

SAS Guide
Lab Guidelines for Econ Students

1. How to Logon and start SAS:

Step 1: Logon:

SAS is available in **JC101** on computers **PC01, PC05, PC09, PC10, and PC19** inclusive. On any of those machines, Press “Ctrl-Alt-Delete” keys and fill-in the requested User ID and Passwords using both the keyboard and mouse. If you don't have a computer account at Clark yet you need to contact ITS to get one.

Step 2: Starting SAS:

Click the “Start” button (on the bottom left of the screen). From the list, select the “All Programs” icon. Then click on the “The SAS System” icon, finally select the SAS “triangle” icon and wait for the SAS screen to show-up.

2. How to Run the SAS Program.

Step 1: Entering the “program editor”

When SAS starts up, 4 different windows will open up (one of which is hidden). Most of your work will be done in the windows on the right. To move between the three windows on the right use the tabs on the bottom of the screen (entitled: “output,” “log,” and “editor.”).

Step 2: Creating a new SAS “program” or opening an already existing one.

Select the “editor” window, select the “file” menu item and choose “open” to open a file already saved on your disk (this is the way that you will begin all of your assignments for Econ 273). If you feel adventurous, you can write a new SAS file using the “editor” by choosing “new” in the “file” menu item (Be sure to save it!!!). When you are editing a program, maximize the window to give yourself more room (by clicking on the “window” icon, top-right-center icon, of the “editor” window).

Step 3: To Run A Program.

Select the “run” menu item and choose “submit” to run your error-free SAS program. An alternative is to click on the icon of a “running boy” on the toolbar.

Once you have run your program, select the “log” tab at the bottom of the screen. You can scroll down this to see if there are any errors (errors will be in red) in your program. After you have fixed these errors, you can view and save your “output” file by selecting the “output” tab at the bottom of the screen.

Each time that you run your program the “log” and “output” information will be appended to those files. It can be helpful to *clear* these files to ensure that you are reading the correct “log” and “output” information (the information that pertains to the latest program that you submitted). You can do this by selecting the “edit” menu item and choosing “clear all” (you will have to do this, individually, for each screen that you want to clear).

A Sample Program

```

/*****
/* Read in data, run regression and keep residuals plots and compute */
/* correlations, and means */
*****/

/* print output title */

title 'Regression Analysis of Money Dataset';

/* define dataset to be used */

data mondat;

/* read in the dataset (in ACSII code) from A: drive of your computer */
/* Note: "mon1.dat" read in below is the same as mon.dat, but the */
/* explanatory header information has been removed, so that */
/* all that remains is columns of data. All of the datasets which you */
/* use MUST have this same basic structure. */

infile 'a:\mon1.dat';

/* specify the variable names in each columns of your dataset */

input m1 m2 int ip prices;

/* Transform variables into logs and log differences, creating new variables. */
/* You could do this for all of your variables!!! */

lm1      =      log(m1);
dlm1     =      lm1-lag(lm1);
lm2      =      log(m2);
dlm2     =      lm2-lag(lm2);
lip      =      log(ip);
dlip     =      lip-lag(lip);

d0_m1    =      m1-lag(m1);
d1_m1    =      lag(d0_m1);
d0_m2    =      m2-lag(m2);
d0_ip    =      ip-lag(ip);
d0_pr    =      prices-lag(prices);

/* create a "time" variable, which is nothing more than a sequence of */
/* equally spaced integer values, here time = 1,2,3,4,5,6,7, ... */

time = _N_;

/* compute means of some variables */

proc means;
var m1 m2 int ip prices;
run;

proc means;
var dlm1 dlm2 int dlip d0_pr;
run;

proc means;
var dlm1 dlm2;
run;
```

```

/* compute correlation coefficients */

proc corr data=mondatt;
var d0_m1 d0_m2 int d0_ip d0_pr;
run;

/* plot the log of the money variables against each other */

proc plot;
plot lm1*lm2;
run;

/* plot log money against time */
proc plot;
plot lm1*time;
run;

/* a fancier plot procedure which allows plots of more than one
series. This plot will open in a separate window. */

symbol1 i=spline v=dot;
proc gplot;
plot m1*time ip*time /overlay;
run;

/* Run a regression (least squares), note that intercept is automatically */
/* included the output of this regression should include t-statistics */
/* coefficient estimates, etc. save the residual series to newdata */
/* and print it to the output file */

proc reg;
model dlm1=dlm2;
output out=newdata r=resids;
run;

proc print data=newdata;
var resids;
run;

/* further commands which will be useful for future work */

proc reg;
model dlm1 = dlip / dw;
output out=resdat r=uhat;

data newdat; set newdat resdat;
t= _N_;
uhat_l1=lag1(uhat);
uhat_l2=lag2(uhat);

proc reg data=newdat;
model uhat = uhat_l1 uhat_l2 / noint;

proc plot data=newdat;
plot uhat*t;

```

```
title 'plot of uhat over time';
```

```
/* end of program */
```