

# Computer Based Statistics: Introduction in SAS

Kerstin Kehrle and Franziska Peter

20.04.2009

## What is SAS?

- SAS stands for **S**tatistical **A**nalysis **S**ystem
- Powerful and flexible tool for many purposes: data management, estimation, optimization, visualization...
- Mixture of 'easy-to-use' procedures and manual programming
- Macro language for recurring code sequences

## A few general concepts of SAS

- Data steps: create data sets, create new variables, subset existing datasets,...
- Procs: analyze your data, estimation, visualization,...
- Macros: often recurring pieces of code can be written as macros, very efficient programming!!!

## The SAS User Interface

### 1. Log window:

- **Blue** is ok (in general)
- **Green** is a warning (should be checked)
- **Red** is an error message (**do not ignore!**)

### 2. Output window: Results are printed in this screen

### 3. Editor window: Write your program commands in here.

## Basic handling of data

To use a SAS-dataset saved on your harddisk or to create a permanent dataset, assign a library pointing to the respective folder:

```
libname name "drive:\ folder";
```

Now, you can find all datasets already in *folder* or newly created datasets in the respective library *name* in your SAS-Explorer.

**Task:** *Create a new folder in your home directory for this course and assign a library to this folder.*

## Working with temporary files vs. permanent files

The name of a SAS dataset consists of two parts:

*library.dataset*

Permanent dataset: choose *name* as your library, if you want to write the dataset permanently into *folder*

Temporary dataset: skip the library or choose *work* as your library to create temporary datasets.

Temporary datasets will be lost when you terminate SAS.

## Getting data into your SAS system

Reading Ascii data into SAS via Data Steps

```
library.dataset;  
Infile 'your_path.dataset';  
Input variable1 variable2 ...;  
Format variable format;  
Label variable='variable label';  
run;
```



## Getting data into your SAS system

**Task:** Read in the following variables from the data set *edata.dat*: *firm* (firm indicator), *date* (*yymmdd6.*), *evntdate* (Event Date), *evntdum* (Negative (1) or positive (2) surprise in earnings), *ret* (daily stock return), *mrktret* (daily market return).

Change the format of the variables *date* and *evntdate* to "date8." and label all the variables.



## Data Step

### Working with Data

```
DATA dataset;  
SET library.dataset;  
variable_new=variable1.+variable2.;  
RUN;
```



## Introducing SAS

---

**Task:** *Assign a library to your folder that contains the sas dataset ABX.sas7bdat. Open the data sets in the SAS-Explorer and have a look at the variables.*

*The data set contains bid and ask quotes of stocks traded on the Toronto stock exchange (\_tsx) and the New York stock exchange (\_ny) as well as bid and ask quotes of the CAD\_USD exchange rate (\_fx).*

*Use a data step to generate a temporary data set and calculate the midquotes  $\frac{bid+ask}{2}$  for both markets as well as the exchange rate midquotes. Further convert the New York midquote into Canadian Dollars.*

## Some procedures (Procs)

- Proc SORT: Sorts the data
- Proc MEANS: calculates mean, median, standard error, confidence limits for the mean,...
- Proc CORR: calculates several correlation coefficients
- Proc FREQ: one-way frequency table, n-way contingency tables, several test statistics

## Example for Proc MEANS

```
Proc MEANS data=dataset;  
var variables;  
by variable;  
output out= outdataset mean=meanvar ...;  
run;
```

**Task:**

*Use the data contained in edata.dat which you have read into SAS in the previous Task. 1. Sort the data set according to firm evntdate date;*

- 2. Calculate the mean of daily returns and market returns for each firm;*
- 3. Calculate the correlation between daily returns and market returns for each firm;*
- 4. Calculate the frequencies of a positive and negative surprise in earnings for each firm;*

## Macros Example

```
%macro macroname(dataset, variable1, variable2);  
&dataset. ;  
set library.&dataset. ;  
variable_new=&variable1.+&variable2.;  
run;  
%mend macroname;
```

*A macro can be called with*

```
%macroname(inputdataset, inputvar1, inputvar2);
```



## Introducing SAS

---

**Task:** *Assign a library to your folder that contains the sas datasets ABX.sas7bdat and ABY.sas7bdat. Open the data sets in the SAS-Explorer.*

*The data set contains bid and ask quotes of stocks traded on the Toronto stock exchange (\_tsx) and the New York stock exchange (\_ny) as well as bid and ask quotes of the CAD\_USD exchange rate (\_fx).*

*Write a macro that uses a data step to calculate the midquotes  $\frac{bid+ask}{2}$  for both markets as well as the exchange rate midquotes. Further convert the New York midquote into Canadian Dollars.*

*Call the macro for both stocks (ABX and ABY).*



**Additional Task:** *Compute a macro that calculates and adds the spread (ask-bid) to the existing data set and also returns another temporary data set which contains the average spread. Keep the variables, the library and the input data set flexible. Call the macro using one of the data sets given above.*



## Creating a table with SQL code

```
Proc SQL;  
create table name as select  
variables  
from dataset;  
quit;
```

## Merging two tables with SQL code

```
Proc SQL;  
create table name as select  
a.variables  
,b.variables  
from dataset1 a dataset2 b  
on a.idvar1=b.idvar2;  
quit;
```

## Processing data

**Task:** *Create an index for the different firms in the data set.*

*1.Step: Produce table with distinct firm numbers;*

*2.Step: Use a data step to create an index number 1, 2, ... for each firm;*

*3.Step: Merge back the index number to the original dataset and write it permanently in the destination folder;*