dissimilarities):

```
  24.7  5.7  30.8 Albania              12.5 11.9  14.4 Bulgaria
  13.4 11.7  11.3 Czechoslovakia       12   12.4   7.6 Former_E._Germany
  11.6 13.4  14.8 Hungary              14.3 10.2    16 Poland
  13.6 10.7  26.9 Romania                14    9  20.2 Yugoslavia
  17.7   10    23 USSR                 15.2  9.5  13.1 Byelorussia_SSR
  13.4 11.6    13 Ukrainian_SSR        20.7  8.4  25.7 Argentina
  46.6   18   111 Bolivia              28.6  7.9    63 Brazil
  23.4  5.8  17.1 Chile                27.4  6.1    40 Columbia
  32.9  7.4    63 Ecuador              28.3  7.3    56 Guyana
...
```

- Want to find groups of similar countries (and how many groups, which countries in each group).
- Tree would be unwieldy with 96 countries.
- More automatic way of finding number of clusters?
- Two countries per line: how to read into SAS?

SAS code and issues

```
data birthrate;
  infile "birthrate.dat";
  input birth death infant country $ @@;

proc cluster method=average ccc standard;
  id country;
```

- In DATA step, `@@` means "continue reading on same line".
- Using average linkage.
- "CCC" is "cubic clustering criterion", helps us decide how many clusters.
- "standard" means to use standardized data (scaled to have mean 0 and SD 1) so each variable truly comparable.

Clustering history (a little)

96 lines, just show some:

Cluster History

NCL	--Clusters Joined---		FREQ	SPRSQ	RSQ	ERSQ	CCC	Norm RMS Dist	Time
96	Austria	Canada	2	0.0000	1.00	.	.	0.0165	
95	Czechosl	Ukrainia	2	0.0000	1.00	.	.	0.0175	
...									
20	CL82	CL34	6	0.0016	.967	.	.	0.2664	
19	CL32	CL38	7	0.0018	.965	.952	4.10	0.2709	
18	Bolivia	CL29	6	0.0011	.964	.949	4.53	0.2794	
17	CL21	Oman	6	0.0014	.963	.945	4.87	0.3191	
16	CL23	CL26	16	0.0059	.957	.942	3.84	0.3225	
...									
8	CL12	CL74	24	0.0067	.907	.887	2.16	0.4773	
7	Mexico	Korea	2	0.0026	.904	.873	3.27	0.5037	
6	Afghanis	CL13	8	0.0045	.900	.854	4.47	0.5328	
5	CL15	CL10	45	0.0517	.848	.827	1.57	0.5697	
4	CL9	CL8	42	0.1001	.748	.788	-2.3	0.7742	
3	CL5	CL4	87	0.3980	.350	.723	-12	1.0708	
2	CL3	CL7	89	0.0385	.311	.593	-6.8	1.1662	
1	CL2	CL6	97	0.3114	.000	.000	0.00	1.5693	

Look for large values of CCC compared to neighbours, here 17 clusters or 6. We'll try 6.

The 6 best clusters

- Only purpose for running previous analysis was to get good number of clusters.
- 6 clusters obtained by "chopping the tree" may not be best division of countries into 6 clusters.
- Do better by deciding on 6 (or however many) clusters first, *then* trying for best division of countries into 6 clusters.
- This is where K-means clustering comes in. Choose best division of individuals (countries) into K (6) clusters so that sum of squared distances from individuals to cluster averages made smallest (over all possible divisions into K clusters).
- Use PROC FASTCLUS (which does not have "standard" option so have to standardize first).

Code

```
proc standard mean=0 std=1;

proc fastclus maxclusters=6 out=clust;
  id country;

proc sort data=clust;
  by cluster;

proc print data=clust;
  by cluster;
```

Sort data by cluster and print sorted data.

Cluster means and SDs

Cluster Means

Cluster	birth	death	infant
1	-0.435769031	-1.143859869	-0.728110805
2	1.204946595	0.697233337	1.016509747
3	1.301924159	2.117634622	1.866220472
4	-0.219972241	2.111657686	-0.454443499
5	-1.173710389	-0.185637473	-0.953436985
6	0.416099253	-0.516998811	0.264875362

Cluster Standard Deviations

Cluster	birth	death	infant
1	0.3560992452	0.3384785179	0.2086886380
2	0.2838078359	0.3886873578	0.4595354494
3	0.2072519523	0.4982442191	0.4178547653
4	0.2870875322	0.7759545638	0.2767385711
5	0.1523496837	0.3449633244	0.1225870222
6	0.3884813426	0.2398267650	0.4102515861

Cluster membership

---------- Cluster=1 ----------

Obs	birth	death	infant	country	DISTANCE
1	-0.33439	-1.10513	-0.52402	Albania	0.17862
2	-0.62967	-0.52417	-0.63491	Argentin	0.53740
3	-0.43036	-1.08361	-0.82189	Chile	0.18142
4	-0.13508	-1.01906	-0.32399	Columbia	0.42671
5	-0.12770	-1.38485	-0.68709	Venezuel	0.46416
6	-0.06126	-1.51395	-0.84581	Bahrain	0.63514
7	-0.51156	-0.97603	-0.98279	Israel	0.34370
8	-0.17937	-1.85822	-0.85451	Kuwait	0.89881
9	-0.47465	-1.51395	-0.62838	United_A	0.51244
10	-0.59276	-0.88996	-0.49793	China	0.27987
11	-1.29403	-1.27727	-1.06106	Hong_Kon	1.01806
12	0.17496	-1.12665	-0.67187	Malaysia	0.58493
13	-0.84374	-1.21271	-1.03062	Singapor	0.61480
14	-0.58538	-0.99754	-0.77189	Sri_Lank	0.21867
15	-0.51156	-0.67479	-0.58490	Thailand	0.36033

Cluster 2

---------- Cluster=2 ----------

Obs	birth	death	infant	country	DISTANCE
16	0.97958	0.14285	1.15669	Iran	0.53619
17	0.95744	1.00353	1.39368	Banglade	0.73436
18	0.89838	1.24022	1.63285	Cambodia	1.07380
19	0.76551	0.85291	1.58936	Nepal	0.88866
20	1.42249	0.16437	0.26306	Botswana	0.75211
21	1.24533	0.80988	0.39352	Congo	0.53986
22	0.75074	1.28325	1.04580	Gabon	0.86684
23	1.11984	0.48713	0.76314	Ghana	0.12230
24	1.31177	0.09982	0.37178	Kenya	0.67632
25	1.09031	0.27196	1.74156	Namibia	0.92807
26	1.42249	1.02505	1.08928	Nigeria	0.58894
27	1.13460	1.06808	1.15451	Sudan	0.60039
28	1.29700	0.35802	1.37194	Swazilan	0.56207
29	1.69562	1.02505	1.04580	Uganda	0.73647
30	1.57013	0.68078	1.11103	Tanzania	0.49353
31	1.20842	0.72381	0.61095	Zaire	0.30705
32	1.61442	0.61623	0.54572	Zambia	0.54965

Cluster 3 and 4

---------- Cluster=3 ----------

Obs	birth	death	infant	country	DISTANCE
33	1.28224	1.54146	1.21974	Bolivia	0.97448
34	0.82456	1.69208	2.75477	Afghanis	1.06314
35	1.32653	2.01483	1.78505	Angola	0.23936
36	1.42988	2.12242	1.78505	Ethiopia	0.21566
37	1.34129	2.27303	1.91550	Gambia	0.09648
38	1.40773	3.04765	1.63285	Malawi	0.91953
39	1.16413	1.64904	1.87202	Mozambiq	0.56353
40	1.40035	2.70337	2.15467	Sierra_L	0.56234
41	1.54060	2.01483	1.67633	Somalia	0.39955

---------- Cluster=4 ----------

Obs	birth	death	infant	country	DISTANCE
42	-0.01697	2.66034	-0.25876	Mexico	0.61689
43	-0.42297	1.56297	-0.65013	Korea	0.61689

Cluster 5

Obs	birth	death	infant	country	DISTANCE
44	-1.23498	0.22892	-0.88060	Bulgaria	0.33826
45	-1.16854	0.18589	-0.94800	Czechosl	0.28590
46	-1.27189	0.33651	-1.02845	Former_E	0.45403
47	-1.30142	0.55168	-0.87190	Hungary	0.66583
48	-1.10211	-0.13687	-0.84581	Poland	0.12899
49	-1.15378	-0.02928	-0.60882	Romania	0.34023
50	-1.12425	-0.39507	-0.75449	Yugoslav	0.35379
51	-0.85112	-0.17990	-0.69361	USSR	0.42007
52	-1.03567	-0.28748	-0.90886	Byelorus	0.23981
53	-1.16854	0.16437	-0.91104	Ukrainia	0.26595
54	-0.82898	-0.26597	-0.71753	Uruguay	0.44895
55	-1.27189	-0.05080	-1.02193	Belgium	0.13118
56	-1.18331	-0.15838	-1.06759	Finland	0.13997
57	-1.24236	0.22892	-1.03062	Denmark	0.34609
58	-1.15378	-0.30900	-1.03280	France	0.23045
59	-1.31618	0.07830	-1.03280	Germany	0.24157
60	-1.41214	-0.35204	-0.95452	Greece	0.34233
61	-1.04305	-0.37355	-1.03062	Ireland	0.31979
62	-1.44167	-0.37355	-1.00236	Italy	0.38293
63	-1.18331	-0.48114	-1.03932	Netherla	0.39409
64	-1.10211	-0.02928	-1.02410	Norway	0.13492
65	-1.27927	-0.28748	-0.90886	Portugal	0.21422
66	-1.36785	-0.56721	-1.01758	Spain	0.50927
67	-1.08734	0.05679	-1.07193	Sweden	0.22485
68	-1.23498	-0.28748	-1.03932	Switzerl	0.21892
69	-1.15378	0.14285	-1.01105	U.K.	0.25402
70	-1.05781	-0.73934	-1.01975	Austria	0.65632
71	-1.42691	-0.88996	-1.09585	Japan	0.84206
72	-1.08734	-0.76086	-1.03715	Canada	0.67483
73	-0.92494	-0.58872	-0.99584	U.S.A.	0.55498

Cluster 6

Obs	birth	death	infant	country	DISTANCE
74	-0.04650	-0.63176	0.17609	Brazil	0.47528
75	0.27092	-0.73934	0.17609	Ecuador	0.38853
76	-0.06864	-0.76086	0.02389	Guyana	0.63871
77	0.41118	-0.91148	-0.28050	Paraguay	0.86208
78	0.27092	-0.54569	1.19582	Peru	0.75143
79	0.98696	-0.65327	0.30655	Iraq	0.72179
80	0.71383	-0.95451	-0.23702	Jordan	0.93690
81	0.18234	-0.45962	-0.15005	Lebanon	0.61194
82	1.20842	-0.65327	-0.32399	Oman	1.20413
83	0.95005	-0.69631	0.35003	Saudi_Ar	0.69244
84	-0.00221	-0.52417	0.45875	Turkey	0.31280
85	0.09376	-0.13687	0.78489	India	0.51477
86	-0.04650	-0.30900	0.43700	Indonesi	0.38480
87	0.50714	-0.43810	0.28481	Mongolia	0.26224
88	0.07899	-0.58872	1.14799	Pakistan	0.74443
89	0.29307	-0.67479	-0.21527	Philippi	0.69706
90	0.18972	-0.28748	0.19784	Vietnam	0.32897
91	0.46285	-0.54569	0.41526	Algeria	0.18049
92	0.70645	-0.28748	-0.11961	Egypt	0.71885
93	1.09031	-0.30900	0.58920	Libya	0.81294
94	0.46285	-0.22293	0.58920	Morocco	0.32133
95	0.21187	-0.20142	0.37178	South_Af	0.29044
96	0.13805	-0.76086	-0.06308	Tunisia	0.61470
97	0.92053	-0.11535	0.24132	Zimbabwe	0.73814

Summary of clusters

- Cluster 3 has highest means on all variables; describe as "very poor" countries.
- Cluster 2 also higher than average on all, but not as high as Cluster 3:"poor" but not "very poor".
- Cluster 4 has high death rate but low birth rates and infant mortality rates: "would-be western".
- Cluster 6 has slightly above-average birth and infant mortality rates, and lower-than-average death rate: "third world".
- Cluster 1 has lower-than-average everything, and especially low death rate: "becoming western".
- Cluster 5 also is low on everything, and especially low on birth rate: "western world".
- New variable "distance" shows how far a country is from its cluster average. Small value means "typical of its cluster"; large implies "does not fit any cluster very well". Eg. Afghanistan vs. cluster 3.

Using PROC DISCRIM on clusters

- Summary on previous page took some working out.
- Idea: use output clusters as "grouping" variable for PROC DISCRIM with "can" option: get canonical variables that might shed some light on how clusters differ.
- Code below. Add onto end of previous (uses output data set with cluster membership in it):

```
proc discrim can out=zz;
  class cluster;
  var birth death infant;

proc sort;
  by cluster;

proc print;
  var country birth death infant can1 can2 can3;
  by cluster;
```


Output from discriminant analysis

The DISCRIM Procedure

Observations	97	DF Total	96
Variables	3	DF Within Classes	91
Classes	6	DF Between Classes	5

Eigenvalues of Inv(E)*H = CanRsq/(1-CanRsq)

	Eigenvalue	Difference	Proportion	Cumulative
1	25.2031	20.7304	0.8449	0.8449
2	4.4727	4.3181	0.1499	0.9948
3	0.1546		0.0052	1.0000

Test of H0: The canonical correlations in the current row and all that follow are zero

	Likelihood Ratio	Approximate F Value	Num DF	Den DF	Pr > F
1	0.00603979	88.02	15	246.09	<.0001
2	0.15826107	34.06	8	180	<.0001
3	0.86611339	4.69	3	91	0.0043

The canonical variables

3 canonical variables possible, all significant (though eigenvalue for last small).

Raw Canonical Coefficients

Variable	Can1	Can2	Can3
birth	2.706578482	-1.095888137	-1.935532844
death	0.683696755	2.777557832	-0.585485143
infant	2.017039026	-0.834166707	2.334460951

- Can1 positive where birth rate and infant mortality rate both high, negative where both low.
- Can2 positive where death rate high, negative where low.
- Can3 positive where infant mortality rate high compared to birth rate, negative where low.

The clusters

Class Means on Canonical Variables

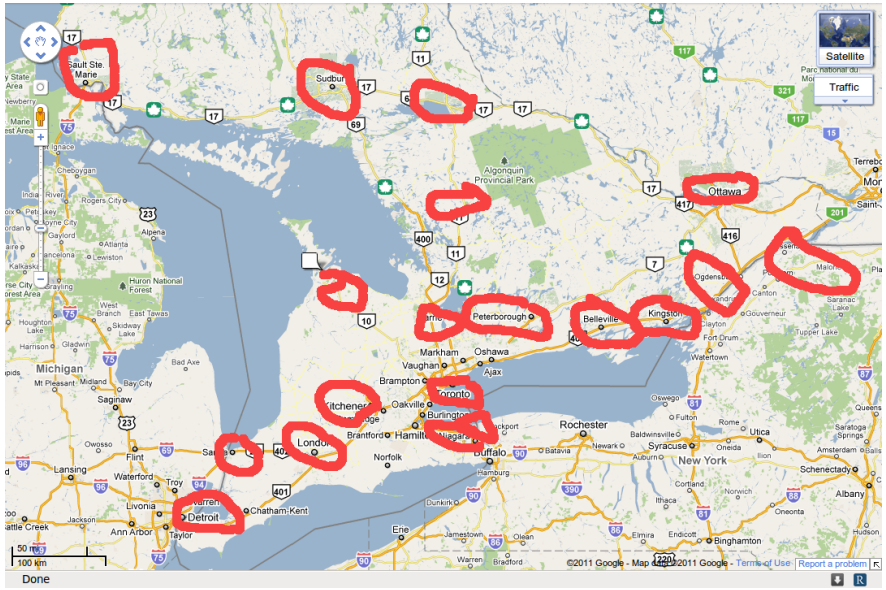
CLUSTER	Can1	Can2	Can3
1	-3.430124271	-2.092217034	-0.186588012
2	5.788318527	-0.231819353	-0.367431160
3	8.735819354	2.898350402	0.596858239
4	-0.068268900	6.485397951	-1.871461306
5	-5.226778630	1.565961865	0.154681580
6	1.306998818	-2.112942540	0.115662540

1. Low on everything (Chile).
2. High birth and infant mortality, average death rate (Ghana).
3. High (or very high) on everything (Gambia).
4. High death rate, high birth rate compared to infant mortality rate (Mexico).
5. Very low birth and infant mortality, highish death rate (Canada).
6. High birth and infant mortality but low death rate (Algeria).

Final example: a hockey league

- An Ontario hockey league has teams in 21 cities. How can we arrange those teams into 4 geographical divisions?
- Distance data in spreadsheet.
- Take out spaces in team names.
- Save as "text/csv", and use text editor to remove all double-quotes.
- Open new file on Mathlab.
- Copy lines with team names and distances to clipboard, paste into Mathlab file.
- PROC FASTCLUS doesn't work on distance data, so go back to PROC CLUSTER.

A map



My code

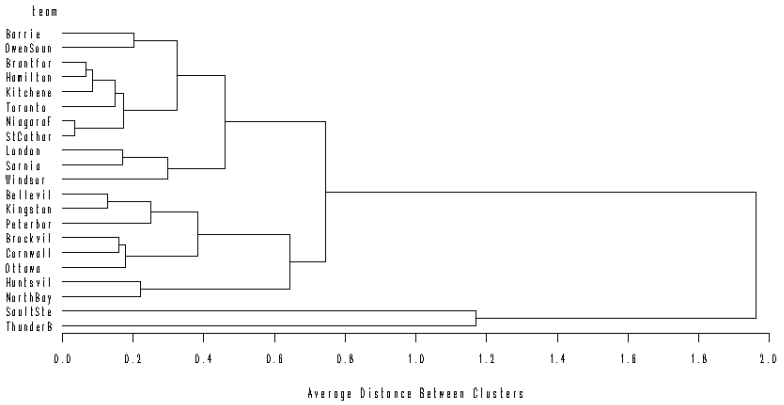
```
options linesize=75;

data dist(type=distance);
  infile "ontario-road-distances.dat" delimiter=",";
  input team $ Barrie Belleville Brantford Brockville
    Cornwall Hamilton Huntsville Kingston Kitchener
    London NiagaraFalls NorthBay Ottawa OwenSound
    Peterborough Sarnia SaultSteMarie StCatharines
    ThunderBay Toronto Windsor;

proc cluster method=ward outtree=tree;
  id team;

proc tree horizontal;
  id team;
```

Use same team names in same order as data file. Hope tree output gives some idea of which teams to place in which divisions.

Clustering history

	Cluster History			Norm RMS	T i
NCL	--Clusters Joined---		FREQ	Dist	e
20	NiagaraF	StCathar	2	0.0339	
19	Brantfor	Hamilton	2	0.0678	T
18	CL19	Kitchene	3	0.0864	
17	Bellevil	Kingston	2	0.1271	
16	CL18	Toronto	4	0.1489	
15	Brockvil	Cornwall	2	0.161	
14	London	Sarnia	2	0.1695	
13	CL16	CL20	6	0.1742	
12	CL15	Ottawa	3	0.1782	
11	Barrie	OwenSoun	2	0.2034	
10	Huntsvil	NorthBay	2	0.2203	
9	CL17	Peterbor	3	0.2497	
8	CL14	Windsor	3	0.2977	
7	CL11	CL13	8	0.3246	
6	CL9	CL12	6	0.3842	
5	CL7	CL8	11	0.4606	
4	CL6	CL10	8	0.6431	
3	CL5	CL4	19	0.7445	
2	SaultSte	ThunderB	2	1.1694	
1	CL3	CL2	21	1.9625	

The tree



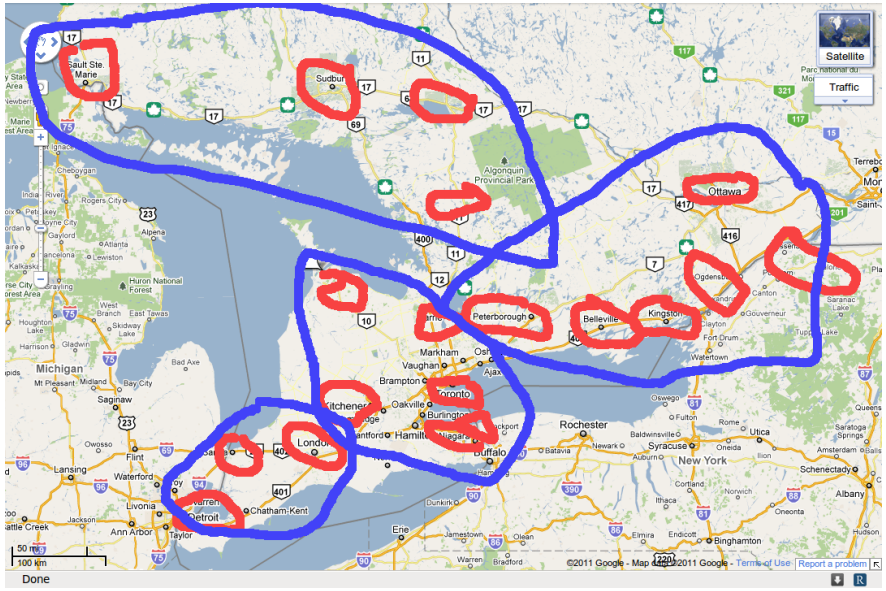
Splitting into divisions, 1st try

- Sault Ste Marie and Thunder Bay are very distant from everywhere else.
- Clustering history says 4 clusters between distance 0.6431 and 0.7445, so "chop tree" there, to get:
  - Sault Ste Marie
  - Thunder Bay
  - Belleville, Kingston, Peterborough, Brockville, Cornwall, Ottawa, Huntsville, North Bay (8 teams)
  - the rest (11 teams)
- Divisions of 1 team make no sense, so try splitting big divisions and placing 2 northernmost teams somewhere.

2nd try

- Next split at distance 0.6431 splits Huntsville and North Bay from the eastern teams. Place them in a northern division with Sault Ste Marie and Thunder Bay.
- Next split at distance 0.4606 splits London, Sarnia and Windsor off from the big group. That leaves us with this:
  - (north, 4) Huntsville, North Bay, Sault Ste Marie, Thunder Bay
  - (east, 6) Belleville, Kingston, Peterborough, Brockville, Cornwall, Ottawa
  - (west, 3) London, Sarnia, Windsor
  - (south, 8) Niagara Falls, St Catharines, Brantford, Hamilton, Kitchener, Toronto, Barrie, Owen Sound
- That's not too bad. Getting the divisions to be the same size is beyond our scope!

Another map



Where we are going

Multidimensional Scaling

- Have distances between individuals.
- Want to draw a picture (map) in 2 dimensions showing individuals so that distances (or order of distances) as close together as possible.
- If want to preserve actual distances, called *metric multidimensional scaling* (in SAS, `level=absolute`)
- If only want to preserve order of distances, called *non-metric multidimensional scaling* (in SAS, `level=ordinal`).
- Metric scaling has solution that can be worked out exactly.
- Non-metric only has iterative solution.
- Assess quality of fit via quantity "stress", whether use of resulting map is reasonable. (Try something obviously 3-dimensional and assess its failure.) Stress has min 0 and max 1.

Metric scaling: European cities

The file `europe.dat` contains road distances (in km) between 16 European cities. Can we reproduce a map of Europe from these distances?

First, reading in the data (as TYPE=DISTANCE):

```
data euro(type=distance);
  infile "europe.dat" delimiter=",";
  input city $ Amsterdam Athens Barcelona Berlin
    Cologne Copenhagen Edinburgh Geneva London
    Madrid Marseille Munich Paris Prague Rome Vienna;
```

- Values in spreadsheet.
- Save as .csv.
- Take out quotes.
- Values separated by commas, suitable for reading by SAS.

The code, using PROC MDS

```
proc mds level=absolute out=y outres=z;
proc print data=y;
proc sort data=z;
  by residual;
proc print data=z;
  var _row_ _col_ residual;
symbol1 pointlabel=('#_label_');
proc gplot data=y;
  plot dim1 * dim2;
```

- Run PROC MDS using `level=absolute` to reproduce the exact distances (to scale).
- Two output data sets: one containing the coordinates for our map, and one containing the observed and predicted (from map) distances and residuals.
- Print coordinates.
- Sort residuals and print them (with the cities they belong to).
- Plot coordinates, labelling each point by its city.

The coordinates

In Dim1 and Dim2:

			TYPE	_LABEL_	_NAME_	Dim1	Dim2
1	2	.	CRITERION			0.07	.
2	2	.	CONFIG	Amsterdam	Amsterdam	-300.71	558.62
3	2	.	CONFIG	Athens	Athens	2599.74	-375.74
4	2	.	CONFIG	Barcelona	Barcelona	-704.34	-1012.29
5	2	.	CONFIG	Berlin	Berlin	402.29	619.72
6	2	.	CONFIG	Cologne	Cologne	-83.70	396.98
7	2	.	CONFIG	Copenhagen	Copenhagen	97.17	1241.96
8	2	.	CONFIG	Edinburgh	Edinburgh	-1232.60	906.77
9	2	.	CONFIG	Geneva	Geneva	-185.99	-342.22
10	2	.	CONFIG	London	London	-574.43	406.08
11	2	.	CONFIG	Madrid	Madrid	-1341.37	-1088.16
12	2	.	CONFIG	Marseille	Marseille	-319.76	-750.10
13	2	.	CONFIG	Munich	Munich	326.13	-25.17
14	2	.	CONFIG	Paris	Paris	-525.60	49.92
15	2	.	CONFIG	Prague	Prague	541.20	285.90
16	2	.	CONFIG	Rome	Rome	541.38	-1031.08
17	2	.	CONFIG	Vienna	Vienna	760.58	158.80

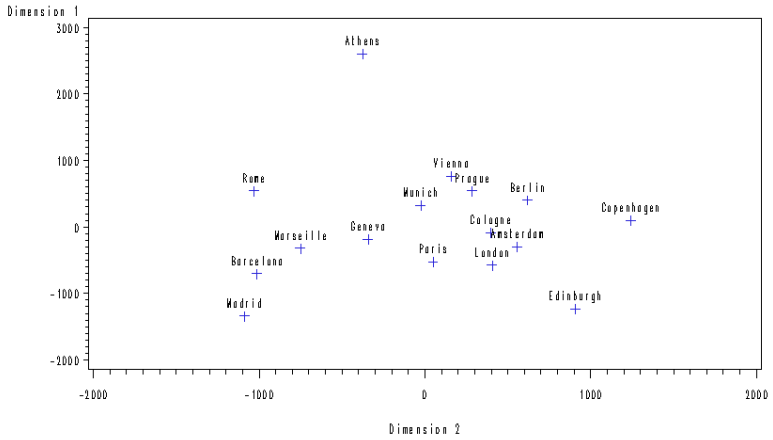
Stress 0.07 is small.

The sorted residuals (edited)

	Obs	_ROW_	_COL_	RESIDUAL
	1	Vienna	London	-445.723
	2	Edinburgh	Athens	-273.247
	3	Cologne	Athens	-230.477
	4	London	Edinburgh	-170.966
	5	Madrid	Cologne	-170.119
	6	London	Athens	-170.038
...				
	115	Rome	Madrid	215.393
	116	Rome	Barcelona	225.139
	117	Madrid	Edinburgh	374.108
	118	Rome	Athens	390.827
	119	Copenhagen	Athens	434.100
	120	Edinburgh	Copenhagen	492.631

Edinburgh and Athens feature in a lot of the large residuals.

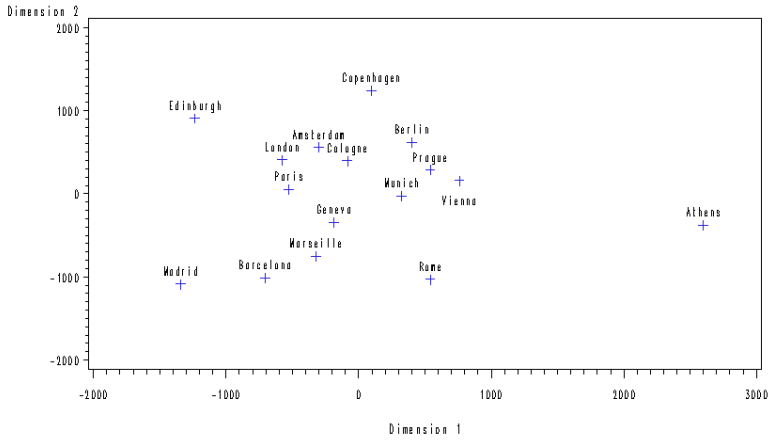
The map



Comments on map

- The map looks upside down!
- MDS doesn't know about directions, only distances, so map could come out reflected (vertically or horizontally) or rotated.
- Given all that, cities look in about right relative places.
- City pairs with largest positive residuals have large bodies of water between them (affecting road distance considerably):
  - Edinburgh–Copenhagen (North Sea)
  - Rome–Athens (Adriatic)
- As it happens, plotting `Dim2*Dim1` produces almost reasonable map:

Map 2



Non-metric scaling: languages

- Recall language data (from cluster analysis): 1–10, measure dissimilarity between two languages by how many number names *differ* in first letter. Data:

```
en 0 2 2  7 6  6   6  6  7  9 9
no 2 0 1  5 4  6   6  6  7  8 9
dk 2 1 0  6 5  6   5  5  6  8 9
nl 7 5 6  0 5  9   9  9 10  8 9
de 6 4 5  5 0  7   7  7  8  9 9
fr 6 6 6  9 7  0   2  1  5 10 9
es 6 6 5  9 7  2   0  1  3 10 9
it 6 6 5  9 7  1   1  0  4 10 8
pl 7 7 6 10 8  5   3  4  0 10 9
hu 9 8 8  8 9 10 10 10 10  0 8
sf 9 9 9  9 9  9   9  8  9  8 0
```

- Only want to reproduce *order* of dissimilarities; actual numbers don't matter. (Map only reproduces *relative* closeness of languages.)

Code

- Read data as distances, use `level=ordinal`. Print coordinates and residuals, plot map (labelled by language):

```
data lang(type=distance);
  infile "one-ten.dat";
  input lang $ en no dk nl de fr es it pl hu sf;

proc mds level=ordinal out=coords outres=dist;
  id lang;

proc print data=dist;
  var _row_ _col_ data distance residual;

proc print data=coords;

symbol1 pointlabel=('#lang');
proc gplot data=coords;
  plot dim2 * dim1;
```


Output from PROC MDS

```
Multidimensional Scaling:  Data=WORK.LANG.DATA
 Shape=TRIANGLE Condition=MATRIX Level=ORDINAL
   Coef=IDENTITY Dimension=2 Formula=1 Fit=1
Mconverge=0.01 Gconverge=0.01 Maxiter=100 Over=2 Ridge=0.0001
```

Iteration	Type	Badness-of-Fit Criterion	Change in Criterion	Convergence Measures: Monotone	Convergence Measures: Gradient
0	Initial	0.2009	.	.	.
1	Monotone	0.1478	0.0531	0.1358	0.6781
2	Gau-New	0.1126	0.0352	.	.
3	Monotone	0.1020	0.0105	0.0483	0.3363
4	Gau-New	0.0997	0.002376	.	.
5	Monotone	0.0928	0.006869	0.0374	0.2226
6	Gau-New	0.0923	0.000483	.	.
7	Monotone	0.0915	0.000823	0.0138	0.2190
8	Gau-New	0.0914	0.0000983	.	.
9	Monotone	0.0910	0.000349	0.009497	0.2341
10	Gau-New	0.0888	0.002191	.	0.0533
11	Gau-New	0.0887	0.000106	.	0.0169
12	Gau-New	0.0887	0.0000126	.	0.006850

Iterative procedure converges (stress stops getting smaller at 0.0887, which is small).

The residuals (selected)

Shown: pair of languages, dissimilarity, distance on map, residual (based on ordered data). Large residual means data and distance on map don't match.

Obs	_ROW_	_COL_	DATA	DISTANCE	RESIDUAL
7	de	en	6	0.81928	0.49528
55	sf	hu	8	2.00452	0.35904
49	sf	nl	9	3.15422	-0.34249
40	hu	nl	8	2.02361	0.33995
48	sf	dk	9	2.48611	0.32562
31	pl	dk	6	1.62422	-0.30966
6	nl	dk	6	1.61869	-0.30413
50	sf	de	9	3.10815	-0.29643
5	nl	no	5	1.31502	-0.27280
32	pl	nl	10	3.23354	0.24178
54	sf	pl	9	2.54350	0.26823

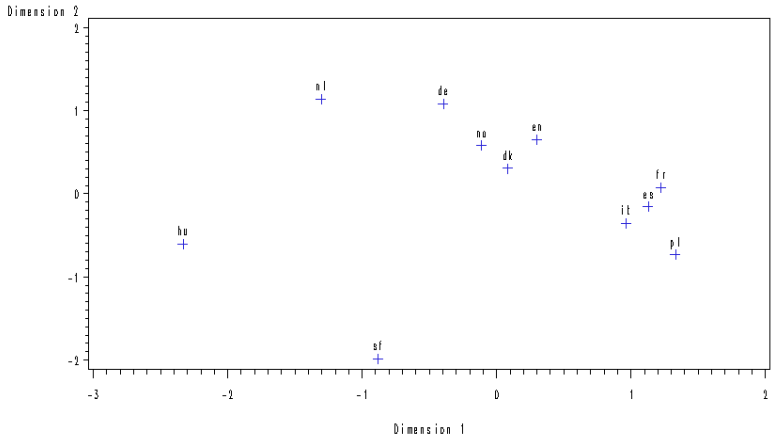
- Positive residual: actual dissimilarity greater than expected (compared to map)
- Negative residual: actual dissimilarity less than expected from map.

The coordinates

Obs	_DIMENS_	_MATRIX_	_TYPE_	lang	_NAME_	Dim1	Dim2
1	2	.	CRITERION			0.08872	.
2	2	.	CONFIG	en	en	0.30099	0.65225
3	2	.	CONFIG	no	no	-0.11417	0.58068
4	2	.	CONFIG	dk	dk	0.08220	0.30450
5	2	.	CONFIG	nl	nl	-1.30472	1.13912
6	2	.	CONFIG	de	de	-0.39587	1.08307
7	2	.	CONFIG	fr	fr	1.22529	0.07596
8	2	.	CONFIG	es	es	1.12900	-0.15541
9	2	.	CONFIG	it	it	0.96244	-0.35587
10	2	.	CONFIG	pl	pl	1.33098	-0.73409
11	2	.	CONFIG	hu	hu	-2.33345	-0.60349
12	2	.	CONFIG	sf	sf	-0.88268	-1.98673

- 1st row: stress value (max 1, min 0).
- CONFIG lines: Dim1 and Dim2 have coordinates.

The map



Comments on map

- See how distant Hungarian and Finnish are from each other, and the rest.
- See tight grouping of Italian, French and Spanish (Polish nearby).
- See looser grouping of Germanic languages at top (English, German, Dutch, Norwegian, Danish).

Guidelines for stress values

Smaller is better:

Stress value	Interpretation
Less than 0.05	Excellent: no prospect of misinterpretation (rarely achieved)
0.05–0.10	Good: most distances reproduced well, small prospect of false inferences
0.10–0.20	Fair: usable, but some distances misleading.
More than 0.20	Poor: may be dangerous to interpret

- Cities and languages examples both had stress in "good" range.

A cube

```
a-----b
|\    |\
| c---- d
| |   | |
e-|---f |
 \|    \|
  g-----h
```

Cube has side length 1, so distance across diagonal on same face is $\sqrt{2} \simeq 1.4$ and "long" diagonal of cube is $\sqrt{3} \simeq 1.7$.

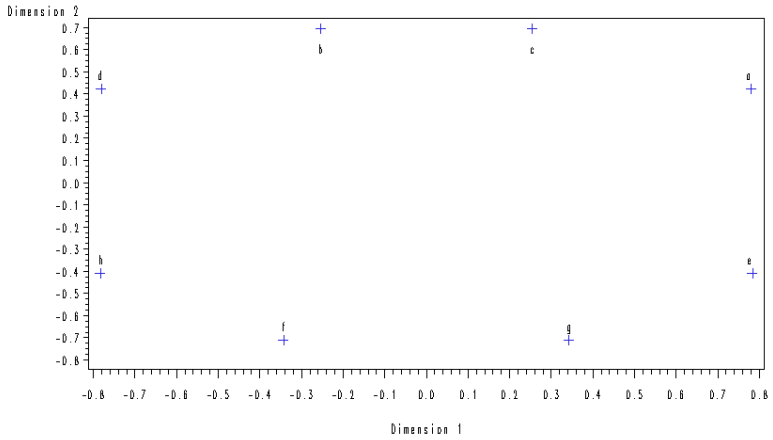
Try MDS on this obviously 3-dimensional data.

Converges OK

Iteration	Type	Badness-of-Fit Criterion	Change in Criterion	Convergence Measure
0	Initial	0.2987	.	0.6106
1	Lev-Mar	0.2275	0.0711	0.1308
2	Gau-New	0.2251	0.002446	0.0409
3	Gau-New	0.2248	0.000263	0.0164
4	Gau-New	0.2248	0.0000426	0.006667

but stress, at 0.2248, in "poor" range. Map probably won't reproduce cube very well.

"Map" of cube



Comments

- Map doesn't resemble cube.
- Some of the residuals are large: eg. g and f: actual distance is 1.4, map distance 0.7.
- Might have guessed this with stress in "poor" range.
- SAS lets you choose dimension of map. Use this PROC MDS line:

```
proc mds dim=3 level=absolute outres=res2;
```

(no point saving coordinates since we cannot plot them.)

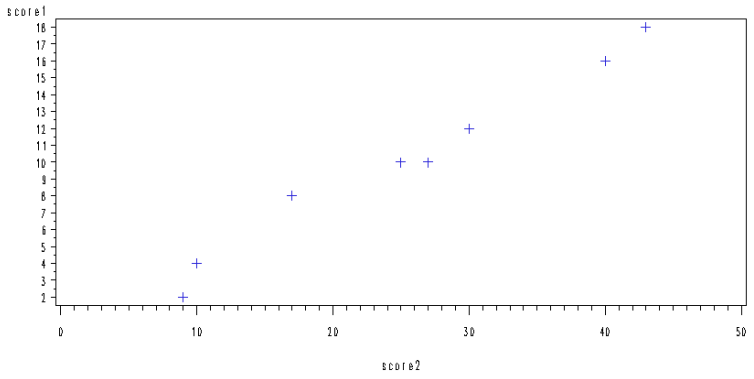
- Resulting stress is 0.0342, "excellent".
- Largest residual (in size) is -0.1, most much smaller.
- Can't "squeeze" 3-D data into 2 dimensions!

Where we are going

Principal Components

- Have measurements on (possibly large) number of variables on some individuals.
- Question: can we describe data using fewer variables (because original variables correlated in some way)?
- Look for direction (linear combination of original variables) in which values *most spread out*. This is *first principal component*.
- Second principal component then direction uncorrelated with this in which values then most spread out. And so on.
- See whether small number of principal components captures most of variation in data.
- Might try to interpret principal components.
- If 2 components good, can make plot of data.
- (Akin to ideas in discriminant/canonical variables analysis, but no groups here.)

Small example: 2 test scores for 8 people



Principal component analysis

Strongly correlated, so data nearly 1-dimensional. Make a score summarizing this one dimension.
Code like this:

```
options linesize=70;

data test;
  infile "test12.dat";
  input score1 score2;

proc princomp out=fred;

proc print;
```


The output

Correlation Matrix

	score1	score2
score1	1.0000	0.9891
score2	0.9891	1.0000

Eigenvalues of the Correlation Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	1.98907796	1.97815591	0.9945	0.9945
2	0.01092204		0.0055	1.0000

- The two variables are very highly correlated.
- Look at *eigenvalues*:
  - First one is much bigger than rest, so data "almost" 1-dimensional.
  - Last column: first principal component accounts for almost all (99.45%) of variability in data, so we do fine by summarizing data by 1st principal component.
  - Generally: consider retaining components with eigenvalues bigger than 1.

Eigenvectors

Eigenvectors

	Prin1	Prin2
score1	0.707107	0.707107
score2	0.707107	-.707107

- Eigenvectors show how principal components depend on original variables (standardized).
- 1st principal component is basically sum of `score1` and `score2`, standardized.
- If correlation between 2 test scores had been negative, 1st eigenvector would have said "look at difference".

Output data set

Obs	score1	score2	Prin1	Prin2
1	2	9	-1.93801	-0.13749
2	16	40	1.60878	-0.05216
3	8	17	-0.71306	0.19418
4	18	43	2.03571	0.03979
5	10	25	-0.00698	0.00698
6	4	10	-1.62274	0.06612
7	10	27	0.10468	-0.10468
8	12	30	0.53161	-0.01273

- Values on `Prin1` identify each person as a "high scorer" (positive) or "low scorer" (negative).
- `Prin1` and `Prin2` called *principal component scores*.

Track running data

(1984) track running records for distances 100m to marathon, arranged by country. Countries labelled by (mostly) Internet domain names — actual names not in file.

```
10.39  20.81  46.84  1.81  3.70  14.04  29.36  137.72  ar (Argentina)
10.31  20.06  44.84  1.74  3.57  13.28  27.66  128.30  au (Australia)
10.44  20.81  46.82  1.79  3.60  13.26  27.72  135.90  at (Austria)
10.34  20.68  45.04  1.73  3.60  13.22  27.45  129.95  be (Belgium)
10.28  20.58  45.91  1.80  3.75  14.68  30.55  146.62  bm (Bermuda)
10.22  20.43  45.21  1.73  3.66  13.62  28.62  133.13  br (Brazil)
10.64  21.52  48.30  1.80  3.85  14.45  30.28  139.95  bu (Burma)
10.17  20.22  45.68  1.76  3.63  13.55  28.09  130.15  ca (Canada)
10.34  20.80  46.20  1.79  3.71  13.61  29.30  134.03  cl (Chile)
....
10.71  21.43  47.60  1.79  3.67  13.56  28.58  131.50  tr (Turkey)
 9.93  19.75  43.86  1.73  3.53  13.20  27.43  128.22  us (United States)
10.07  20.00  44.60  1.75  3.59  13.20  27.53  130.55  ru (USSR)
10.82  21.86  49.00  2.02  4.24  16.28  34.71  161.83  ws (Western Samoa)
```


Data and aims

- Times in seconds 100m-400m, in minutes for rest (800m, 1500m, 5000m, 10000m, marathon).
- This taken care of by (automatic) standardization.
- 8 variables; can we summarize by fewer and gain some insight?
- In particular, if 2 components tell most of story, what do we see in a plot?

Code

```
options linesize=70;

data track;
  infile "men_track_field.dat";
  input m100 m200 m400 m800 m1500 m5000 m10000 marathon
    country $;

proc princomp out=PC;

proc print data=PC;
  var country Prin1 Prin2;

symbol1 pointlabel=('#country');

proc gplot;
    plot Prin2*Prin1;
```


Correlation matrix

Correlation Matrix

	m100	m200	m400	m800	m1500	m5000	m10000	marathon
m100	1.0000	0.9226	0.8411	0.7560	0.7002	0.6195	0.6325	0.5199
m200	0.9226	1.0000	0.8507	0.8066	0.7750	0.6954	0.6965	0.5962
m400	0.8411	0.8507	1.0000	0.8702	0.8353	0.7786	0.7872	0.7050
m800	0.7560	0.8066	0.8702	1.0000	0.9180	0.8636	0.8690	0.8065
m1500	0.7002	0.7750	0.8353	0.9180	1.0000	0.9281	0.9347	0.8655
m5000	0.6195	0.6954	0.7786	0.8636	0.9281	1.0000	0.9746	0.9322
m10000	0.6325	0.6965	0.7872	0.8690	0.9347	0.9746	1.0000	0.9432
marathon	0.5199	0.5962	0.7050	0.8065	0.8655	0.9322	0.9432	1.0000

All variables positively correlated, but less so as gap between running distances increases.

The eigenvalues

Eigenvalues of the Correlation Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	6.62214613	5.74452784	0.8278	0.8278
2	0.87761829	0.71829715	0.1097	0.9375
3	0.15932114	0.03527176	0.0199	0.9574
4	0.12404939	0.04416911	0.0155	0.9729
5	0.07988027	0.01191512	0.0100	0.9829
6	0.06796515	0.02154562	0.0085	0.9914
7	0.04641953	0.02381943	0.0058	0.9972
8	0.02260010		0.0028	1.0000

Only 1st is bigger than 1, but 2nd is much bigger than others, so include that as well.

The eigenvectors

Eigenvectors

	Prin1	Prin2	Prin3	Prin4
m100	0.317556	0.566878	0.332262	0.127628
m200	0.336979	0.461626	0.360657	-.259116
m400	0.355645	0.248273	-.560467	0.652341
m800	0.368684	0.012430	-.532482	-.479999
m1500	0.372810	-.139797	-.153443	-.404510
m5000	0.364374	-.312030	0.189764	0.029588
m10000	0.366773	-.306860	0.181752	0.080069
marathon	0.341926	-.438963	0.263209	0.299512

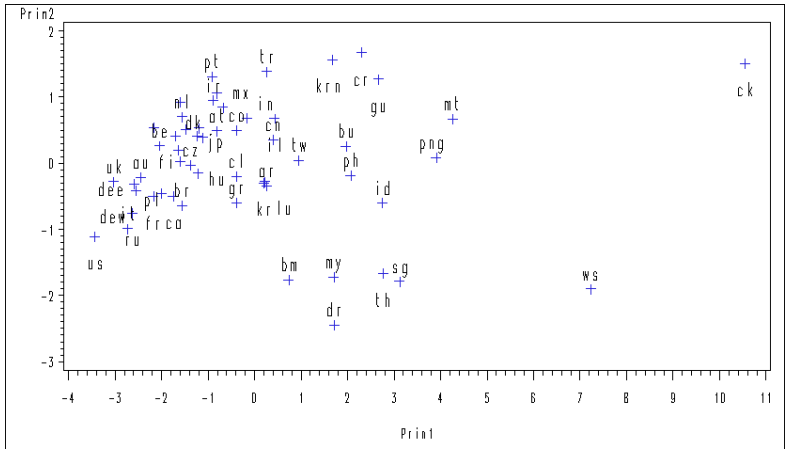
- `Prin1` basically average of record times: "how good a country is at running".
- `Prin2` contrasts sprinting with distance running.
- `Prin3` contrasts longer sprints with everything else.
- More, but not going to keep `Prin3` and beyond.

Principal component scores (selected)

`Prin1` measures "good overall" (negative best), `Prin2` measures "sprinting vs. distance" (negative: sprinting, positive: distance). (SAS has turned this around: check back with original data.)

Obs	country	Prin1	Prin2
1	ar	0.2619	-0.34488
2	au	-2.4464	-0.21617
4	be	-2.0413	0.26195
8	ca	-1.7464	-0.50035
12	ck	10.5556	1.50877
18	fr	-2.1719	-0.50289
19	dee	-2.5901	-0.31067
20	dew	-2.5527	-0.41137
29	it	-2.7269	-0.98986
30	jp	-1.2379	0.41357
31	ke	-2.1683	0.53371
37	mx	-0.6785	0.84175
43	pl	-2.0006	-0.46260
44	pt	-0.9164	1.30473
45	rm	-1.1965	0.53077
53	us	-3.4306	-1.11019
54	ru	-2.6269	-0.75696
55	ws	7.2312	-1.90208

Component scores plot



Data TYPE=CORR

- Just as data TYPE=DISTANCE (we used for clustering), also TYPE=CORR, used for matrices of correlations.
- Procedure PROC CORR will produce this as output.
- Example small data set (three variables):

```
3  7 20
4 10 16
6 15 11
9 18  8
```

x_2 is just over twice x_1, while x_3 goes down as x_1 and x_2 goes up. So expect some high positive and negative correlations.

Using PROC CORR

- Code like this:

```
data xc;
  infile "xcorr.dat";
  input x1 x2 x3;

proc corr out=fred;

proc print;
```

- PROC CORR itself produces some output (ignored) and output data set looks like this:

Obs	_TYPE_	_NAME_	x1	x2	x3
1	MEAN		5.50000	12.5000	13.7500
2	STD		2.64575	4.9329	5.3151
3	N		4.00000	4.0000	4.0000
4	CORR	x1	1.00000	0.9705	-0.9600
5	CORR	x2	0.97054	1.0000	-0.9980
6	CORR	x3	-0.96001	-0.9980	1.0000

- Last 3 lines are matrix of correlations; values as expected.

TYPE=CORR data set

- Full data set includes mean and SD of each variable, and number of observations for each (same for each variable). Entry in new variable _TYPE_ says what each row of numbers is.
- Each correlation is of *two* variables, so entry in row of _NAME_ and variable column says which two variables involved.
- Can create TYPE=CORR data set yourself. Not everything has to be specified:
  - if _TYPE_ missing, CORR assumed.
  - if _NAME_ missing, variables may not get names (but OK if planning to use all in analysis).
  - Correlation eg. between x_1 and x_2 same as between x_2 and x_1, so can give redundant correlations as . (missing).
  - PROC PRINCOMP only uses correlations (not mean, SD or sample size — no testing). So can still do if you have only correlations.

Doing principal components with (above) correlations only

- Create data file like this:

```
1 0.9705 -0.9600
. 1      -0.9980
. .       1
```

- Code like below. Note: no actual data implies no component scores (and no output data set).

```
data yc(type=corr);
  infile "ycorr.dat";
  input x1 x2 x3;

proc princomp;
```

- Can also use PROC PRINT to check proper reading of data.
- PROC PRINCOMP handles this kind of data automatically.

Output

The PRINCOMP Procedure

Eigenvalues of the Correlation Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	2.95242147	2.90600335	0.9841	0.9841
2	0.04641812	0.04525771	0.0155	0.9996
3	0.00116041		0.0004	1.0000

Eigenvectors

	Prin1	Prin2	Prin3
x1	0.573007	0.811856	-.112035
x2	0.580528	-.305586	0.754721
x3	-.578489	0.497500	0.646408

- Data behind correlations effectively one-dimensional.
- Principal component made of first two variables minus third almost entirely summarizes data.

Where we are going

Principal components and factor analysis

- Principal components:
  - Purely mathematical.
  - Find eigenvalues, eigenvectors of correlation matrix.
  - No testing whether observed components reproducible, or even probability model behind it.
- Factor analysis:
  - some way towards fixing this (confirmatory factor analysis, later, a long way).
  - In factor analysis, each variable modelled as: "common factor" (eg. verbal ability) and "specific factor" (left over).
  - SAS: choose the common factors to "best" reproduce pattern seen in correlation matrix.
  - Iterative procedure, different answer from principal components.

Example

- 145 children given 5 tests, called PARA, SENT, WORD, ADD and DOTS. 3 linguistic tasks (paragraph comprehension, sentence completion and word meaning), 2 mathematical ones (addition and counting dots).
- Correlation matrix:

```
para 1     0.722 0.714 0.203 0.095
sent 0.722 1     0.685 0.246 0.181
word 0.714 0.685 1     0.170 0.113
add  0.203 0.246 0.170 1     0.585
dots 0.095 0.181 0.113 0.585 1
```

- Is there small number of underlying "constructs" (unobservable) that explains this pattern of correlations?
- First item on each line is name of variable: use SAS special variable `_name_` to read these in.
- First task: figure out number of factors:
  - again can count eigenvalues > 1
  - draw *scree plot* and look for "elbow".

Code

```
data rmat(type=corr);
  infile "rex2.dat";
  input _name_ $ para sent word add dots;

proc factor scree method=prinit;
```

- Names on INPUT line same as names of variables in file.
- On PROC FACTOR line, specify method of extracting factors (there are others) and ask for scree plot.
- As in principal components, can ask for output data set containing factor scores, but:
  - only if have actual data rather than correlations
  - only goal for this run of PROC FACTOR is to determine a good number of factors.

Output

Start with eigenvalues:

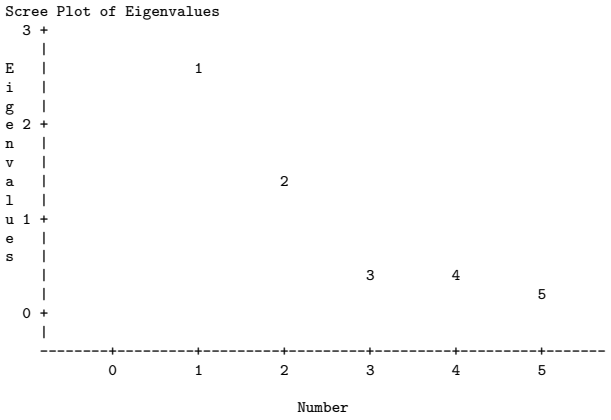
Preliminary Eigenvalues: Total = 5 Average = 1

	Eigenvalue	Difference	Proportion	Cumulative
1	2.58746987	1.16575215	0.5175	0.5175
2	1.42171772	1.00652661	0.2843	0.8018
3	0.41519110	0.10409071	0.0830	0.8849
4	0.31110040	0.04657948	0.0622	0.9471
5	0.26452092		0.0529	1.0000

2 factors will be retained by the MINEIGEN criterion.

2 eigenvalues bigger than 1, so SAS keeps 2 factors. 80% of variability explained by these, not bad.

Scree plot



Looking for where plot "turns corner" or "has elbow": at 3rd eigenvalue, so keep $3 - 1 = 2$ factors.

Eigenvalues of reduced correlation matrix

After SAS has finished iterating, the eigenvalues are different:

Eigenvalues of the Reduced Correlation Matrix:
Total = 3.31477718 Average = 0.66295544

	Eigenvalue	Difference	Proportion	Cumulative
1	2.28220070	1.25031114	0.6885	0.6885
2	1.03188956	1.00687378	0.3113	0.9998
3	0.02501578	0.02604204	0.0075	1.0073
4	-.00102626	0.02227632	-0.0003	1.0070
5	-.02330258		-0.0070	1.0000

Sometimes they are slightly negative, but this is nothing to worry about. SAS chose 2 factors, so other eigenvalues very close to 0.

Factor pattern and communality estimates

Factor Pattern

	Factor1	Factor2
para	0.83498	-0.24200
sent	0.82533	-0.13946
word	0.78992	-0.22671
add	0.40982	0.63174
dots	0.33454	0.70949

Factor 1 mostly "words" and factor 2 mostly "numbers", but could be clearer. Called "factor loadings", easier to interpret if close to 0 or ± 1.

Final Communality Estimates: Total = 3.314090

para	sent	word	add	dots
0.75574929	0.70062380	0.67537332	0.56705315	0.61529069

Show how each variable related to the factors (jointly): a low communality means the variable concerned not related to any of the factors. Here, though, all communalities reasonably high. (Actually R-squareds from regression of variable on factor.)

What to do next

- 2 factors appears to be good. No longer worry about scree plot or getting SAS to choose: we specify.
- *Factor rotation:*
  - So far, choose 1st factor to maximize spread, and 2nd factor ditto, while unrelated to 1st factor.
  - Now know we'll have 2 factors, so choose them to jointly maximize spread.
  - Introduces extra "degree of freedom", can use to get "interpretable" factors by idea of *factor rotation*.
  - *Varimax* rotation tries to drive *columns* of factor pattern close to 0 or ± 1.
  - *Quartimax* rotation tries to arrange that each variable only appears in *one* factor.

More code

Replace previous PROC FACTOR call with following:

```
proc factor n=2 method=prinit rotate=varimax;
```

We decide on 2 factors, ask for varimax rotation.
Produces output from before plus following:

Rotated factors

Rotated Factor Pattern

	Factor1	Factor2
para	0.86556	0.08098
sent	0.81899	0.17284
word	0.81804	0.07868
add	0.14966	0.73801
dots	0.05112	0.78274

Now rather clearer that factor 1 is verbal ability and factor 2 mathematical.

Final Communality Estimates: Total = 3.314090

para	sent	word	add	dots
0.75574929	0.70062380	0.67537332	0.56705315	0.61529069

Communalities unaffected by rotation.

A bigger example: BEM sex role inventory

- 369 women asked to rate themselves on 44 traits, like "self-reliant" or "shy".
- Rating 1 "never or almost never true of me" to 7 "always or almost always true of me".
- 44 personality traits is a lot. Can we find a smaller number of factors that capture aspects of personality?
- The whole BEM sex role inventory on next page.

The whole inventory

1. self reliant	21.reliable	41.warm
2. yielding	22.analytical	42.solemn
3. helpful	23.sympathetic	43.willing to take a stand
4. defends own beliefs	24.jealous	44.tender
	25.leadership ability	45.friendly
5. cheerful	26.sensitive to other's needs	46.aggressive
6. moody	27.truthful	47.gullible
7. independent	28.willing to take risks	48.inefficient
8. shy	29.understanding	49.acts as a leader
9. conscientious	30.secretive	50.childlike
10.athletic	31.makes decisions easily	51.adaptable
11.affectionate	32.compassionate	52.individualistic
12.theatrical	33.sincere	53.does not use harsh language
13.assertive	34.self-sufficient	
14.flatterable	35.eager to soothe hurt feelings	54.unsystematic
15.happy		55.competitive
16.strong personality	36.conceited	56.loves children
17.loyal	37.dominant	57.tactful
18.unpredictable	38.soft spoken	58.ambitious
19.forceful	39.likable	59.gentle
20.feminine	40.masculine	60.conventional

Reading a SAS data set

- Data come to us as a SAS data set (somebody else has read the numbers in from a file and created a SAS data set, which they saved).
- First step is to specify the `libname`, where the data set file is, which is usually in same folder as code. This can be given any name, like `fred`, resulting in

```
libname fred '.';
```

- Then data step only needs to contain one line (no infile, input etc):

```
data x;
  set fred.datasetname;
```

links our SAS data set `x` to SAS data set file `datasetname` in current directory (folder).

More; number of factors

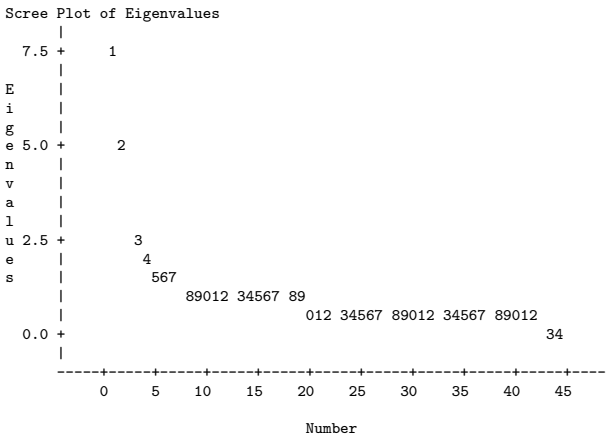
- In our case, data in file `factor.sas7bdat`, so code as below. Also, data step can contain other things like defining new variables, or `drop` variables we don't need.

```
libname sasdata '.';

data bem;
  set sasdata.factor;
```

- Run PROC FACTOR with scree plot, look at eigenvalues.
- No rotation yet, since interpretation later.

```
proc factor scree method=prinit;
```

Scree plot



Scale makes it hard to tell, but might be an elbow at 5, favouring 4 factors.

The eigenvalues

Preliminary Eigenvalues: Total = 44 Average = 1

	Eigenvalue	Difference	Proportion	Cumulative
1	7.53227628	2.51208242	0.1712	0.1712
2	5.02019387	2.61617135	0.1141	0.2853
3	2.40402251	0.33369433	0.0546	0.3399
4	2.07032818	0.37202817	0.0471	0.3870
5	1.69830001	0.28605615	0.0386	0.4256
6	1.41224387	0.06851943	0.0321	0.4577
7	1.34372444	0.18080134	0.0305	0.4882
8	1.16292310	0.01544149	0.0264	0.5146
9	1.14748161	0.04705296	0.0261	0.5407
10	1.10042865	0.02197431	0.0250	0.5657
11	1.07845434	0.07540628	0.0245	0.5902
12	1.00304806	0.04746411	0.0228	0.6130
13	0.95558395	0.03978141	0.0217	0.6348
14	0.91580253	0.05321790	0.0208	0.6556
15	0.86258464	0.01134500	0.0196	0.6752
16	0.85123963	0.03066264	0.0193	0.6945
...				
43	0.23079710	0.08266928	0.0052	0.9966
44	0.14812782		0.0034	1.0000

Interpreting eigenvalues

- No "obvious" gaps – maybe first 2 eigenvalues bigger than others (but then only 28.5% of variability explained).
- Scree plot said 4 eigenvalues before "elbow".
- 12 eigenvalues > 1, even then only 61.3% of variability explained.
- Personality is complicated, multidimensional thing.

Extract 4 factors for interpretation

- Specify to extract 4 factors.
- Aim for interpretation of them: rotation (varimax).
- Plot factor scores for first 2.
- Code:

```
proc factor method=prinit n=4 rotate=varimax out=fred;

goptions reset=all;
symbol1 pointlabel=('#subno');

proc gplot data=fred;
  plot Factor2*Factor1;
```

- Make plot of factor scores labelled by subject numbers.

Rotated factor pattern

	Factor1	Factor2	Factor3	Factor4
HELPFUL	0.26184	0.26300	0.27923	0.20967
RELIANT	0.36213	0.07112	0.11709	**0.43997**
DEFBEL	**0.42138**	0.01991	0.27629	0.07063
YIELDING	-0.14990	0.31860	0.15308	0.04241
CHEERFUL	0.14162	**0.50944**	0.02272	0.11443
INDPT	**0.44735**	0.00272	0.01255	**0.43723**
ATHLET	0.30056	0.22166	-0.10326	-0.03315
SHY	**-0.40567**	-0.07819	-0.04059	-0.05705
ASSERT	**0.63003**	-0.04904	0.12778	-0.02520
STRPERS	**0.70736**	0.00870	0.05617	-0.07512
FORCEFUL	**0.67282**	-0.18610	0.04465	-0.03587
AFFECT	0.25423	**0.47711**	0.32397	-0.30032
FLATTER	0.18401	0.26908	0.06747	-0.30375
LOYAL	0.17038	0.31797	0.27964	-0.07210
ANALYT	0.28690	-0.00555	0.19432	0.05692

More

FEMININE	0.06328	0.27971	0.18228	0.15442
SYMPATHY	-0.02104	0.13347	**0.65757**	-0.00735
MOODY	0.05025	-0.32997	0.11292	-0.34756
SENSITIV	0.08165	0.04258	**0.59779**	0.06167
UNDSTAND	0.01071	0.22379	**0.68323**	0.14200
COMPASS	0.05335	0.18929	**0.75108**	0.04977
LEADERAB	**0.70626**	0.04234	0.08985	0.20489
SOOTHE	0.03670	0.31150	**0.53622**	-0.05341
RISK	**0.45177**	0.14371	0.09032	0.02003
DECIDE	**0.47222**	0.10438	0.06711	0.35742
SELFSUFF	**0.39617**	0.10659	0.08957	**0.63085**
CONSCIEN	0.21155	0.16877	0.28705	**0.43193**
DOMINANT	**0.67958**	-0.26115	-0.05550	0.02484
MASCULIN	0.30166	-0.29009	-0.09734	-0.06293
STAND	**0.58910**	0.03865	0.22935	0.14560
HAPPY	0.11130	**0.62439**	-0.00707	0.12417

More

SOFTSPOK	-0.30162	0.30583	0.13379	0.22252
WARM	0.09721	**0.61767**	**0.39400**	-0.12470
TRUTHFUL	0.08921	0.20685	0.23252	0.07630
TENDER	0.07217	**0.60209**	0.37809	-0.10875
GULLIBLE	-0.07654	0.14233	0.04295	-0.36485
LEADACT	**0.71462**	0.00697	-0.02843	0.17498
CHILDLIK	0.00468	-0.07610	-0.07340	**-0.40445**
INDIV	**0.43371**	0.10224	0.03320	0.18009
FOULLANG	-0.00735	0.16780	0.01744	0.03762
LOVECHIL	0.00090	0.30809	0.13968	-0.09332
COMPETE	**0.50472**	0.19757	-0.11419	-0.06369
AMBITIOU	**0.41041**	0.18988	0.00370	0.11983
GENTLE	-0.02111	**0.61269**	0.35327	-0.03461

Interpretation

- I used 0.40 (or close) as cutoff.
- Factor 1: defends own beliefs, independent, not-shy, assertive, strong personality, forceful, has leadership ability, takes risks, is decisive, self-sufficient, dominant, willing to take a stand.
- Factor 2: cheerful, affectionate, happy, warm, tender, gentle.
- Factor 3: sympathetic, sensitive, understanding, compassionate, soothes hurt feelings, warm.
- Factor 4: self-reliant, independent, self-sufficient, conscientious, not-childlike.
- Decide for yourself what traits in each factor have in common!
- Some traits appear in more than one factor, some in none.

Communalities

```
   HELPFUL      RELIANT       DEFBEL     YIELDING     CHEERFUL        INDPT
0.25966592   0.34347676   0.25927866   0.14921084   0.29319613   0.39145262

    ATHLET          SHY       ASSERT      STRPERS     FORCEFUL       AFFECT
0.15123194   0.17558566   0.41630723   0.50923469   0.49059572   0.48740930

   FLATTER        LOYAL       ANALYT     FEMININE     SYMPATHY        MOODY
0.20308368   0.21353167   0.12334064   0.13931465   0.45071468   0.24495805

  SENSITIV     UNDSTAND      COMPASS     LEADERAB       SOOTHE         RISK
0.36963797   0.53717166   0.60527481   0.55064036   0.38876534   0.23331239

    DECIDE     SELFSUFF     CONSCIEN     DOMINANT     MASCULIN        STAND
0.36614500   0.57430213   0.34219775   0.53373041   0.18858344   0.42233319

     HAPPY     SOFTSPOK         WARM     TRUTHFUL       TENDER     GULLIBLE
0.41771222   0.25191883   0.56174959   0.11063193   0.52249779   0.16107602

   LEADACT     CHILDLIK        INDIV     FOULLANG     LOVECHIL      COMPETE
0.54215483   0.17477789   0.23208669   0.02992972   0.12313782   0.31086844

                          AMBITIOU        GENTLE
                        0.21885749    0.50182441
```

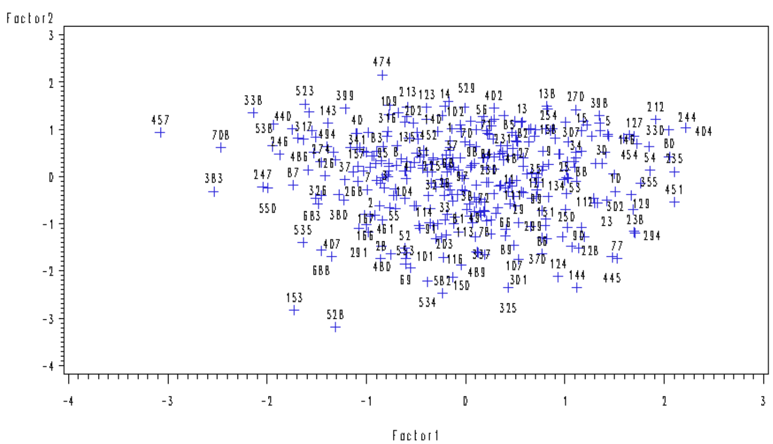

Exploratory factor analysis

Interpreting communalities

- Low communality means variable not related to any factor.
- Eg. yielding, athletic, shy, feminine, masculine, truthful, gullible, childlike, uses foul language (very low), loves children.
- Large number of low communalities means that more factors necessary to describe data well.

Exploratory factor analysis

Factor scores plot



Unusual subjects: 474, 457, 528.

Exploratory factor analysis

Looking at unusual subjects

Create new data set by picking out these subjects, and look at:

```
data fred2;
    set fred;
    if subno=474 or subno=457 or subno=528;

proc print;
```

```
            Y C           F         F S   S U   L
      H R   I H         S O   F     E Y   E N C E
      E E D E E   A   A T R A L   A M M   N D O A S
   S  L L E L E I T   S R C F A L N I P M S S M D O
   U  P I F D R N H   S P E F T O A N A O I T P E O R
O  B  F A B I F D L S E E F E T Y L I T O T A A R T I
b  N  U N E N U P E H R R U C E A Y N H D I N S A H S
s  O  L T L G L T T Y T S L T R L T E Y Y V D S B E K

1 457 6 4 1 7 5 7 7 7 3 1 1 4 1 7 4 4 7 3 7 7 6 1 7 5
2 474 6 6 6 6 5 7 5 2 3 1 3 6 6 6 6 7 7 7 7 7 7 5 7 3
3 528 6 7 3 6 5 6 5 5 3 4 3 4 3 6 5 3 3 3 3 3 3 3 3 2
```

The unusual individuals

```
      S C D M     S   T   G   C   F L   A           _
      E O O A     O   R   U L H   O O C M           O
    D L N M S     F   U T L E I   U V O B G         B
    E F S I C S H T   T E L A L I L E M I E         S
    C S C N U T A S W H N I D D N L C P T N         T
O I U I A L A P P A F D B A L D A H E I T           A
b D F E N I N P O R U E L C I I N I T O L           T
s E F N T N D Y K M L R E T K V G L E U E           _

1 1 4 7 1 1 1 6 6 6 5 6 7 1 1 3 4 7 2 2 7 01101    0    0    0
2 3 7 7 3 3 7 3 7 7 7 7 4 4 4 7 4 7 3 6 7 01101    0    0    0
3 3 6 6 4 2 3 7 5 4 6 4 5 1 3 6 4 4 6 6 5 01101    0    0    0

      F           F           F           F           F
      a           a           a           a           a
      c           c           c           c           c
      t           t           t           t           t
O     o           o           o           o           o
b     r           r           r           r           r
s     1           2           3           4           5

1 -3.07459     0.93204    -0.45515    -0.54083     0.08734
2 -0.84162     2.15352    -1.07810     0.02538     0.71570
3 -1.31338    -3.18950    -0.10796     0.31081     0.66190
```


What makes them unusual

- #457 (low on F1): defends own beliefs (1), independent (7!), not-shy (shy=7), assertive (3), strong personality (1), forceful (1), has leadership ability (1), takes risks (5), is decisive (1), self-sufficient (4), dominant (1), willing to take a stand (1).
- #474 (high on F2): cheerful (5), affectionate (6), happy (3), warm (7), tender (7), gentle (7).
- #528 (low on F2) cheerful (5), affectionate (4), happy (7!), warm (4), tender (4), gentle (5).

#528's values are low for those variables.

12 factors

Just for fun, I tried 12 factors (the number of eigenvalues > 1). High loadings (bigger than 0.5) are now:

1. assertive, strong personality, forceful, dominant
2. sympathetic, sensitive, understanding, compassionate, soothes hurt feelings
3. affectionate, loyal, warm, tender, gentle (0.48)
4. self-reliant, independent, self-sufficient
5. competitive, ambitious, athletic (0.33), takes risks (0.36)
6. cheerful, not-moody, happy
7. leadership ability, acts like a leader, dominant (0.34)
8. feminine, not-masculine (0.38)
9. soft-spoken, gentle (0.48)
10. willing to take a stand (0.47), truthful (0.43), defends own beliefs (0.35), not-gullible (0.30)
11. childlike, not-self-sufficient (0.30)
12. decisive, takes risks (0.34), willing to take a stand (0.30)

Where we are going

Confirmatory factor analysis

- Exploratory: what do data suggest as hidden underlying factors (in terms of variables observed)?
- Confirmatory: have *theory* about how underlying factors depend on observed variables; test whether theory supported by data:
  - does theory provide *some* explanation (better than nothing)
  - can we do better?
- Also can compare two theories about factors: is more complicated one significantly better than simpler one?

Children and tests again

- Previously had this data (based on 145 children):

```
para 1     0.722 0.714 0.203 0.095
sent 0.722 1     0.685 0.246 0.181
word 0.714 0.685 1     0.170 0.113
add  0.203 0.246 0.170 1     0.585
dots 0.095 0.181 0.113 0.585 1
```

- SAS: use `type=corr`. Special variable `_NAME_` for reading in variable names; numbers read as correlations by default.
- Now have to specify sample size. Now have to use special variable `_TYPE_` which is CORR for correlation, N for sample size.
- Only one sample size, but need to be 5 values: others can be missing.

New data file and code

Note that sample size has no variable name (all variables have $n = 145$):

```
n . 145 . . . .
corr para 1 0.722 0.714 0.203 0.095
corr sent 0.722 1 0.685 0.246 0.181
corr word 0.714 0.685 1 0.170 0.113
corr add 0.203 0.246 0.170 1 0.585
corr dots 0.095 0.181 0.113 0.585 1
```

Read it in with

```
data rex(type=corr);
  infile "rex3.dat";
  input _type_ $ _name_ $ para sent word add dots;
```


How to specify theories

- SAS uses PROC CALIS for confirmatory factor analysis (and many other things besides).
- Specify relationship between variables and factors (looks like regression analysis with "error").
- Two competing theories:
  - One-factor "general intelligence" model: all the test scores are high or low together for a child.
  - Two-factor "verbal and mathematical intelligence" model: a child might be good at the verbal tests, or good at the mathematical tests (or both or neither). These are 2 factors we found before.

Code for the 1-factor model

Specify how each variable related to the factor(s) hypothesized. I use symbol `f` for common factor(s) and `e` for specific factors.

```
proc calis method=lsml;
  lineqs
    para=x1 f1 + e1,
    sent=x2 f1 + e2,
    word=x3 f1 + e3,
    add =x4 f1 + e4,
    dots=x5 f1 + e5;
  std
    f1=1,
    e1-e5=eps1-eps5;
  bounds
    eps1-eps5>0;
```

Note punctuation in `lineqs` section (and other sections): commas at end of each line, except semicolon at end of last.

Output (heavily edited)

To start:

```
                    The 5 Endogenous Variables

Manifest        para  sent  word  add   dots
Latent

                    The 6 Exogenous Variables
Manifest
Latent          f1
Error           e1    e2    e3    e4    e5
```

- "Endogenous" means "going in".
- "Manifest" means "observed".
- "Latent" means "not able to be observed".
- "Exogenous" means "coming out".
- Original variables are endogenous and manifest.
- Factors are exogenous and latent (or "error", for specific factors).

Did it converge?

Look for "maximum likelihood estimation":

Iter	Restarts	Function Calls	Active Constraints	Objective Function	Objective Function Change	Max Abs Gradient Element	Lambda	Ratio Between Actual and Predicted Change
1	0	2	0	0.41335	0.0104	0.0256	0	1.206
2	0	3	0	0.41302	0.000329	0.00349	0	1.174
3	0	4	0	0.41301	9.497E-6	0.000603	0	1.171
4	0	5	0	0.41301	2.771E-7	0.000099	0	1.171
5	0	6	0	0.41301	8.072E-9	0.000017	0	1.171
6	0	7	0	0.41301	2.35E-10	2.905E-6	0	1.171

Optimization Results

Iterations	6	Function Calls	8
Jacobian Calls	7	Active Constraints	0
Objective Function	0.4130083436	Max Abs Gradient Element	2.9047445E-6
Lambda	0	Actual Over Pred Change	1.1706449333
Radius	0.0000463356		

GCONV convergence criterion satisfied.

Answer: yes. Objective function stopped changing, and the largest gradient element very close to 0. Also, see last line.

Assessing and testing the fit

There follows a long list of things, of which we need only these:

Goodness of Fit Index (GFI)	0.8764
GFI Adjusted for Degrees of Freedom (AGFI)	0.6291
Chi-Square	59.4732
Chi-Square DF	5
Pr > Chi-Square	<.0001
Independence Model Chi-Square	298.65
Independence Model Chi-Square DF	10

- GFI and AGFI like R-squared and adjusted R-squared in regression.
- AGFI quite a bit smaller here because we estimated a lot of things.
- Model that fits perfectly has 0 DF.
- 1st chi-square and P-value says "are we significantly worse than perfect", ie. "can we do better"? Answer here "yes".

Are we better than nothing?

Chi-Square	59.4732
Chi-Square DF	5
Pr > Chi-Square	<.0001
Independence Model Chi-Square	298.65
Independence Model Chi-Square DF	10

- Independence model has no common factors (only specific factors), so by comparing our model chisquare and DF with it, we answer "are we better than nothing?". Take difference of chi-squares, $298.65 - 59.47 = 239.18$, difference of DF, $10 - 5 = 5$ to get very small P-value.
- 1-factor model doing better than nothing, but can do better.

Improving the model

Obvious way to improve things: original idea of 2 common factors, one verbal (para, sent, words), one mathematical (add, dots). Code for that:

```
proc calis method=lsml;
  lineqs
    para=x1 f1 + e1,
    sent=x2 f1 + e2,
    word=x3 f1 + e3,
    add =x4 f2 + e4,
    dots=x5 f2 + e5;
  std
    f1=1,
    f2=1,
    e1-e5=eps1-eps5;
  bounds
    eps1-eps5>0;
  cov
    f1 f2 = rho;
```


Endogenous and exogenous variables

The 5 Endogenous Variables

Manifest	para	sent	word	add	dots
Latent					

The 7 Exogenous Variables

Manifest					
Latent	f1	f2			
Error	e1	e2	e3	e4	e5

Now 2 exogenous latent variables (common factors).

Convergence

All good:

Iter	Rest arts	Func Calls	Act Con	Objective Function	Obj Fun Change	Max Abs Gradient Element	Lambda	Actual Over Pred Change
1	0	2	0	0.02038	0.00325	0.00679	0	1.019
2	0	3	0	0.02035	0.000026	0.000721	0	1.028
3	0	4	0	0.02035	2.16E-7	0.000043	0	1.058
4	0	5	0	0.02035	1.61E-9	5.325E-6	0	1.081

Optimization Results

Iterations	4	Function Calls	6
Jacobian Calls	5	Active Constraints	0
Objective Function	0.0203513722	Max Abs Gradient Element	5.3251548E-6
Lambda	0	Actual Over Pred Change	1.0814713689
Radius	0.0008266204		

ABSGCONV convergence criterion satisfied.

Quality of fit

Goodness of Fit Index (GFI)	0.9919
GFI Adjusted for Degrees of Freedom (AGFI)	0.9697

GFI and (especially) AGFI much better than 0.88 and 0.63 from before. Near-perfect fit.

Chi-Square	2.9306
Chi-Square DF	4
Pr > Chi-Square	0.5695

No longer significantly worse than perfect fit: no point trying to do better.

Better than nothing?

Predictably yes:

Chi-Square	2.9306
Chi-Square DF	4
Pr > Chi-Square	0.5695
Independence Model Chi-Square	298.65
Independence Model Chi-Square DF	10

Chi-square $298.65 - 2.93 = 295.72$ with $10 - 4 = 6$ DF. P-value extremely small.

Communalities and estimated correlation

Squared Multiple Correlations

	Variable	Error Variance	Total Variance	R-Square
1	para	0.25049	1.00000	0.7495
2	sent	0.30038	1.00000	0.6996
3	word	0.32651	1.00000	0.6735
4	add	0.04949	1.00000	0.9505
5	dots	0.63996	1.00000	0.3600

Correlations Among Exogenous Variables

Var1	Var2	Parameter	Estimate
f1	f2	rho	0.25197

Communalities (in R-squared column) nice and high (possibly excepting DOTS). Correlation between factors estimated at 0.25.

Using SAS to figure out those P-values

To save hauling out your calculator and tables to figure out the comparison between 298.65 with 10 DF and 2.9306 with 4 DF, make a file `stat.dat` with this in it:

```
298.65 10 2.9306 4
```

and a file `stat.sas` with this in it:

```
data xx;
  infile "stat.dat";
  input c1 df1 c2 df2;
  mystat=c1-c2;
  mydf=df1-df2;
  pval=1-probchi(mystat,mydf);

proc print;
```

This works out the P-value in `pval`; printing out the whole "data set" shows it to you.

The P-value

Obs	c1	df1	c2	df2	mystat	mydf	pval
1	298.65	10	2.9306	4	295.719	6	0

. . . is close to 0.
Can also compare the 1- and 2-factor models to see if the 2-factor one fits significantly better. The chi square statistics are 59.4732 with 5 DF and 2.93 with 4 DF, so change `stat.dat` to read `59.4372 5 2.93 4` and re-run to get:

Obs	c1	df1	c2	df2	mystat	mydf	pval
1	59.4372	5	2.9306	4	56.5066	1	5.5955E-14

P-value is the merest smidgen bigger than 0. The 2-factor model is a significantly better description of the data than the 1-factor.

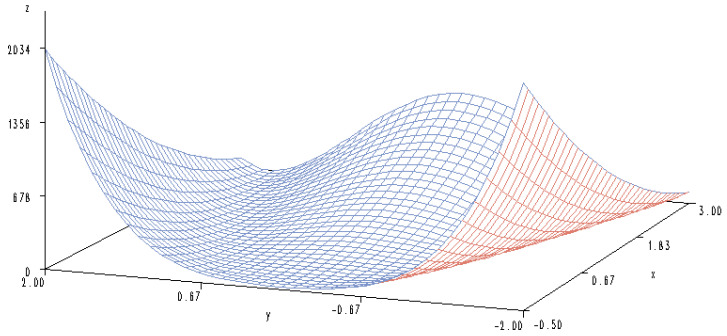
Where we are going

Spatial statistics

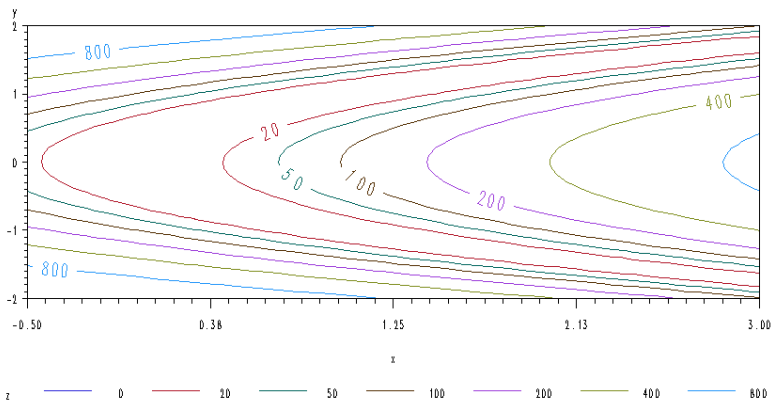
- Concerned with data on (at least) 3 variables:
  - 2 measure location in space
  - others measure some features of that location.
- Related to GIS (SAS does, *if* you have licence).
- Our aim: data at some locations, estimate what data would be at larger set of locations.
- Summarize in 3D or contour plot.
- Concern: data from nearby points probably *correlated*.

A 3D plot

Some data with `x`, `y` as location, `z` as height. Draw picture of surface:



The same, as contour plot



More frequently...

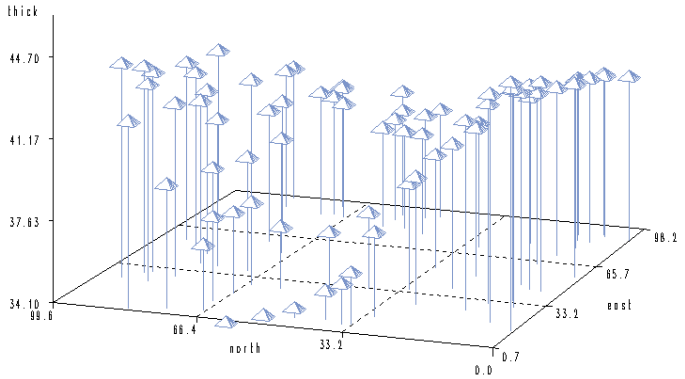
- have data at set of (possibly irregular set of) locations
- want to estimate the surface, and make plot, allowing for spatial autocorrelation.
- Estimation has two stages:
  - estimate autocorrelation structure and nature of any anisotropy (`proc variogram`)
  - feed these into estimation of entire surface (`proc krige2d`), a procedure called **kriging**.
- Kriging based on idea that degree of correlation between pair of measurements based only on *distance* between them, not direction or absolute locations.

Example data

- Thickness of coal seam measured at various locations in a coalfield.
- Aim: find where coal seam thickest, most profitable to mine.
- Do this by estimating thickness everywhere and making plot.
- First step: read in data and make 3D plot:

```
data thick;
  input east north thick @@;
  datalines;
   0.7  59.6  34.1   2.1  82.7  42.2   4.7  75.1  39.5
  ...
  91.5  55.4  39.0  92.9  46.8  39.1  93.4  70.9  39.7
  94.8  71.5  39.7  96.2  84.3  40.3  98.2  58.2  39.5
;

proc g3d;
    scatter north*east=thick;
```

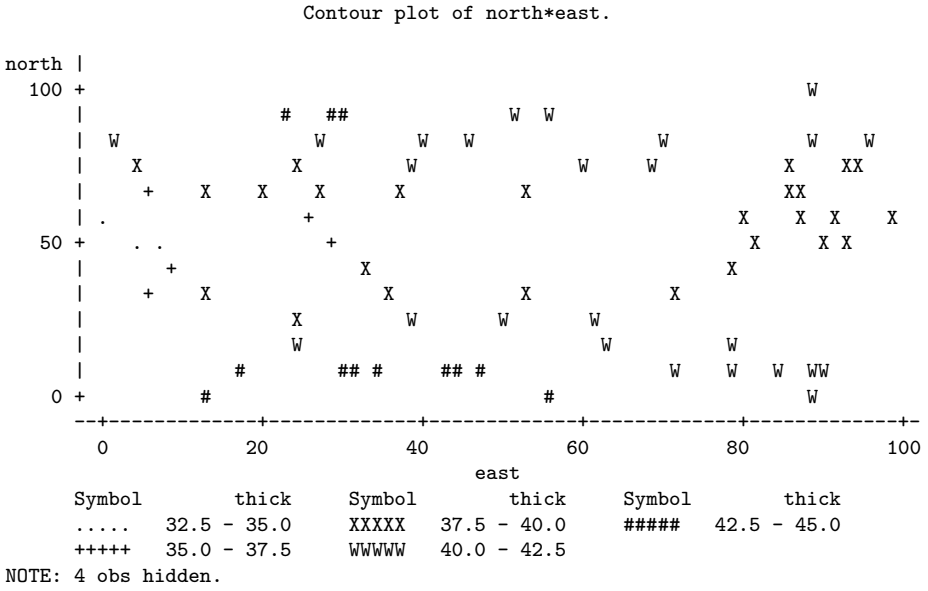

3D plot of data



or get (crude) contour plot:

```
proc plot;
    plot north*east=thick / contour=5;
```

Contour plot



Discussion

- Could be kind of parabolic surface.
- Or: mean height actually *constant* with local deviations from it, consistently in same direction as nearby ones (spatial autocorrelation).
- Hard to tell difference (like time series: is mean changing, or is pattern caused by autoregressive/moving average process?)
- If there is eg. linear trend, fit it first, and work with residuals from this regression.

The variogram, first run

- Determines whether/how correlation depends on distance.
- Needs two options `lagdistance` and `maxlags`, but start with no idea about them. Make SAS run with defaults to get sense of what they should be:

```
proc variogram;
    compute novariogram;
    coordinates xc=east yc=north;
    var thick;
```


First run output

Pairwise Distance Intervals

Lag Class	Bounds		Number of Pairs	Percentage of Pairs
0	0.00	6.97	45	1.62%
1	6.97	20.91	263	9.48%
2	20.91	34.84	383	13.80%
3	34.84	48.78	436	15.71%
4	48.78	62.72	495	17.84%
5	62.72	76.66	525	18.92%
6	76.66	90.60	412	14.85%
7	90.60	104.53	179	6.45%
8	104.53	118.47	35	1.26%
9	118.47	132.41	2	0.07%
10	132.41	146.35	0	0.00%

- Max distance between pair of points between 118 and 132. SAS divides distances (by default) into $10 + 1$ classes, and counts # point pairs in each.

More classes?

- Want:
  - as many lag classes as possible
  - at least 30 pairs in each class (except class 0), up to reasonable distance.
- Can certainly use more classes here. Try 30:

```
proc variogram;
     compute nhclasses=30 novariogram;
     coordinates xc=east yc=north;
     var thick;
```


30 classes

Pairwise Distance Intervals

Lag Class	Bounds		Number of Pairs	Percentage of Pairs
0	0.00	2.32	4	0.14%
1	2.32	6.97	41	1.48%
2	6.97	11.61	69	2.49%
3	11.61	16.26	86	3.10%
4	16.26	20.91	108	3.89%
5	20.91	25.55	120	4.32%
...				
13	58.07	62.72	209	7.53%
...				
21	95.24	99.89	60	2.16%
22	99.89	104.53	30	1.08%
23	104.53	109.18	19	0.68%
24	109.18	113.83	11	0.40%
25	113.83	118.47	5	0.18%
26	118.47	123.12	1	0.04%
27	123.12	127.76	1	0.04%
28	127.76	132.41	0	0.00%
29	132.41	137.06	0	0.00%
30	137.06	141.70	0	0.00%

- All right. Usually lag class 1 is limiting factor; wouldn't want much smaller.

The next stage

- Output also includes:

Pairs Information

Number of Lags	31
Lag Distance	4.65

- This is what we want for `lagdistance`: round up to 5 to be safe.
- `maxlags` is (a bit less than) highest numbered lag class with ≥ 30 point pairs in it: here 22, round down to 20.
- Save output data set with variogram in it, print:

```
proc variogram data=thick outv = outv;
  compute lagdistance = 5 maxlag = 20;
  coordinates xc=east yc=north;
  var thick;

proc print;
```


Output data set

SAS produces some output, repeated in output data set:

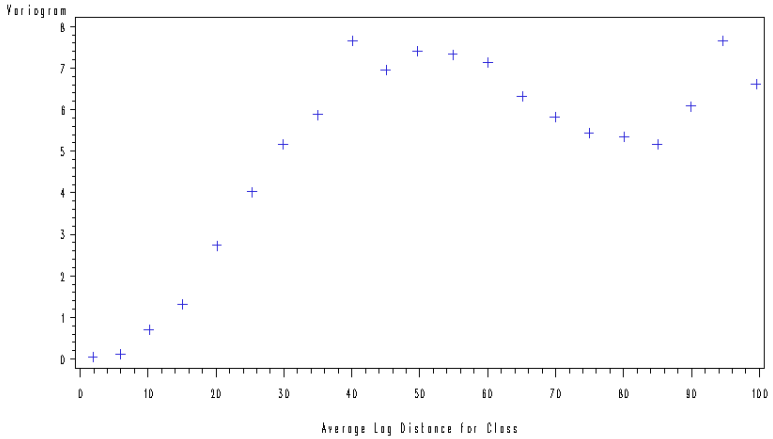
Obs	VARNAME	LAG	COUNT	DISTANCE	AVERAGE	VARIOG	STDERR	COVAR
1	thick	-1	75	.	40.1387	.	.	5.59592
2	thick	0	4	1.8919	40.1250	0.04250	0.03005	6.73456
3	thick	1	51	5.9367	40.4382	0.12363	0.02448	5.54671
4	thick	2	76	10.1651	40.0428	0.70243	0.11395	3.72434
5	thick	3	104	15.1243	40.1115	1.31000	0.18166	3.29897
6	thick	4	123	20.1472	40.0516	2.73240	0.34842	2.68629
7	thick	5	136	25.3109	39.8081	4.02140	0.48767	1.88510
8	thick	6	130	29.8661	39.8746	5.16485	0.64062	0.64092
9	thick	7	150	35.0573	39.8130	5.88077	0.67905	-0.51211
10	thick	8	137	40.1762	39.9540	7.65146	0.92448	-1.93853
11	thick	9	163	45.0273	39.8837	6.95408	0.77030	-1.85804
12	thick	10	165	49.6994	39.8558	7.40564	0.81533	-2.31356
13	thick	11	159	54.8782	39.8881	7.32824	0.82189	-2.23589
14	thick	12	219	60.0973	40.0637	7.13244	0.68160	-2.08081
15	thick	13	194	65.1025	40.2987	6.31673	0.64137	-1.71279
16	thick	14	180	69.9306	40.2514	5.81919	0.61340	-0.91277
17	thick	15	190	74.9328	40.3763	5.43221	0.55733	0.05297
18	thick	16	155	80.1055	40.4206	5.35065	0.60779	0.36238
19	thick	17	151	85.0293	40.4940	5.15768	0.59358	1.69427
20	thick	18	117	89.9044	40.2175	6.08030	0.79496	0.98993
21	thick	19	73	94.6578	40.1733	7.66295	1.26838	0.18459
22	thick	20	47	99.5352	40.8447	6.61277	1.36411	0.30689

Discussion

- Variogram itself in `variog`.
- `stderr` says how accurately variogram estimated. If too big at end, reduce `maxlag`.
- Plot should go up to a limit, except for sampling error.
- Plot:

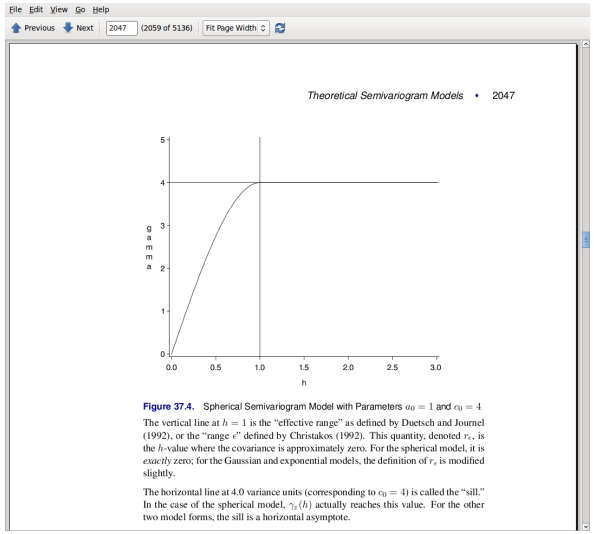
```
proc gplot;
    plot variog*distance;
```


Variogram plot



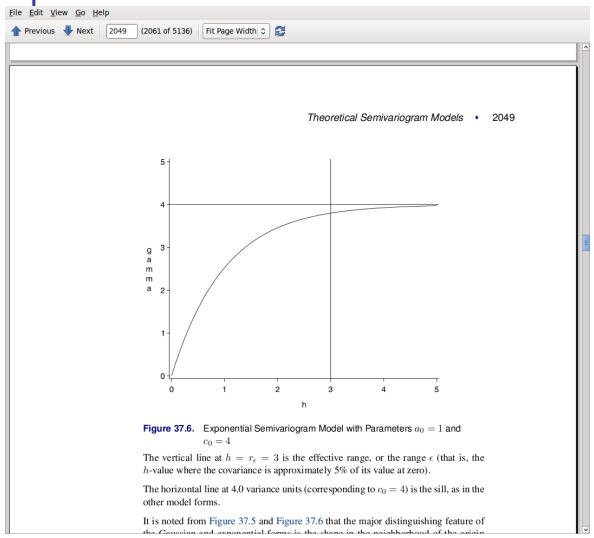
- Now need to decide on *shape*.

Spherical



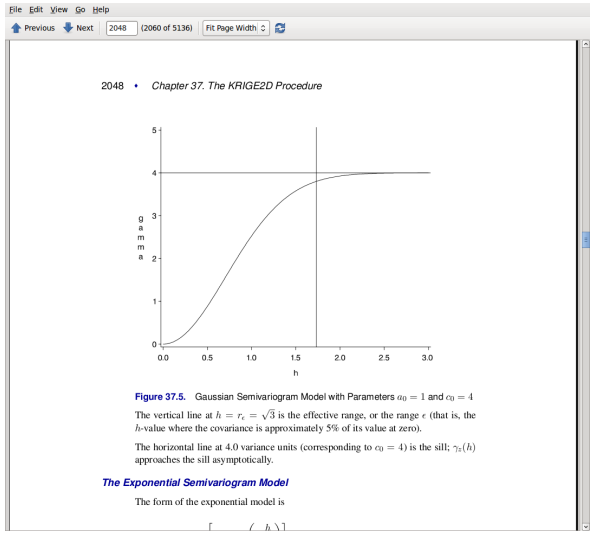
Range is 1, scale is 4.
Rises fast, then levels off abruptly.

Exponential



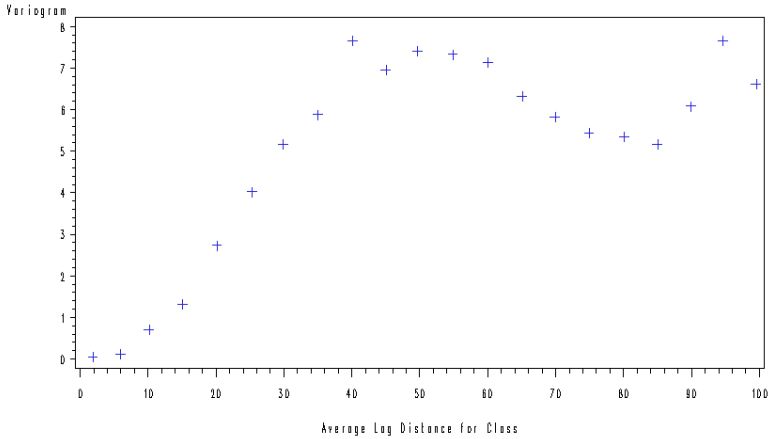
Range is 1, scale is 4. (Get range as distance where height is 95% of max, divided by 3.)
Rises fast, then approaches limit gradually.

Gaussian



Range is 1, scale is 4. (Range is distance where height is 95% of max, divided by 1.7).
Rises slow then faster, approaches limit gradually.

Returning to our data



Slow rise at start suggests Gaussian model, max height (scale) about 7, range about $50/1.7 = 30$. Feed these into kriging routine.

Kriging code

```
proc krige2d data=thick outest=est;
  coord xc=east yc=north;
  grid x=0 to 100 by 5 y=0 to 100 by 5;
  pred var=thick r=10;
  model scale=7 range=30 form=gauss;

proc print data = est (obs = 10);
```


Output and output data set (part)

Covariance Model Information

Type	Gaussian
Sill	7
Range	30
Effective Range	51.961524

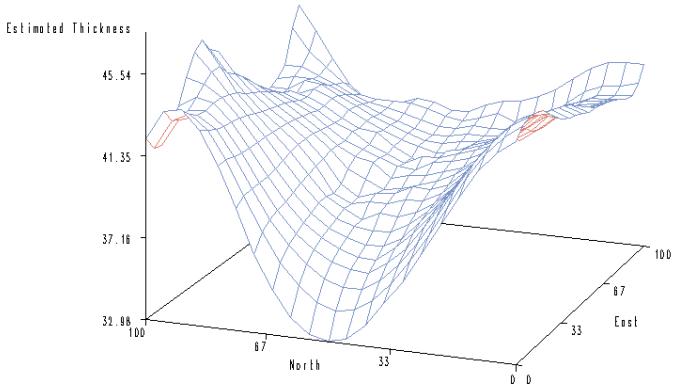
Obs	LABEL	VARNAME	GXC	GYC	NPOINTS	ESTIMATE	STDERR
1	Pred1.Model1	thick	0	0	20	44.0107	0.66714
2	Pred1.Model1	thick	0	5	20	43.3504	0.65143
3	Pred1.Model1	thick	0	10	20	42.3169	0.59026
4	Pred1.Model1	thick	0	15	20	40.9308	0.52172
5	Pred1.Model1	thick	0	20	20	39.4097	0.36240
6	Pred1.Model1	thick	0	25	20	37.8804	0.22627
7	Pred1.Model1	thick	0	30	20	36.3949	0.15932
8	Pred1.Model1	thick	0	35	20	35.2236	0.10873
9	Pred1.Model1	thick	0	40	20	33.9929	0.06815
10	Pred1.Model1	thick	0	45	20	33.2266	0.05748

Making plots

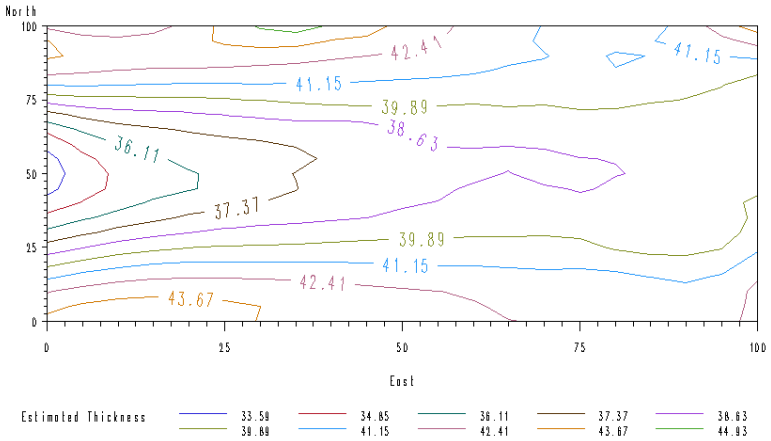
Plot as 3D plot and contour plot:

```
proc g3d data=est;
  plot gyc*gxc=estimate;
  label gyc      = 'North'
        gxc      = 'East'
        estimate = 'Estimated Thickness';

proc gcontour data=est;
  plot gyc*gxc=estimate / nlevels=10 autolabel;
  label gyc      = 'North'
        gxc      = 'East'
        estimate = 'Estimated Thickness';
```

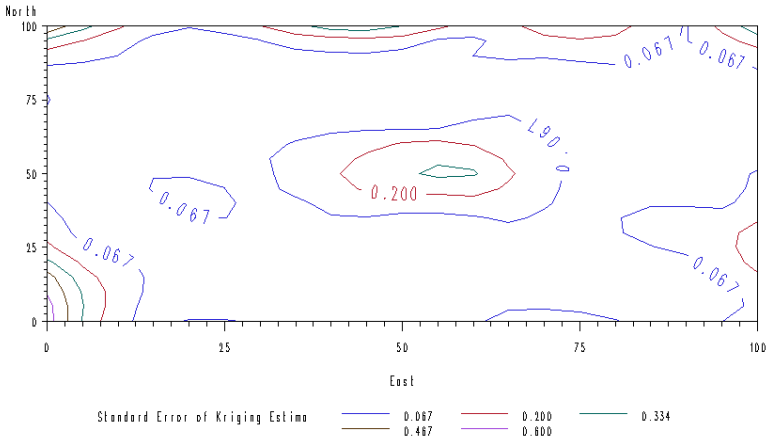

3D plot



Contour plot



Contour plot of SE of estimate



Where we are going

Multi-way frequency analysis

- A study of gender and eyewear-wearing finds the following frequencies:

Gender	Contacts	Glasses	None
Female	121	32	129
Male	42	37	85

- Is there association between eyewear and gender?
- Normally answer this with chisquare test (based on observed and expected frequencies from null hypothesis of no association).
- Two categorical variables and a frequency.
- We assess in way that generalizes to more categorical variables.

Data format

Data file like this:

```
female contacts 121
female glasses   32
female none     129
male   contacts  42
male   glasses   37
male   none      85
```

as the two categorical variables (gender, type of eyewear) and frequency (number of observations in that category combination).

Some code, using PROC CATMOD

```
data lens;
  infile "lenswear.dat";
  input sex $ lenswear $ frequency;

proc catmod;
  weight frequency;
  model sex*lenswear=_response_;
  loglin sex lenswear sex*lenswear;
```

In PROC CATMOD, specify frequency, then SAS black magic to get right thing, then model (on LOGLIN line!).

Maximum likelihood analysis

Maximum Likelihood Analysis

Maximum likelihood computations converged.

Maximum Likelihood Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
sex	1	16.10	<.0001
lenswear	2	64.63	<.0001
sex*lenswear	2	17.16	0.0002
Likelihood Ratio	0	.	.

- Conclude from `sex*lenswear` line that interaction is significant.
- That is, frequency depends on the sex-lenswear *combination* (not just on either variable singly).
- Or, there is association between sex and lenswear (as usual chisquare test concludes).

Understanding the association

Analysis of Maximum Likelihood Estimates

Parameter		Estimate	Standard Error
sex	female	0.2217	0.0552
lenswear	contacts	0.1146	0.0757
	glasses	-0.6138	0.0889
sex*lenswear	female contacts	0.3074	0.0757
	female glasses	-0.2943	0.0889

Estimates over each variable sum to 0, so complete table as over.

All the estimates

Parameter		Estimate	Error
sex	female	0.2217	0.0552
	male	-0.2217	
lenswear	contacts	0.1146	0.0757
	glasses	-0.6138	0.0889
	none	0.4992	
sex*lenswear	female contacts	0.3074	0.0757
	female glasses	-0.2943	0.0889
	female none	-0.0131	
	male contacts	-0.3074	
	male glasses	0.2943	
	male none	0.0131	

- Look for large (plus or minus) estimates.
- Females more likely to wear contacts and males glasses than expected (if no association).
- Overall, more females in study, and people less likely to wear glasses than other types of eyewear (and most likely to wear none).

Another example: reading, gender and occupation

Profession	Sex	Preferred reading Scifi	Spy	Total
Politician	Male	15	15	30
	Female	10	15	25
	Total	25	30	55
Administrator	Male	10	30	40
	Female	5	10	15
	Total	15	40	55
Bellydancer	Male	5	5	10
	Female	10	25	35
	Total	15	30	45

Altogether 80 males and 75 females.

Data into SAS

This time there are 3 categorical variables (profession, sex, preferred reading) and a frequency. Arrange with one frequency on each line (without totals):

```
politician male scifi 15
politician male spy 15
politician female scifi 10
politician female spy 15
administrator male scifi 10
administrator male spy 30
administrator female scifi 5
administrator female spy 10
bellydancer male scifi 5
bellydancer male spy 5
bellydancer female scifi 10
bellydancer female spy 25
```


The code

```
data small;
  infile "multiway.dat";
  input profession $ sex $ readtype $ freq;

proc catmod;
  weight freq;
  model profession*sex*readtype=_response_;
  loglin profession sex readtype profession*sex
    profession*readtype sex*readtype
    profession*sex*readtype;
```

Loglin line could have been written `profession|sex|readtype` (include main effects and all interactions between variables), but done this way for a reason.

Assessing what to take out

From the "maximum likelihood analysis of variance":

Maximum Likelihood Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
profession	2	3.46	0.1777
sex	1	0.01	0.9256
readtype	1	7.61	0.0058
profession*sex	2	17.58	0.0002
profession*readtype	2	2.62	0.2691
sex*readtype	1	0.66	0.4168
profession*sex*readtype	2	1.89	0.3894
Likelihood Ratio	0	.	.

- Model fits perfectly (see Likelihood Ratio line)
- As ANOVA, remove 3-way interaction.
- Change `loglin` line to this:

```
loglin profession sex readtype profession*sex
  profession*readtype sex*readtype;
```


Output from this

Maximum Likelihood Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
profession	2	3.58	0.1674
sex	1	0.00	0.9453
readtype	1	13.02	0.0003
profession*sex	2	23.00	<.0001
profession*readtype	2	4.32	0.1155
sex*readtype	1	0.62	0.4321
Likelihood Ratio	2	1.85	0.3969

- Bottom line: "is there evidence of lack of fit?" Answer no: model fits OK.
- Now look at two-way interactions and take out non-significant ones.
- Code for that:

```
loglin profession sex readtype profession*sex;
```

Output

Maximum Likelihood Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
profession	2	2.90	0.2348
sex	1	0.03	0.8686
readtype	1	12.68	0.0004
profession*sex	2	22.79	<.0001
Likelihood Ratio	5	6.56	0.2557

- Model still fits OK (last line).
- Two-way interaction significant: stays.
- Main effects involving profession and sex have to stay.
- Main effect involving reading type significant, so stays.
- Done. Now interpret the estimates.

The maximum likelihood estimates

with missing ones filled in:

Parameter		Estimate	Standard Error	Chi-Square	Pr > ChiSq
profession	administ	0.0526	0.1257	0.18	0.6753
	bellydan	-0.2169	0.1374	2.49	0.1144
sex	female	0.0149	0.0903	0.03	0.8686
readtype	scifi	-0.2989	0.0839	12.68	0.0004
	spy	0.2989			
profession*sex	administ female	-0.5053	0.1257	16.17	<.0001
	bellydan female	0.6114	0.1374	19.82	<.0001
	politician female	-0.1061			
	administ male	0.5053			
	bellydan male	-0.6114			
	politician male	0.1061			

- Readtype: people overall prefer spy novels
- Interaction: bellydancers tend to be female and administrators male (more so than even split of males/females would suggest).

A different way to read the data

- Entering the words into the data file is repetitive. Start with data as laid out in table (in `freq.dat`):

```
15 15
10 15
10 30
5 10
5 5
10 25
```

- Then use "loops" to associate with variables:

```
data myfreq;
  infile "freq.dat";
  do profession="politician   ","administrator","bellydancer";
    do sex="male  ","female";
      do readtype="scifi","spy";
        input freq @@;
        output;
      end;
    end;
  end;
```

- Resulting data set and PROC CATMOD as before.

Simpson's paradox: the airlines example

	Alaska Airlines		America West	
Airport	On time	Delayed	On time	Delayed
Los Angeles	497	62	694	117
Phoenix	221	12	4840	415
San Diego	212	20	383	65
San Francisco	503	102	320	129
Seattle	1841	305	201	61
Total	3274	501	6438	787

- Alaska: 13.3% flights delayed $(501/(3274+501))$.
- America West: 10.9% $(787/(6438+787))$.
- America West more punctual, right?

Percentage delayed by airport

Airport	Alaska	America West
Los Angeles	11.4	14.4
Phoenix	5.2	7.9
San Diego	8.6	14.5
San Francisco	16.9	28.7
Seattle	14.2	23.2
Total	13.3	10.9

- America West better overall, yet *worse at every single airport*!
- Can PROC CATMOD explain?
- 3 categorical variables (airline, airport, on time/delayed), frequency.

Data for SAS

```
losangeles alaska ontime 497
losangeles alaska delayed 62
losangeles aw ontime 694
losangeles aw delayed 117
phoenix alaska ontime 221
phoenix alaska delayed 12
phoenix aw ontime 4840
phoenix aw delayed 415
...
sanfran alaska ontime 503
sanfran alaska delayed 102
sanfran aw ontime 320
sanfran aw delayed 129
seattle alaska ontime 1841
seattle alaska delayed 305
seattle aw ontime 201
seattle aw delayed 61
```


Code

```
data airline;
  infile "airport.dat";
  input airport $ airline $ status $ freq;

proc catmod;
  weight freq;
  model airport*airline*status=_response_;
  loglin airport|airline|status;
```

Or write out all the effects on the `loglin` line.

Alternative form for data

- Data file:

```
497 62 694 117
221 12 4840 415
212 20 383 65
503 102 320 129
1841 305 201 61
```

- Code to read this:

```
data myfreq;
  infile "freq2.dat";
  do airport="losangeles  ","phoenix","sandiego",
    "sanfrancisco","seattle";
    do airline="alaska     ","americawest";
      do status="ontime ","delayed";
        input freq @@;
        output;
      end;
    end;
  end;
```

Output

Maximum Likelihood Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
airport	4	185.99	<.0001
airline	1	118.66	<.0001
airport*airline	4	1138.97	<.0001
status	1	1487.23	<.0001
airport*status	4	99.56	<.0001
airline*status	1	29.09	<.0001
airport*airline*status	4	3.26	0.5156
Likelihood Ratio	0	.	.

- Complicated model fits perfectly (not interesting)
- 3-way interaction non-significant: remove.
- Change loglin line to:

```
loglin airport|airline|status @ 2;
```

(include all interactions $\leq$ 2-way).

Output now

Maximum Likelihood Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
airport	4	231.19	<.0001
airline	1	163.72	<.0001
airport*airline	4	3225.58	<.0001
status	1	2700.13	<.0001
airport*status	4	246.27	<.0001
airline*status	1	41.74	<.0001
Likelihood Ratio	4	3.22	0.5223

- Model fits OK (no evidence of lack of fit).
- All 2-way interactions significant: stop here.

Airline by status, adding missing ones

Analysis of Maximum Likelihood Estimates

Parameter		Estimate	Standard Error	Chi-Square	Prob > ChiSq
....					
airline*status	alaska delayed	-0.1361	0.0211	41.74	<.0001
	alaska ontime	0.1361			
	aw delayed	0.1361			
	aw ontime	-0.1361			

- Alaska *more* likely to be on time and America West *more* likely to be delayed, allowing for effects of other variables.
- This in contrast to overall %'s.
- Other interactions shed some light.

Airport by airline

Analysis of Maximum Likelihood Estimates

Parameter		Estimate	Standard Error	Chi-Square	Prob > ChiSq
....					
airport*airline	losangel alaska	-0.0164	0.0261	0.39	0.5303
	phoenix alaska	-1.4049	0.0302	2165.96	<.0001
	sandiego alaska	-0.1618	0.0348	21.57	<.0001
	sanfran alaska	0.3461	0.0287	145.07	<.0001
	seattle alaska	1.2539			

- America West figures negatives of Alaska figures.
- Frequency less than expected for AA into Phoenix (AA flies less often into Phoenix).
- Frequency more than expected for AA into San Francisco and Seattle (AA flies more often into San Francisco and Seattle).
- Conversely, America West flies more into Phoenix and less into San Francisco and Seattle.

Airport by status

Analysis of Maximum Likelihood Estimates

Parameter		Estimate	Standard Error	Chi-Square	Pr > ChiSq
airport*status	losangel delayed	-0.0335	0.0360	0.87	0.3520
	phoenix delayed	-0.4110	0.0305	181.94	<.0001
	sandiego delayed	-0.0762	0.0487	2.44	0.1180
	sanfran delayed	0.3268	0.0343	90.68	<.0001
	seattle delayed	0.1929			

- On-time estimates negatives of delayed figures.
- Fewer flights to Phoenix are delayed (than to other places).
- More flights to San Francisco and Seattle delayed.

Resolution of this Simpson's paradox

- Alaska Airlines flies mostly into San Francisco and Seattle, while America West flies mostly into Phoenix (airport by airline)
- Flights into Phoenix are more likely to be on time, while flights into San Francisco and Seattle are more likely to be delayed.
- In "overall % late", AA gets penalized for flying into airports where hard to be on time.
- When you allow for who flies where, AA comes out more punctual (as seen in airport-by-airport statistics).

Ovarian cancer: a four-way table

- Retrospective study of ovarian cancer done in 1973.
- Information about 299 women operated on for ovarian cancer 10 years previously.
- Recorded:
  - stage of cancer (early or advanced)
  - type of operation (radical or limited)
  - X-ray treatment received (yes or no)
  - 10-year survival (yes or no)
- Survival looks like response (suggests logistic regression). PROC CATMOD finds any associations at all.

The data

for SAS purposes:

```
early radical no no 10
early radical no yes 41
early radical yes no 17
early radical yes yes 64
early limited no no 1
early limited no yes 13
early limited yes no 3
early limited yes yes 9
advanced radical no no 38
advanced radical no yes 6
advanced radical yes no 64
advanced radical yes yes 11
advanced limited no no 3
advanced limited no yes 1
advanced limited yes no 13
advanced limited yes yes 5
```

Stage, type, x-ray, survival, frequency.

The code

hopefully looking familiar by now:

```
data cancer;
  infile "cancer.dat";
  input stage $ operation $ xray $ survival $ count;

proc catmod;
  weight count;
  model stage*operation*xray*survival=_response_;
  loglin stage|operation|xray|survival;
```


Alternative data entry

- Data like this:

  ```
  10 41 17 64 1 13 3 9
  38 6 64 11 3 1 13 5
  ```

- All values for each stage first. Within each stage, all values for kind of operation; within these, all values for X-ray, then all values for survival:

```
data freq;
  infile "freq3.dat";
  do stage="early    ","advanced";
    do operation="radical","limited";
      do xray="no ","yes";
        do survival="no ","yes";
          input count @@;
          output;
        end;
      end;
    end;
  end;
```


Output #1

Maximum Likelihood Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
operation*xray	1	0.80	0.3712
stage*operation*xray	1	1.33	0.2495
survival	1	0.15	0.6979
stage*survival	1	40.09	<.0001
operation*survival	1	1.69	0.1930
stage*operation*survival	1	0.11	0.7425
xray*survival	1	0.48	0.4871
stage*xray*survival	1	0.87	0.3502
operation*xray*survival	1	0.48	0.4874
stage*operat*xray*surviv	1	0.57	0.4499
Likelihood Ratio	0	.	.

- Four-way interaction and all 3-way interactions not significant: remove all, and check resulting model for fit.
- Change loglin line to this:

  ```
  loglin stage|operation|xray|survival @ 2;
  ```

  that is, keep main effects and interactions up to 2-way.

Output #2

Maximum Likelihood Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
stage	1	0.27	0.6033
operation	1	102.15	<.0001
stage*operation	1	0.59	0.4415
xray	1	10.01	0.0016
stage*xray	1	0.62	0.4324
operation*xray	1	0.01	0.9326
survival	1	0.23	0.6294
stage*survival	1	99.45	<.0001
operation*survival	1	2.06	0.1511
xray*survival	1	0.09	0.7696
Likelihood Ratio	5	7.17	0.2084

- Model still fits all right.
- Only significant 2-way interaction is stage by survival.
- Take out others and check fit again.
- Change loglin line to

  ```
  loglin stage operation xray survival stage*survival;
  ```

Output #3

Maximum Likelihood Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
stage	1	1.50	0.2202
operation	1	110.28	<.0001
xray	1	17.46	<.0001
survival	1	0.55	0.4584
stage*survival	1	100.74	<.0001
Likelihood Ratio	10	10.99	0.3583

- Model fit still OK (no evidence of lack of fit)
- Stage and survival main effects have to stay.
- Operation and X-ray main effects are significant, so they stay.
- Done. Interpret maximum likelihood estimates.

Maximum likelihood estimates

Analysis of Maximum Likelihood Estimates

Parameter		Estimate	Standard Error	Chi-Square	Pr > ChiSq
stage	advanced	-0.0930	0.0759	1.50	0.2202
operation	limited	-0.8271	0.0788	110.28	<.0001
xray	no	-0.2492	0.0596	17.46	<.0001
survival	no	0.0562	0.0759	0.55	0.4584
stage*survival	advanced no	0.7613	0.0759	100.74	<.0001

- Stage by survival interaction: stage of cancer and survival associated. Higher frequency with being in advanced stage and not surviving: advanced stage associated with non-survival.
- Fewer women had the limited operation (more had the radical one)
- Fewer woman had no X-ray treatment (more did have X-ray treatment).
- Interaction with "response" (survival) usually of most interest.

General procedure

- Start with "complete model" including all possible interactions.
- Look at highest-order interaction(s) remaining, remove if non-significant.
- If an interaction significant, keep also everything contained within that interaction. Eg. A*B interaction significant, keep A and B main effects.
- Continue until everything either significant or must be kept.
- Then look at maximum likelihood estimates (can fill in those not shown) and interpret according to whether $+$ or $-$.
- Main effects not usually very interesting.
- Interactions with "response" usually of most interest: show association with response.