

# SAS (Statistical – Analysis – System)

System zur Verwaltung, Analyse und Visualisierung von Daten.  
Verfügbar auf nahezu allen gängigen Betriebssystemen  
(MS-DOS, Windows, OS/2, UNIX, ....). Die derzeit auf den Rechnern  
des Rechenzentrums verfügbare Version ist 9.2.



## Wie starte ich SAS ?

Das Programmpaket SAS befindet sich installiert auf einem Rechner des Rechenzentrums (RZ) der Universität. Somit ist es für alle Kursteilnehmer notwendig, neben einem gültigen Account auf den Rechnern der Fakultät (Thales, Turing,.....) ebenfalls einen gültigen Account für die Rechner des Rechenzentrums zu besitzen (Antrag stellen usw.), um auf das Programmpaket SAS zugreifen zu können. Im Folgenden wird dies vorausgesetzt.

Der Rechner des RZ, auf dem SAS verfügbar ist, hat den Namen andromeda. Somit muss zuerst eine Verbindung vom Rechner der Fakultät zur andromeda hergestellt werden. Im Folgenden sehen Sie den Ablauf für jeden Start einer SAS-Sitzung von einem Rechner (z.B. der Turing) unserer Fakultät:

1. Anmelden auf dem Fakultätsrechner
2. Verbindung zu RZ-Rechner andromeda herstellen:  
`turing$ ssh -XC -l [username] login.rz.uni-ulm.de`
3. Eingabe des geforderten Passwortes
4. Bereitstellen des SAS-Programmpaketes:  
`andromeda$ option sas92`
5. Aufruf von SAS:  
`andromeda$ sas &`

Danach öffnen sich auf Ihrem Monitor mehrere Fenster, u.a. der SAS-Programm-Editor, das SAS-LOG-Fenster und das SAS-Output-Fenster. Zusätzlich erscheint die SAS-Toolbox.

Beim erstmaligen Aufruf wird von "option sas" ein persönliches SAS-Verzeichnis angelegt, das dann unter der Umgebungsvariable \$\$SASHOME zugänglich ist. Ein Wechsel in dieses Verzeichnis kann also z.B. durch "cd \$\$SASHOME" erfolgen.

## Einige Bemerkungen zu SAS:

1. SAS ist ein Software-System bestehend aus verschiedenen Modulen, die je nach Art der Anwendung und des Problems vom Benutzer aufgerufen werden können. Beispiele hierfür sind:

- SAS/BASE (Daten- und Dateimanagement, einfache Analysen->Grundbaustein)
- SAS/STATISTICS (Komplexe statistische Analysen)
- SAS/GRAPH (Komplexe Graphiken)
- SAS/FSP (Dateneingabe- und Verwaltungsprogramm)
- SAS/ASSIST (komplette Menüsteuerung)
- SAS/QC (Methoden zur Qualitätskontrolle)

.  
. .  
.

## 2. Betriebsarten von SAS

- **Batch Betrieb:** Hier wird mittels eines beliebigen Texteditors eine SAS-Programmdatei erstellt und diese nach Abspeicherung mittels SAS kompiliert. (Bsp.: *sas programm.sas*). Als Resultat (sofern erfolgreich) erhält man die Ergebnisse in den Dateien *programm.out* (oder *.lis*, je nach System) und *programm.log*.

.out : Output des Programms

.log : Log-Datei, enthält Kommentare und Fehlermeldungen (dies ist sehr wichtig, falls ein SAS-Programm nicht richtig ausgeführt wurde).

Dieser Betrieb ist häufig für die Analyse von sehr großen Datenmengen sinnvoll.

- **Interaktiv:** (Fenster-Modus, Display-Manager-Modus) Dieser Modus wird von uns ausschließlich benutzt, da er sich durch eine gewisse Benutzerfreundlichkeit auszeichnet. Hier erfolgt die Eingabe der SAS-Programme durch das Programm-Editor – Fenster. Da auch Log- und Output-Fenster zur Verfügung stehen, wird das Handling sehr vereinfacht.

Die meisten SAS-Optionen lassen sich hier auch mittels der Fensterleiste leicht verändern.

## Der allgemeine Aufbau eines SAS-Programms

Unter einem SAS-Programm versteht man eine Sammlung von SAS-Befehlen. Jedes SAS-Programm lässt sich in sog. DATA- und PROC-Steps einteilen. Ein Step besteht normalerweise aus mehreren Befehlen (Statements). Ein Step beginnt mit einem DATA- oder PROC- Statement und endet beim nächsten DATA-, PROC-, RUN-Statement. Ein Programm wird dann mittels Klick auf "Run-->Submit" ausgeführt. Der Quelltext, der dann "verschwindet", kann mit "Run-->Recall Last Submit" wieder in das Editorfenster geholt werden. Ein Programmbeispiel folgt auf der nächsten Seite.

## Der DATA- und PROC- Step

Im DATA-Step wird grundsätzlich eine SAS-Datei erstellt. Diese kann temporär (nur für die Dauer der Sitzung, danach gelöscht) oder permanent erzeugt werden. Soll diese permanent abgespeichert werden, so muß bei der Angabe des Dateinamens ein sog. *Libref* (Verweis auf ein Verzeichnis) mit angegeben werden. Solche Librefs lassen sich mit Hilfe des LIBNAME-Statements definieren.

Bei der Wahl der Namen und Dateien oder Variablen müssen folgende Einschränkungen beachtet werden.

Gültige **SAS-Namen** dürfen

- nicht länger als 32 Zeichen sein
- nur mit einem Buchstaben oder \_ beginnen, nach dem ersten Zeichen aber auch Zahlen enthalten
- keine Leerzeichen enthalten
- keine Umlaute und Sonderzeichen (wie \$,%,&/, \,;,.,;) enthalten außer \_

(Bsp. für gültige SAS-Namen: datei1, neudat, var\_1 ....)

Regeln für **SAS-Statements:**

- SAS- Statements enden mit Semikolon
- SAS-Statements unterscheiden nicht zwischen Groß- und Kleinschreibung
- beliebig viele SAS-Statements können in eine Zeile geschrieben werden
- ein SAS-Statement kann von einer Zeile in die nächste reichen, solange kein Wort getrennt wird
- SAS-Statements können in beliebigen Spalten des Editors beginnen
- Worte in SAS-Statements werden durch Leer- oder Sonderzeichen getrennt (Bsp.: =, - ,+...)
- Kommentare sind in /\* ... \*/ eingeschlossen (Bsp.: /\* Dies ist ein Kommentar \*/)

Beispiel eines SAS-Programms:

(Hierbei wird vorausgesetzt, dass im Verzeichnis mit dem Libref lib eine Datei namens „beispiel“ existiert. Diese enthalte mindestens die Variablen Name, Alter, Größe und Gewicht)

<pre>LIBNAME lib '\$SASHOME'; DATA eins;     SET lib.beispiel;     KEEP name alter groesse gewicht; RUN;  PROC PRINT DATA=eins;     VAR name alter; RUN;  PROC MEANS DATA=eins;     VAR alter gewicht; RUN;</pre>	<p>DATA – Step</p> <p>PROC – Step</p> <p>PROC – Step</p>	<p>Hier wird eine temporäre Datei mit dem Namen `eins` erstellt, die die Variablen name, alter, groesse und gewicht aus der vorhandenen SAS-Datei beispiel enthält, die in \$SASHOME liegt.</p> <p>Die Prozedur PRINT wird aufgerufen und die Daten von name und alter aus der Datei `eins` ausgegeben.</p> <p>Die Prozedur MEANS wird mit den Daten aus der Datei `eins` aufgerufen. Diese berechnet Größen wie den Mittelwert und Standardabweichung der Variablen alter und gewicht.</p>
---	--	---

**Wie man erkennt, müssen Zeilen mit einem oder mehreren Schlüsselwörtern jeweils mit einem Semikolon abgeschlossen werden** (SAS-Befehle sind nur zu besseren Kennzeichnung groß geschrieben !)

## Die Fenster von SAS

- Session Management:** Hier kann SAS beendet werden (ansonsten am besten minimieren o. ä.)
- Results:** Hier werden alle Ausgaben aufgelistet. Praktisch: über das Menü: *Edit -> Clear all* lässt sich alles löschen – dann weiß man, welche Ergebnisse neu dazukommen.
- ToolBox:** Für Anfänger eher uninteressant, kann man ohne Bedenken schließen
- Explorer:** ebenso
- Output:** klar, kann im Results-Fenster geleert werden
- Graph1:** wie Output, nur für graphische Darstellung
- Log:** die Log-Ausgaben, ggf. mit Fehlermeldungen. Kann über den Kommandozeilen-Modus mit „clear log“ (oder über das Menü) gelöscht werden, dann weiß man, welche Meldungen neu dazukommen
- Program Editor:** der Standardeditor, mit Syntax-Highlighting, dafür verschwindet der Text beim Submit
- Notepad:** ein alternativer Editor, bei dem der Text beim Submit nicht verschwindet (evtl. etwas einfacher zu bedienen), dafür ohne Syntax-Highlighting. Aufruf über Menü: Tools->Text Editor  
Zeilennummern (nötig, um Zeilen einzufügen und zu löschen) können mit Strg+Y *nums on* aktiviert werden.

Alle Fenster können wieder über das Menü geöffnet werden, falls sie versehentlich geschlossen wurden.

## Wichtige SAS-Befehle im Command-Line-Modus

Wird SAS wie oben beschrieben gestartet, so erhält man ein Desktop mit mehreren Fenstern, die jeweils mit einer Menüleiste versehen sind. Mittels dieser Menüleiste lassen sich alle Funktionen von SAS steuern. Nur das Programm-Editor-Fenster enthält einen Menüpunkt ‚Run‘ mit dem die Ausführung von SAS-Programmen vorgenommen wird.

Diese Menüleiste, die für viele Arbeitsschritte sehr übersichtlich und komfortabel ist, kann aber auch sehr hinderlich sein, wenn man beim Verändern eines Programmes schnell kompilieren und dann evtl. am Programm weiterarbeiten möchte (da man ständig mit der Maus zurück in das Programm-Editor-Fenster wechseln muss, um dann dort das ‚abgeschickte‘ Programm wieder einzulesen). Somit kann es hier von Vorteil sein, in den Command-Line-Modus zu wechseln.

Umstellen: In der Menüleiste auf ‚Tools/Options/Turn Command Line on‘ klicken, oder einfach „Strg+Y“.  
Ergebnis: Die Menüleiste verschwindet und es taucht eine Kommandozeile auf (Command ==> \_\_\_).

Hier lassen sich schnell und einfach die wichtigsten Befehle eingeben.

Wichtige Befehle:

1. **Submit** Programm ausführen (Alternative: F3)
2. **Top** auf die erste Seite des aktuellen Windows springen
3. **Zoom** aktuelles Fenster vergrößern oder verkleinern
4. **Help** ins Hilfe Menü wechseln – sinnvolle Alternative: Im Browser folgende Hilfeseiten aufrufen:  
Übersicht PDF-Hilfsdateien:  
<http://kiz.uni-ulm.de/staff/software/doku/application/sasdoc/saspdf/common/mainpdf.htm>  
PROC in BASE: <http://kiz.uni-ulm.de/staff/software/doku/application/sasdoc/saspdf/proc/pdfidx.htm>  
STAT-Paket: <http://kiz.uni-ulm.de/staff/software/doku/application/sasdoc/saspdf/stat/pdfidx.htm>  
GRAPH-Paket: <http://kiz.uni-ulm.de/staff/software/doku/application/sasdoc/saspdf/gref/pdfidx.htm>
5. **Log** ins Log-Window wechseln
6. **Out** ins Output-Window wechseln
7. **PGM** ins Editor-Window wechseln
8. **Recall** (im Editor-Window) zuletzt ausgeführtes Programm zurückholen (Alternative: F4)
9. **File** mit Angabe des Dateinamens, abspeichern des aktuellen Programms  
(Bsp.: file 'sasuser/test.sas')
10. **Lib** ins Lib-Window wechseln (hier werden sämtliche librefs angezeigt)
11. **Keys** ins Keys-Window wechseln (hier lassen sich Tasten mit Befehlen belegen)
12. **Command** Zurückwechseln in den Menüleisten-Modus (komfortabler zum Abspeichern, Drucken oder Einladen von Dateien) (Alternative: Strg + Y)
13. **Clear ...** Gespeicherte Werte löschen, z. B. im Log oder der Ausgabe. => clear log
14. **Nums on** Zeilennummern einschalten
15. **Bye** SAS verlassen (alle temporären Dateien gehen verloren !)

## Wichtige Tastaturkürzel

Der allerwichtigste

Strg + X Überschreibe/Einfüge-Modus wechseln

### Ausführung

F3 Submit  
F4 Recall

### Kopieren / Einfügen

Strg + F Mark (beginnt markierten Bereich)  
Strg + K Cut (Bereich ausschneiden)  
Strg + R Store (Kopieren)  
Strg + T Einfügen

### Zeilen löschen / einfügen

Um Zeilen einzufügen, gehen Sie mit dem Cursor in den Bereich mit den Zeilennummern, und geben „i“ ein, bzw. z. B. „i5“, um 5 Zeilen nach der aktuellen einzufügen.

Um Zeilen zu löschen, kann man gleich vorgehen, nur statt „i“ ist „d“ einzugeben.

### Weitere

Strg + Y In Kommandomodus wechseln / Kommandomodus verlassen

Weitere Kürzel gibt es bei Tools->Options->Keys

Falls irgendwas nicht funktioniert:

# Don't panic !