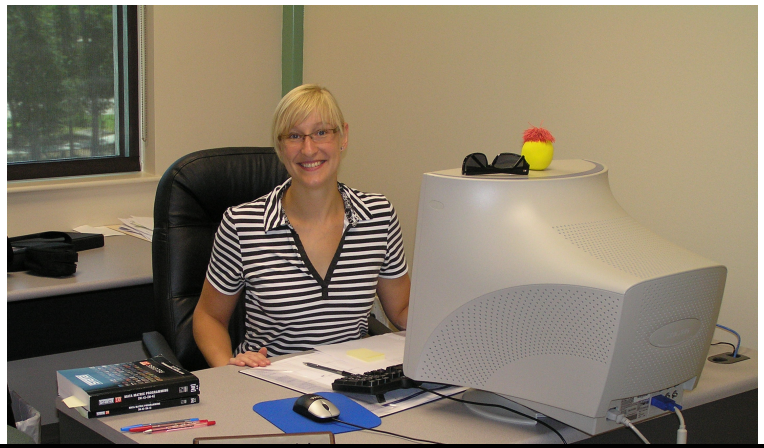
The -biplot- command and software development at StataCorp.

Magdalena Luniak

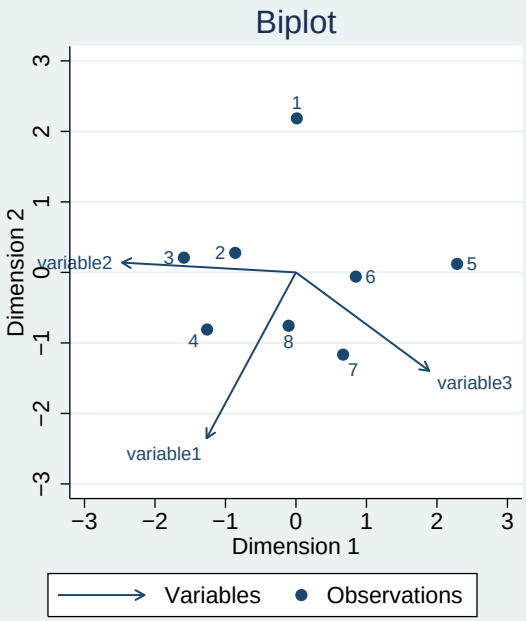
June 26, 2009



Outline

Biplot's properties

- multivariate analysis feature
- graphical two-dimensional representation of dataset:
  - *arrows*: variables
  - *marker symbols*: observations

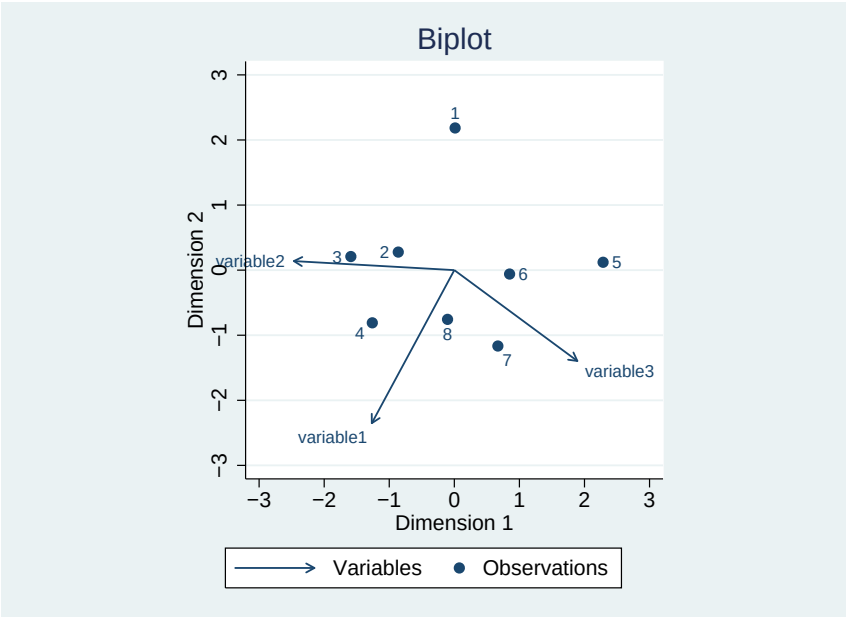



```
Biplot of 8 observations and 3 variables
    Explained variance by component 1   0.6236
    Explained variance by component 2   0.2761
            Total explained variance    0.8997
```

Biplot's properties

Helpfull in understanding the relationship between variables and observations separately and jointly:

- the distance between observations is approximately preserved
- the cosine of the angle between arrows approximates the correlation between variables
- relation of observations to variables

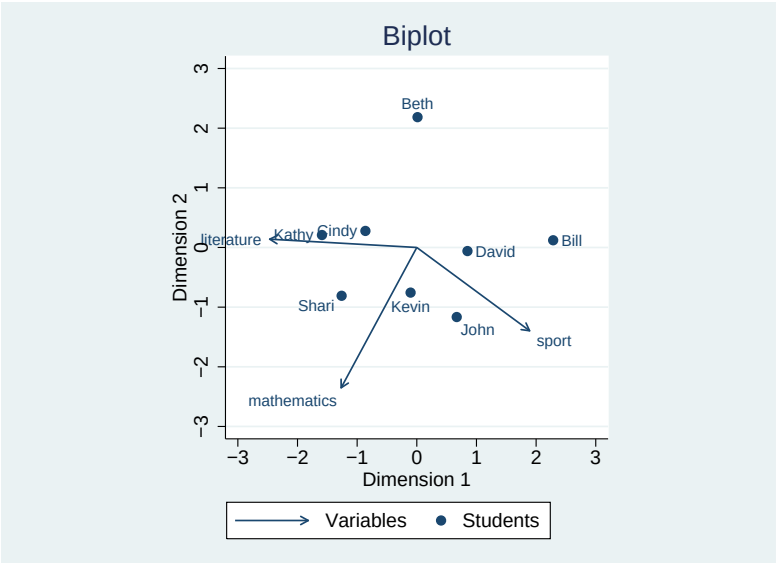


Example: students

Students' scores in mathematics, literature, and sports (1-10)

```
. list
```

	stud	sex	math	literat	sport
1.	Beth	female	1	5	1
2.	Cindy	female	5	10	5
3.	Kathy	female	7	10	2
4.	Shari	female	9	9	4
5.	Bill	male	2	1	10
6.	David	male	4	5	8
7.	John	male	8	3	7
8.	Kevin	male	8	5	5



Methods and formulas

1. Singular value decomposition of the centered data matrix X:
   $X = U_{obs} \times L \times V'_{var}$
   $X = U_{obs} \times L^{\alpha} \times L^{1-\alpha} \times V'_{var}$ for $\alpha \in [0, 1]$
2. Coordinates for Observations and variables:
   $X = G \times H'$
   $G = U_{obs} \times L^{\alpha}$
   $H' = L^{1-\alpha} \times V'_{var}$
3. Coefficient α:
   - columns preserving biplot for $\alpha = 0$
   - rows preserving biplot for $\alpha = 1$
   - symmetric biplot for $\alpha = 0.5$

Outline

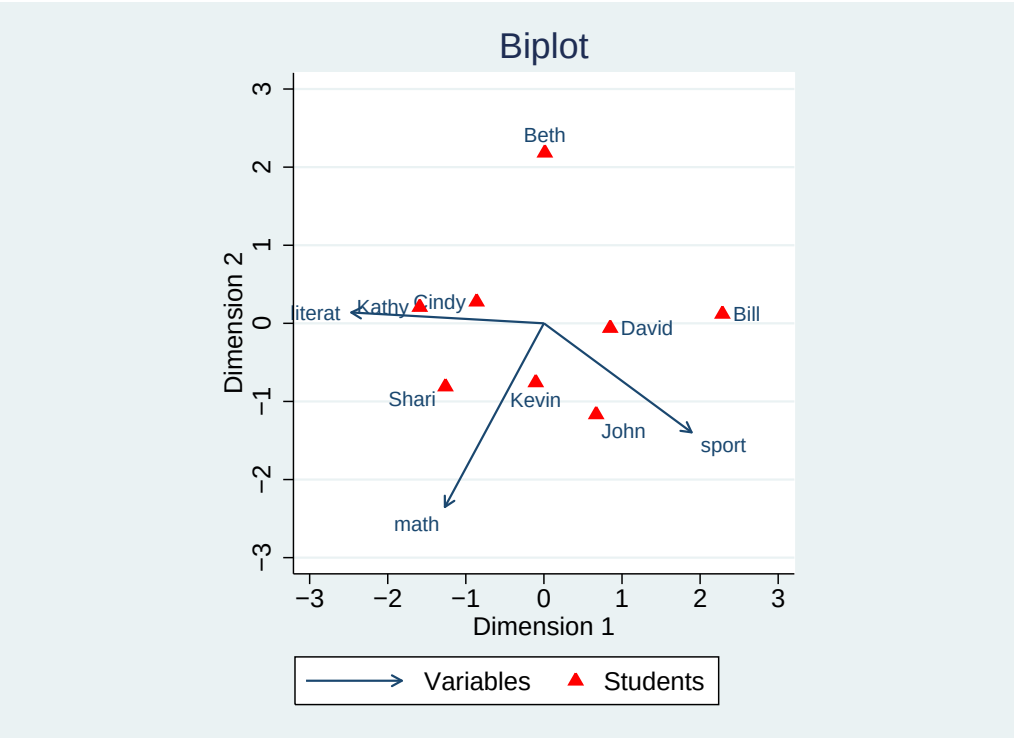
Biplot's syntax

`biplot` *varlist* [*if*] [*in*] [, `options`]

Example of options:

- `rowopts()` affects rendition of observations

```
. biplot math literat sport,
rowopts(mlabel(stud) name(Students) msymbol(T)
mcolor(red))
```



Example of options:

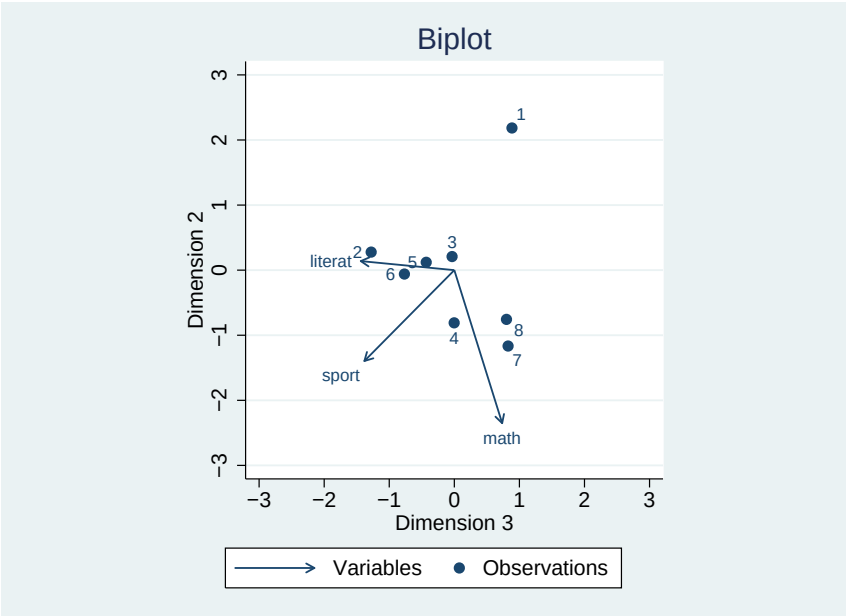
- `dim()` affects dimensions
- `table` displays table showing biplot coordinates

```
. biplot math literat sport, table dim(2 3)
```

Biplot coordinates

Observations	dim3	dim2
1	0.8859	2.1843
2	-1.2777	0.2776
3	-0.0345	0.2088
4	-0.0025	-0.8095
5	-0.4317	0.1207
6	-0.7672	-0.0600
7	0.8262	-1.1657
8	0.8016	-0.7562

Variables	dim3	dim2
math	0.7354	-2.3513
literat	-1.4377	0.1390
sport	-1.3822	-1.3957



Biplot of 8 observations and 3 variables

Explained variance by component 3	0.1003
Explained variance by component 2	0.2761
Total explained variance	0.3764

What will be new in biplot?

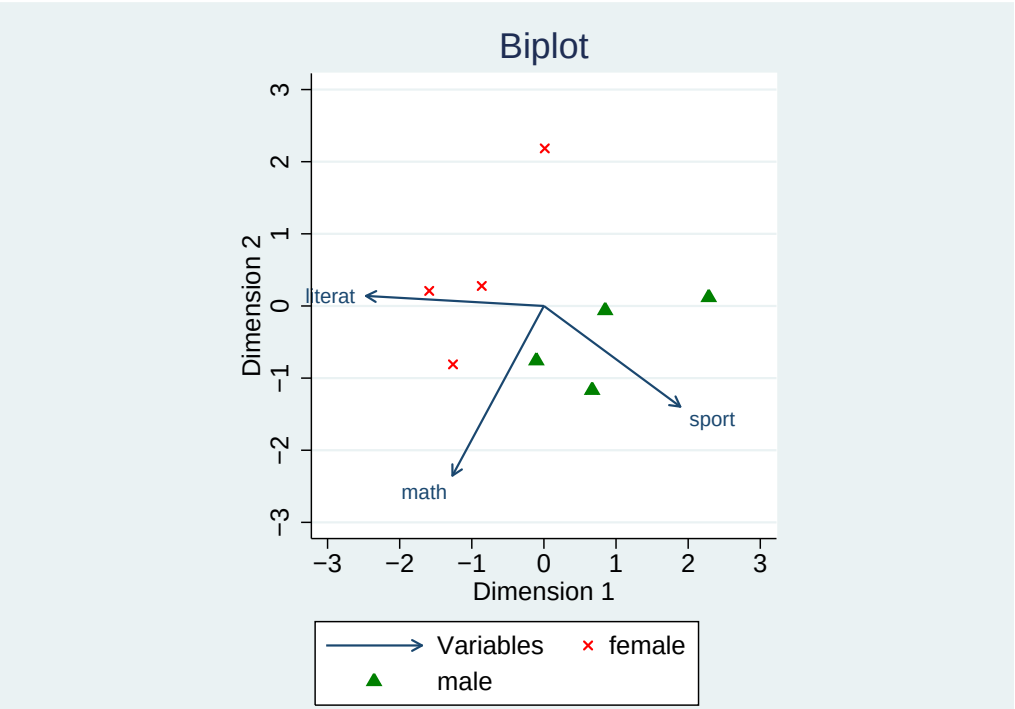
- No limits for number of observations: matrix computations implemented in Mata
- New options:
  - `rowover()` and `row#opts()`
  - `rowlabel`
  - `generate()`

Rows

rowopts(*row_options*)	affect rendition of rows (observations)
row#opts(*row_options*)	affect rendition of rows (observations) in the #th group of *varlist* defined in **rowover()**; available only with **rowover()**
rowlabel(*varname*)	specify label variable for rows (observations)
norow	suppress row points; may not be combined with **rowover()**
generate(*newvar_x newvar_y*)	store biplot coordinates for observations in variables *newvar_x* and *newvar_y*

Example

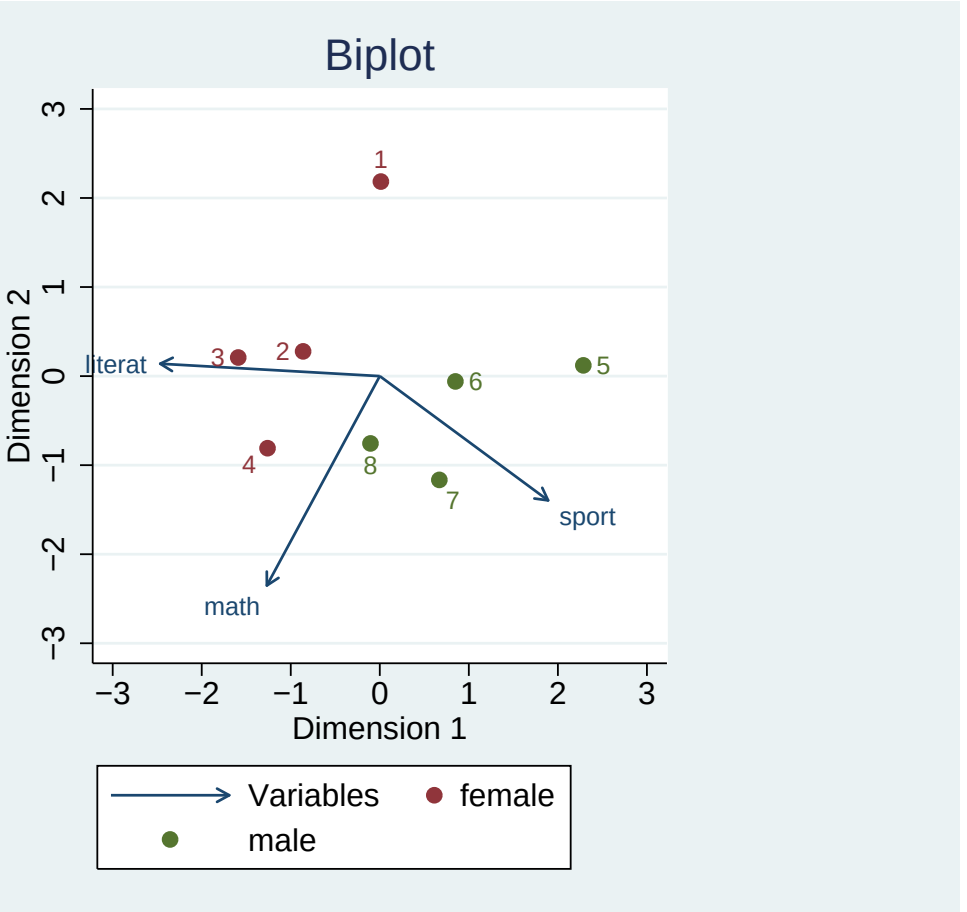
```
. biplot math literat sport,
rowover(sex) norowlabel
row1opts(msymbol(X) mcolor(red))
row2opts(msymbol(T) mcolor(green))
generate(coord1 coord2)
```



generate(coord1 coord2)

```
. list coord1 coord2
```

	coord1	coord2
1.	.0121763	2.184314
2.	-.8614078	.2775752
3.	-1.590056	.2087832
4.	-1.260582	-.8095006
5.	2.28588	.1207452
6.	.8493378	-.0600475
7.	.6690965	-1.165671
8.	-.104445	-.7561985

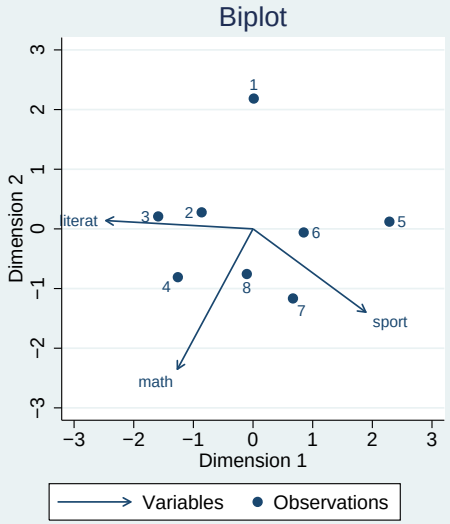


Outline

The story of one option `rowover()`

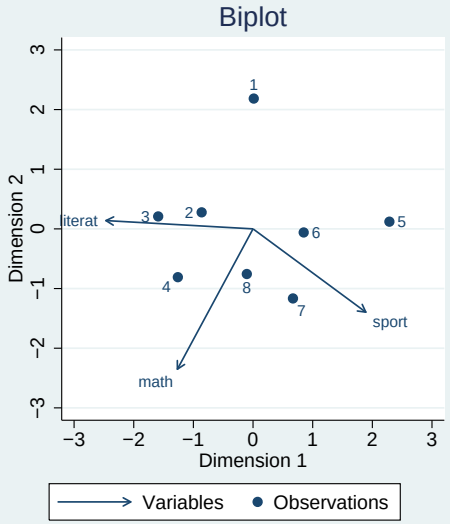
The story of one option `rowover()`

Before

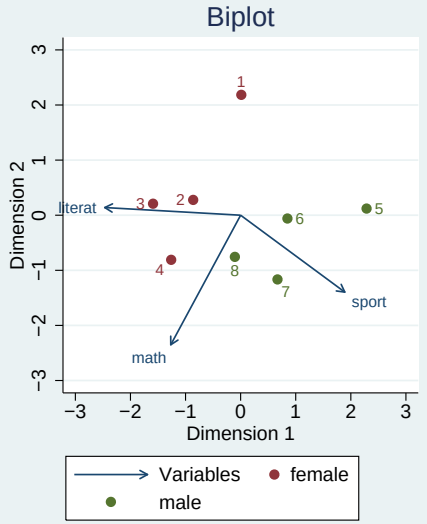


The story of one option rowover()

Before



After



rowover(*varlist*)

identify observations from different groups of *varlist*; may not be combined with **separate** or **norow**

Name ”rowover”

The choice of the new option's name :

- "by"
  - generic
  - in context of graphs means separate graph for every group

Name "rowover"

The choice of the new option's name :

- "by"
  - generic
  - in context of graphs means separate graph for every group
- "over"
  - over different groups
  - overlay graphs
  - still too general

Name "rowover"

The choice of the new option's name :

- "by"
  - generic
  - in context of graphs means separate graph for every group
- "over"
  - over different groups
  - overlay graphs
  - still too general
- "rowover"
  - over different groups in rows
  - overlay graphs
  - concerns rows

Name "rowover"

The choice of the new option's name :

- "by"
  - generic
  - in context of graphs means separate graph for every group
- "over"
  - over different groups
  - overlay graphs
  - still too general
- "rowover"
  - over different groups in rows
  - overlay graphs
  - concerns rows

1st lesson learned

Names used in Stata must suit the whole system of commands

Input

New option has to accept all syntactically correct input.

For example:

- unbalanced parentheses
- unbalanced quotes
- special characters

```
.  biplot math literat sport,
rowover(sex) row1opts(name(‘")"”’))
```

Input

New option has to accept all syntactically correct input.

For example:

- unbalanced parentheses
- unbalanced quotes
- special characters

```
.  biplot math literat sport,
rowover(sex) row1opts(name(‘")"”’))
```

2nd lesson learned

Stata commands must be robust

Certification of the command

New option has to pass certification.
Certification script tests the quality of program:

- covers many possible use cases
- contains the prediction of the correct output
- checks the conformity between obtained and expected results

Example

```
. local expectedResult = 1.233
. biplot math literat sport, generate(coord1 coord2)
. assert ‘expectedResult’ == coord1[1]
```

Certification of the command

New option has to pass certification.
Certification script tests the quality of program:

- covers many possible use cases
- contains the prediction of the correct output
- checks the conformity between obtained and expected results

Example

```
. local expectedResult = 1.233
. biplot math literat sport, generate(coord1 coord2)
. assert ‘expectedResult’ == coord1[1]
```

3rd lesson learned

The quality of the program must be sufficiently tested

Documentation

The new option has to be documented:

- information for users (help files and manuals)
- information for developers (internal documentation)

Description

biplot displays a two-dimensional biplot of a dataset. A biplot simultaneously displays the observations (rows) and the relative positions of the variables (columns). Marker symbols (points) are displayed for observations and arrows are displayed for variables. Observations are projected to two dimensions such that the distance between the observations is approximately preserved. The cosine of the angle between arrows approximates the correlation between the variables.

Options

Main

rowover(*varlist*) distinguishes groups among observations (rows) by highlighting observations on the plot for each group identified by equal values of the variables in *varlist*. By default, the graph contains a legend that consists of group names. **rowover()** may not be combined with **separate** or **norow**.

dim(# #) identifies the dimensions to be displayed. For instance, **dim(3 2)** plots the third dimension (vertically) versus the second dimension (horizontally). The dimension numbers cannot exceed the number of variables. The default is **dim(2 1)**.

Documentation

The new option has to be documented:

- information for users (help files and manuals)
- information for developers (internal documentation)

Description

biplot displays a two-dimensional biplot of a dataset. A biplot simultaneously displays the observations (rows) and the relative positions of the variables (columns). Marker symbols (points) are displayed for observations and arrows are displayed for variables. Observations are projected to two dimensions such that the distance between the observations is approximately preserved. The cosine of the angle between arrows approximates the correlation between the variables.

Options

Main

rowover(*varlist*) distinguishes groups among observations (rows) by highlighting observations on the plot for each group identified by equal values of the variables in *varlist*. By default, the graph contains a legend that consists of group names. **rowover()** may not be combined with **separate** or **norow**.

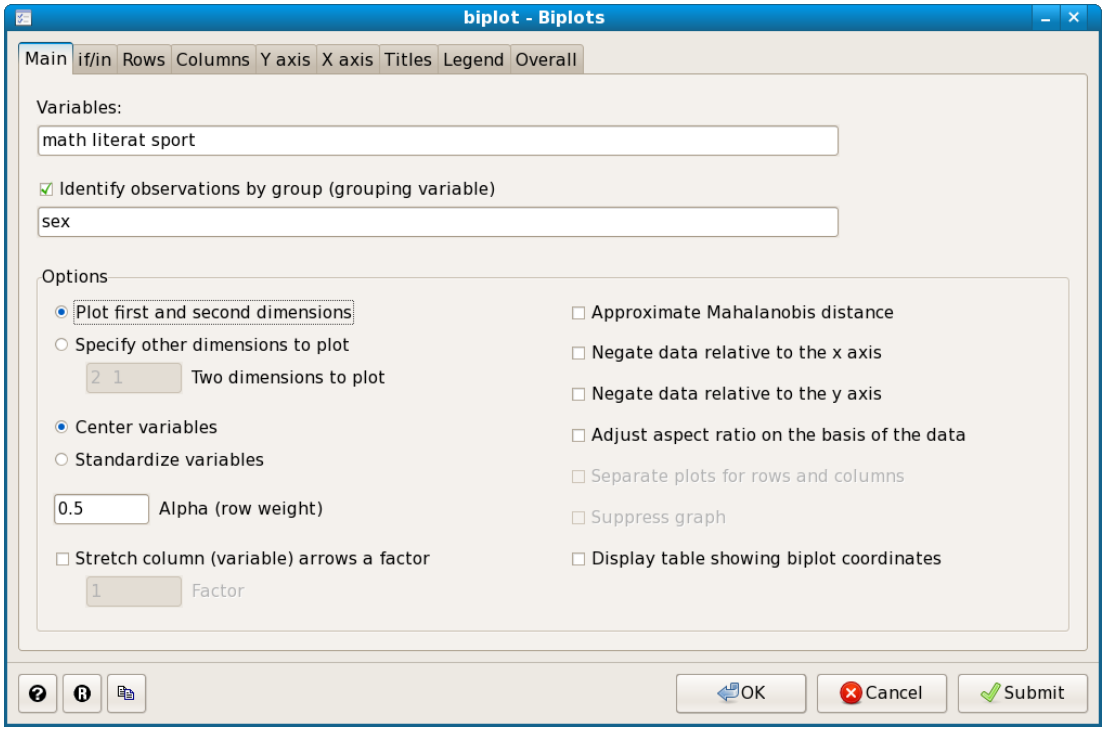
dim(# #) identifies the dimensions to be displayed. For instance, **dim**(3 2) plots the third dimension (vertically) versus the second dimension (horizontally). The dimension numbers cannot exceed the number of variables. The default is **dim**(2 1).

4th lesson learned

Good documentation is as important as a good program

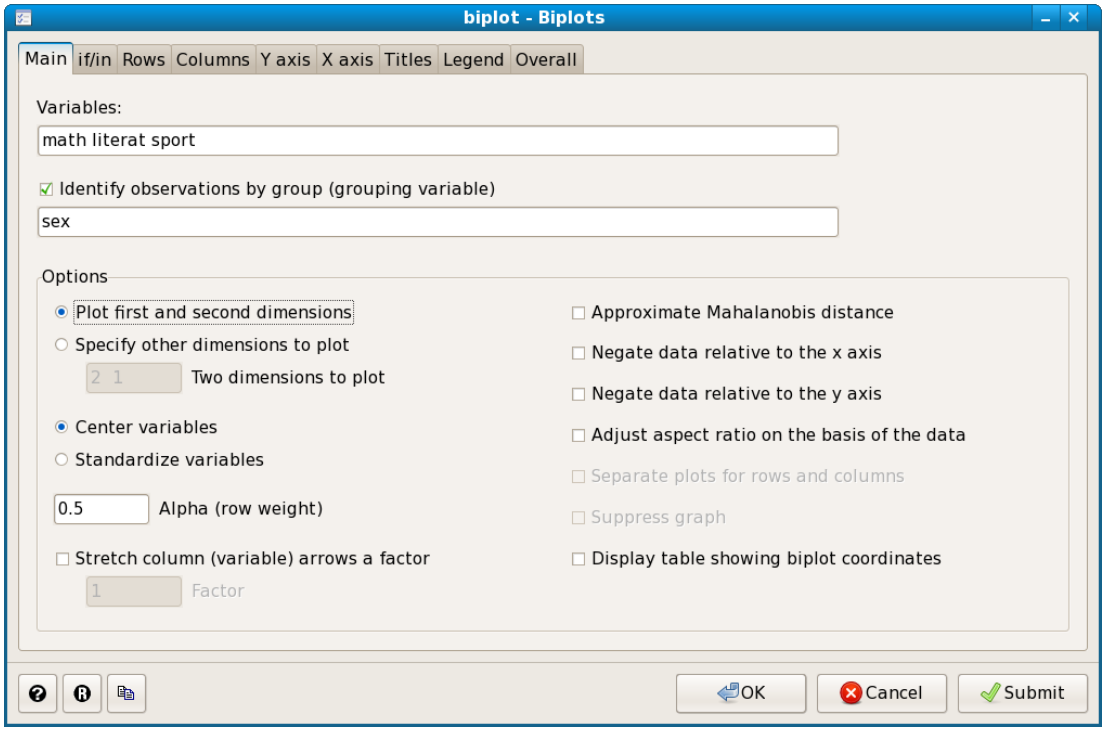
Graphical User Interface

New option has to be added to the dialog box in the proper way.



Graphical User Interface

New option has to be added to the dialog box in the proper way.



5th lesson learned

Do not forget about usability

Certification of the command in system

Certification script tests the quality of program:

- Proof that the command works as a part of the system
- Proof that the command does not have any undesirable side effects on the system
- Proof that the comand works on different platforms and in different editions of Stata

Certification of the command in system

Certification script tests the quality of program:

- Proof that the command works as a part of the system
- Proof that the command does not have any undesirable side effects on the system
- Proof that the comand works on different platforms and in different editions of Stata

6th lesson learned

A command is always a part of the system

Conclusions

Lessons learned:

1. Names used in Stata must suit the whole system of commands
2. Stata commands must be robust
3. The quality of the program must be sufficiently tested
4. Good documentation is as important as a good program
5. Do not forget about usability
6. A command is always a part of the system

Conclusions

Lessons learned:

1. Names used in Stata must suit the whole system of commands
2. Stata commands must be robust
3. The quality of the program must be sufficiently tested
4. Good documentation is as important as a good program
5. Do not forget about usability
6. A command is always a part of the system