



# Statistical analyses of ISHCDB using JMP software

A. Kus, IPP  
for ISH-DB contributors

CWGM 13, Kyoto Univ., Kyoto, 26- 28 Feb 2014



## The JMP software

- Statistical packages and JMP
- Essential features
  
- JMP applications in ISH-DB
  - Statistical studies on confinement database
  - Data processing in profile database
  
- Scripting examples
  - Portions of working scripts
  - Journal files
  - (The full set of scripts to download: [Folder Kus\\_JMPscripts](#))



- JMP Statistical Discovery Software is a comprehensive and interactive statistical package
- There are many software packages enabling statistical analyzes
  - Wikipedia.com lists about 50 different names  
[http://en.wikipedia.org/wiki/Comparison\\_of\\_statistical\\_packages](http://en.wikipedia.org/wiki/Comparison_of_statistical_packages)
  - The Tops could be (alphabetically): MINITAB, **R**, SAS/JMP, S-PLUS, SPSS, STATA, ...
- “The best package” does not exist

## JMP (pronounced “jump”)

- The name “JMP” stands for “John’s Macintosh Project”
- First version created by John Sall (SAS) to use Apple Macintosh’s GUI; Point-and-click interface
- JMP Scripting Language (JSL) is an interpreted language, introduced in 2000 (v4)
- The last software version is number 11 (2012)
- Available on Windows and Mac; both 32-bit and 64-bit (Linux: v5 - v8)



# JMP: Essential features

- **Integrated graphics and dynamic linking:** E.g., having two graphs, one can identify a point in one graph (just klick) and see where that point falls in other graphs (both points are linked with the same rows in the data table)
- **Reports are highly interactive:** Information maybe added and removed after a report is generated
- **Problem-oriented menus:** Each platfform (i.e. analysis module) recognizes the problem (based on the types of variables in the analysis) and creates a relevant menu
- **In a JMP's data table a rich set of metadata may be stored, !!!**  
like scripts, data properties, active and persistent column formulas
- **Scripting** allows to automate and document common analysis tasks and to add user's own statistical analyses
- **A variety of analysis tools** is already available, but there is still enough place and necessity to develop your own scripts
- **Good support:** example datasets and scripts, informative Webpage, user community
- **SASware Ballot for JMP:** an anual opportunity to vote for the new features and services
- **Connection to R:** start R codes from JMP, exchange data between R and JMP

**The full list of all features: <http://www.jmp.com/stat-index/>**



## JMP used for

- Derivation of ISS95 scaling (i.a. H. Yamada, U. Stroth)
- Derivation of ISS04 scaling (i.a. H. Yamada, A. Dinklage, E. Ascasibar)
- Analysis of experimental data and preparation new subsets for ISHCDB (i.a. E. Ascasibar, M. Yokoyama, A. Dinklage, ...)
- Access to W7-AS Oracle database (experiment data)
- Creation of 0D-Ufiles (W7-AS)
- Different studies on confinement database
  - Multicollinearity and principal component analysis (PCA)
  - Cluster analysis
  - Discriminant analysis
  - Regressions under non-standard assumptions (not published)
    - ridge regression and principal component regression (collinearity)
    - SIMEX method (*errors-in-variable* problem)
- **Many tasks could be made more efficient by using JMP scripting**



- Scripting is a way to tell JMP how to do something that would normally be accomplished by way of mouse actions
- Almost any task or analysis that you can perform in JMP can be coded in JSL
- First JMP versions without scripting
- Now, all point-and-click commands are internally converted to an equivalent JSL script

## Two main JSL applications

- Column Formula (usually short script stored within a data table column)  
Purpose: set the value of a column using values of other columns
- Stand-alone scripts (stored in a separate file or in a data table)

## Stand-alone scripts

- Allows storing a sequence of mouse clicks necessary to perform the analysis
- Allows creation of new (larger) applications
- **Have a huge advantage: allow to reproduce and check each single step in the performed analysis, even after years**



Goal: obtain ISS04 scaling.

The procedure

1. Check the possible multicollinearities between variables using PCA
2. Group data
3. Renormalize experimental tauE by ISS95
4. Perform constrained and weighted regression using renormalized tauE
5. Transform regression coefficients into dimensionless variables  
(ISS04 scaling is *dimensionally correct*)
5. Plot experimental tauE vs. ISS04-predicted one



```
Clear Globals(); Clear Log();  
try(currentjournal())<<closeWindow);
```

```
dataset="ISS04.jmp";  
dt=open(dataset);  
dtName=dt<<getName();
```

```
//Add some new columns
```

```
dt<<NewColumn("LOG_PMW",Numeric, Formula(:LOG_P-6));  
dt<<NewColumn("AVE_GROUPING_COMM", Character);  
dt<<NewColumn("GWEIGHT");  
dt<<NewColumn("CFIT_cHB", //collisional high beta constraint  
    Formula(Parameter({a0=0,aa=0,ar=0,an=0,ab=0,ai=0},  
        a0 + aa*LOG_A + ar*LOG_R  
        +(-8*an-5*ab+4*(aa+ar)-5)/3*LOG_PMW  
        +an*LOG_N +ab*LOG_B +ai*LOG_I  
    ))  
);
```





//Define space for analyses

```
cvars={"LOG_TAU","LOG_A","LOG_R","LOG_PMW","LOG_N", "LOG_B","LOG_I"};
```

.....

//Show distributions

```
include("inc_distribution.jsl");
```

//Run PCA to check multicollinearity

```
include("inc_PCA.jsl");
```

//Grouping data

```
groupingColumn="gISS04";
```

```
include("inc_grouping_ISS04.jsl");
```

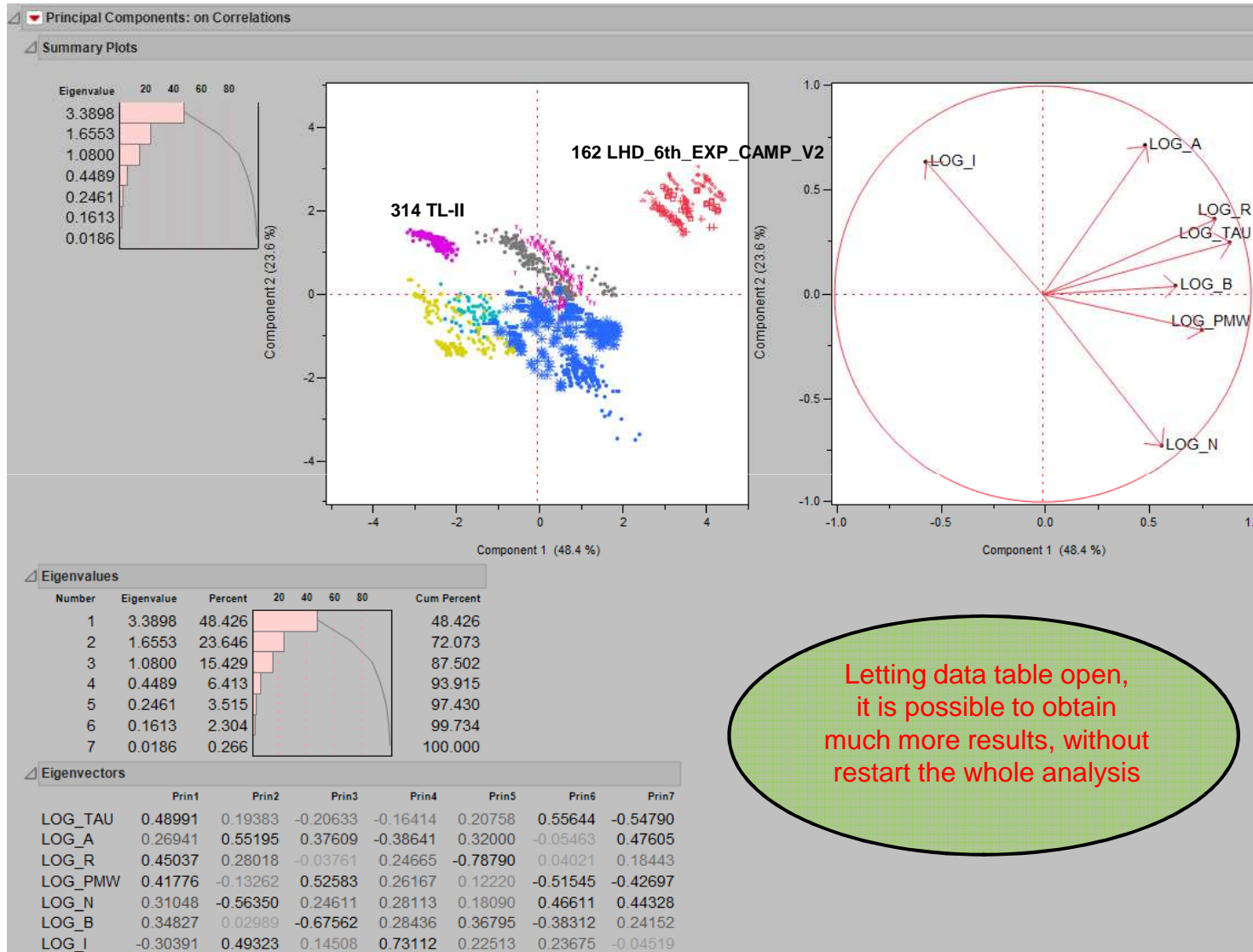
//Run ISS04 fit

```
include("inc_run_ISS04fit.jsl");
```

```
close(dtc,nosave);
```



# Results of the sub-script inc\_PCA.jsl



Summary Plots

- Eigenval. chart
- Score plot
- Loadings

- tau-R-B corr. group
- n-iota not dramatic
- Cond.Index=14

Letting data table open, it is possible to obtain much more results, without restart the whole analysis



//Calculate renorm. factors

```
for(i=1,i<=ndtc,i++,
  if(:AVE_GROUPING_COMM[i]==",
    :RFAC[i]=.,
    :RFAC[i]=10^(:LOG_TAU[i]-column(ReferenceScaling)[i]);
  );
);
```

//Calculate mean and sigma for the renorm. factor

```
Summarize(mm=Mean(:RFAC), sigma=StdDev(:RFAC),
  ggs=by(:AVE_GROUPING_COMM), ndatg=count);
nggs=Nitems(ggs);
```

.....

//Calculate renorm. TAU

```
for(i=1,i<=ndtc,i++,
  :LOGTAU_REN[i] = :LOG_TAU[i]-Log10(:RFAC_AVE[i]);
);
```



.....  
*//Weights*

```
sumndatgs=Sum(ndatg);  
gweights=1/Sqrt(ndatg);
```

*//Set values to needed columns*

```
For(i=1,i<=nggs,i++,  
  For(i2=1,i2<=nrow(),i2++,  
    If(:AVE_GROUPING_COMM[i2]==ggs[i],  
      :GWEIGHT[i2]=gweights[i];  
      :RFAC_AVE[i2]=mm[i];  
      :RFAC_STA[i2]=sigma[i];  
    ));
```

.....  
*//Run the fitting procedure*

```
ft=Nonlinear(Y(:LOGTAU_REN),X(column(cFitCol)),Weight(:GWEIGHT),  
  Reset,Finish,SaveEstimates);  
ft<<journal;
```



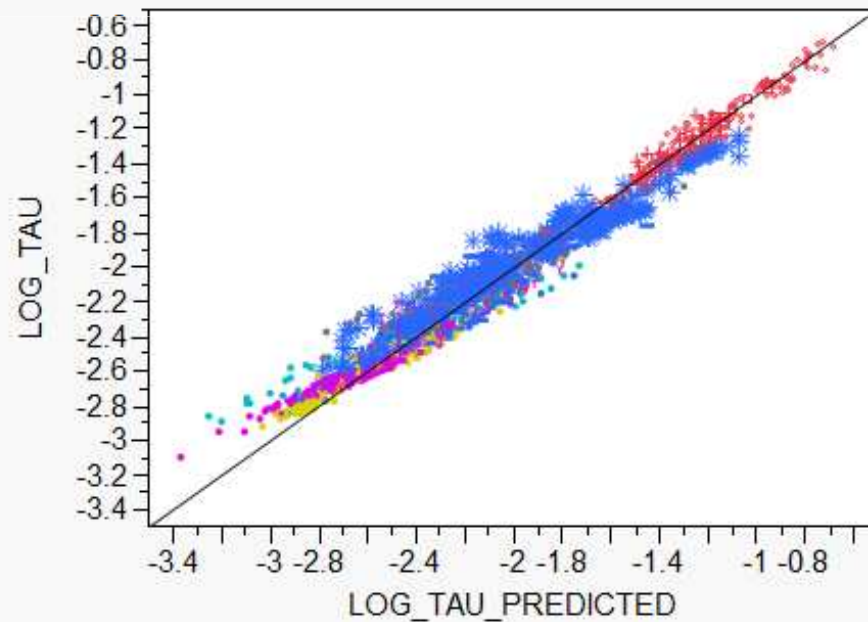
# Results of jslExample.jsl

## Final results of the jslExample script

EngPar	EngPar_val	EngPar_err	DIMLPar	DIMLPar_val
a0	0.13	0.01	arho	-0.79
aa	2.28	0.02	anu	0.00
ar	0.64	0.02	abeta	-0.19
ap	-0.61	.		
an	0.54	0.01		
ab	0.84	0.01		
ai	0.41	0.01		

a0=0.135 RMSE=0.2222

## Overlay Plot



The so called *Outline Box* with regression results.

Overlay Plot is one of many graphics tools available in JMP.



## From the author's experiences

- JMP software has proven itself to be reliable, with many valuable features
- The connection of point-and-click with scripting makes JMP really powerful
- There are many examples available (datasets and scripts)  
<C:\...\Program Files\SAS\JMP\<version>\Samples>
- There are connections to R software package (not tested by the author)
- **Warning:** The possibility to save scripts/formulas in the JMP data table has to be handled with particular caution, as the information is lost when saving data in non-JMP format
  
- Rapid development (**license fees**)
- The licences are not exactly cheap
- As a product of SAS company will probably exist for years

[To try it out for yourself: Dataset and all mentioned scripts are available.](#)