AARHUS UNIVERSITET
BUSINESS AND SOCIAL SCIENCES

SAS IML beyond basics

SAS IML for use in Empirical Finance

See more at www.asb.dk/ag

Analytics Group

Agenda

- Recap from the basic course
- Some matrix operations
- Loops and Double loops
- If sentences
- Critical values, p-values, T-tests
- Generating random numbers

SAS IML beyond basics

- The course today will focus on making you cable of solving the first set of assignments in Empirical Finance.

Recab from basic course

- Remember the following matrix operations:

Matrix multiplication: A*B
Element-wise multiplication: A#B
Transpose of a matrix: A` og t(A)
Horizontal concatenation: A||B
Vertical concatenation: A//B
Matrix exponation: A**2
Element-wise addition (subtraction): A+(-)B
Inverse of a matrix: inv(A)

Lib-reference

- Remember the importantness of defining your library! (If you need to use SAS-data files or need to export your data to e.g. Excel)

```
Libname YOUR_LIBNAME ”PATH”;
Run;
```

Ex.

```
Libname ASB ”C:\Users\rusa\Desktop\EF”;
Run;
```

Import an Excel-file

- The file SP500 contains approximately 10 years of daily log returns.
- The first colomn contains the log return of a market index (MSCI_WORLD), whereas the remaining colomns are126 different stock-returns from the S&P500

AARHUS UNIVERSITET
BUSINESS AND SOCIAL SCIENCES

Import an Excel-file

- The proc import code for .xls file

```
proc import out= ASB.SP500
datafile= "C:\Users\rusa\Desktop\EF\SP500.xls"
dbms = xls replace;
getnames=yes;
run;
```

Convert SAS file to a matrix

```
Proc iml;
Use asb.sp500;
Read all var _num_ into return;
Quit;
```

Average of a time series

- How do we find the average log return of the 3rd stock for the last year? (assume 251 trading days in a year = 250 returns)
- Step 1: Define the relevant vector
- Step 2: Find the sum of this vector
- Step 3: Divide it by the number of observations

Average of a time series

- Step 1: Define the relevant vector (remember that we have already defined the matrix return)

```
N=nrow(return);
Vector=return[N-249:N,4];
```

Average of a time series

- Step 2: Find the sum of this vector

By using simple Matrix-algebra:

```
T=nrow(vector);
Iota=J(T,1,1);
SumV=Iota``*Vector;
```

Average of a time series

- Step 3: Divide it by the number of observations

```
AverageV=SumV / T;
Print AverageV;
```

AverageV = 0.0002493

Assignment 1

- What was the average return for all the independent stocks at T=1152 ? (Not the market!) Name it: AverageR (result: -0.011058)

- Multiply this average number with the equivalent average from T=360 Name it: AverageT (result: 0.0000367)

- For the 7th stock return, add the return on the 100th day to AverageT and name it answer (result: 0.0193124)

Loops

Remember that loops runs the same code over an over again:

```
Proc iml;
A=(1:10)`;
B=J(10,1,0);

Do i=1 to 10;
        B[i,1]=A[i,1]*(i+i);
End;

C=A||B;
Print C;

Quit;
```

C	
1	2
2	8
3	18
4	32
5	50
6	72
7	98
8	128
9	162
10	200

AARHUS UNIVERSITET
BUSINESS AND SOCIAL SCIENCES

Loops

Estimating the rolling beta for 3M: Assume 252 trading days per year.

```
Proc iml;
Use ASB.SP500;
Read all var _num_ into return;

N=ncol(return);
T=nrow(return);
Beta=J(T-250,1,0);
Do i=1 to T-250;
        X=J(251,1,1)||return[i:250+i,1];
        Y=return[i:250+i,2];
        Beta_i=inv(X`*X)*X`*Y;
        Beta[i,1]=beta_i[2,1];
end;
print Beta;
quit;
```

Double loops

We can easily modify the prior code to include rolling betas for different companies:

```
Proc iml;
Use ASB.SP500;
Read all var _num_ into return;

N=ncol(return);
T=nrow(return);
Beta=J(T-250,N-1,0);
Do j=1 to N-1;
Do i=1 to T-250;
        X=J(251,1,1)||return[i:250+i,1];
        Y=return[i:250+i,1+j];
        Beta_i=inv(X`*X)*X`*Y;
        Beta[i,j]=beta_i[2,1];
end;
end;
print Beta;
quit;
```

We modify the output matrix to include every company

The outer loop is lopping across the companies (columns)

The X matrix is always the same across different stocks, while we need to change the Y matrix, both across time and company

AARHUS UNIVERSITET
BUSINESS AND SOCIAL SCIENCES

If sentences

- If sentences are used within the following framework:

```
Proc iml;
If …. Then…;
Else if … else if… else if…;
Else … ;
Quit;
```

```
Proc iml;
A = 10;
If A=1 then B=2;
Else if A=2 then B=3;
Else if A=3 then B=4;
Else if A=4 then B=5;
Else B=10;
Print B;
Quit;
```

Assignment 2

Make a 7 times 5 matrix containing only 1's, name it: One.

By use of a double loop transform the matrix one into a matrix five.

One

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Five

3	3	3	3	3
3	3	3	3	3
3	3	3	3	3
3	3	3	3	3
3	3	3	3	3

Assignment 3

Make a vector of indicators (name it positive), which is 1 if the market return is strictly positive, and 0 if otherwise. As illustrated below (you need to use both loops and if sentences)

0.0075173	1
-0.024543	0
-0.000351	0
0.0175681	1
-0.00619	0
-0.000108	0
-0.009769	0

Basic statistics

- How to calculate T-values, Critical values, P-values etc. in IML?

- Lets do the following test in SAS:

$\mathrm{H}_0: \alpha = 0$ from the simple beta estimation

Standard error on estimate

- Recal from AEM, that the standard error on the parameter estimate can be found as:

$$se(\alpha) = \sqrt{S^2(X'X)^{-1}}$$ where

$$S^2 = \left(\frac{1}{T-1}\right)e'e$$ where

$$e = Y - \hat{Y}$$

AARHUS UNIVERSITET
BUSINESS AND SOCIAL SCIENCES

Standard error on estimate

```
proc iml;
use asb.sp500;
read all var _num_ into z;
T=nrow(z);
Y=z[,2];
X=j(t,1,1)||z[,1];
theta=inv(X`*X)*x`*y;

Yhat=X*theta;
e=Y-Yhat;
e2=e`*e;
sigma2=(1/(T-1))*e2;

se_alpha=sqrt(sigma2*(inv(x`*X)[1,1]));
quit;
```

Theta is now 2*1 matrix containing the alpha and beta estimate from the first stock

Yhat is a T*1 vector since (T*2)*(2*1)

e2 is now a scalar since (T*1)`*(T*1)

se_alpha is also just a scalar

se_alpha
0.0002242

Critical value

- Finding the critical value from a distribution:

- Two-sided using the **quantile** function, general:

CV=quantile('distr',1-alpha/2,param,param)

Critical value

T-distribution with 32 degrees of freedom:
CV=quantile("T",1-0.05/2,32)

cv
2.0369333

F-distribution with 5 and 10 degrees of freedom (one sided):
CV=quantile("F",1-0.05,5,10)

cv
3.3258345

Chi-squared-dist with 5 degrees of freedom: (one sided)
CV=quantile("Chisquare",1-0.05,5)

cv
11.070498

Normal distribution with mean 3 and variance 2:
CV=quantile("Normal",1-0.05/2,3,2)

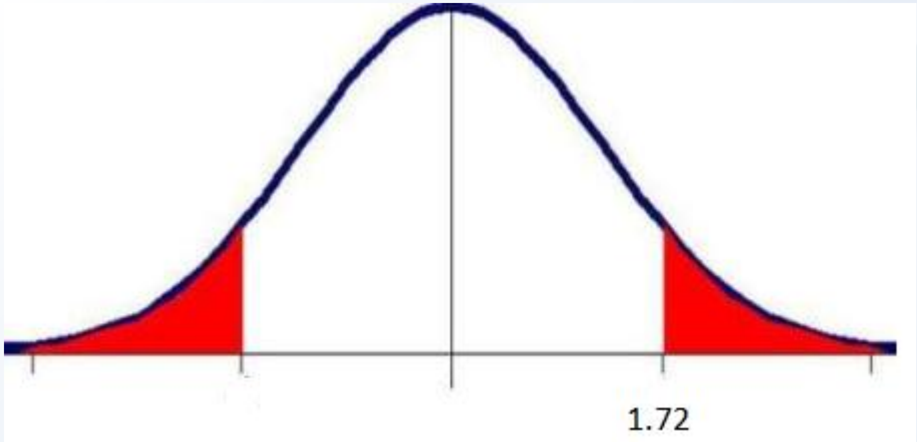
cv
6.919928

Standard normal distribution:
CV=quantile("Normal",1-0.05/2,0,1)

cv
1.959964

P-values

- What is the P-value in the following example?
- The test value (two sided) is 1.72, which need to be compared to a standard normal distribution. So what is the combined areal of the red areas?



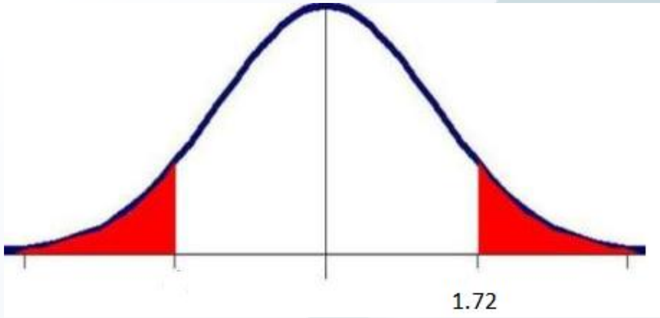
P-values

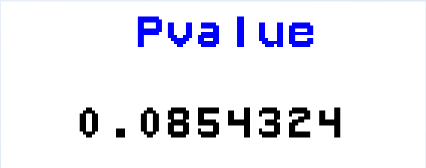
General by use of the **CDF-function** (two sided):

Pvalue=2*(1-CDF(“dist”,value,param1,param2))

In our example:

Pvalue=2*(1-CDF(“Normal”,1.72,0,1))





The example test

Our test was $H_0\colon \alpha = 0$

T-test: $\frac{\hat{\alpha}}{se(\hat{\alpha})} \sim T_{T-1}$

In SAS:

t_alpha	cv	prob_alpha
-0.134816	1.9609069	0.8927681

```
t_alpha=theta[1,1]/se_alpha;
cv=quantile("T",1-0.05/2,T-1);
prob_alpha=2*(1-CDF("T",abs(t_alpha),T-1));
print t_alpha cv prob_alpha;
```

Random number generator

- The overall way to generate random numbers in SAS is done as followed:

```
Proc iml;
Call Randgen (result,"distname",parm1,parm2,parm3);
Quit;
```

Output matrix name From which distribution? Under which assumptions?

Random number generator

- How to generate 10 random numbers from the standard normal distribution?

```
Proc iml;
X=J(10,1,0);
Call randgen(x,"normal",0,1);
Print X;
Quit;
```

Other distributions:

Distribution	distname	parm1	parm2	parm3
Bernoulli	'BERNOULLI'	p		
Beta	'BETA'	a	b	
Binomial	'BINOMIAL'	p	n	
Cauchy	'CAUCHY'			
Chi-Square	'CHISQUARE'	df		
Erlang	'ERLANG'	a		
Exponential	'EXPONENTIAL'			
$F_{n,d}$	'F'	n	d	
Gamma	'GAMMA'	a		
Geometric	'GEOMETRIC'	p		
Hypergeometric	'HYPERGEOMETRIC'	N	R	n
Lognormal	'LOGNORMAL'			
Negative Binomial	'NEGBINOMIAL'	p	k	
Normal	'NORMAL'	θ	λ	
Poisson	'POISSON'	m		
T	'T'	df		
Table	'TABLE'	p		
Triangle	'TRIANGLE'	h		
Uniform	'UNIFORM'			
Weibull	'WEIBULL'	a	b	

Assignment 4

- Simulate 11230 random numbers from a poisson distribution with mean 2.3.
  - Calculate the average of these random numbers and check that this average equals 2.3.

Assignment 5

- Simulate 10000 random numbers from a standard normal distribution and call the vector return
- Simulate 10000 random numbers from a normal distribution with mean 1 and variance 4, name it market
- Using the market model calculate and test whether the beta is equal to 0. (hint: return=alpha+beta*market+error