

SAS/IML: A flexible Approach to Data Analysis

Xiaojing Wu & Tolu Sajobi

**School of Public Health
University of Saskatchewan**

Saskatoon SAS User Group Meeting
October 12, 2011

Outline

- ✧ Introduction
- ✧ Data Processing
 - Import & Export Functionalities
- ✧ Basic Operations in SAS/IML
- ✧ Examples

Introduction

- ✧ SAS/IML is defined as
SAS/IML defined as Interactive Matrix Language (IML)
is a programming platform of SAS software
It is invoked as

```
proc IML;  
  {Body};  
Quit;
```

- ✧ SAS/IML has two platforms
 - a) IML programming interface
 - b) SAS/IML studio

Introduction

- ✧ SAS IML views datasets as matrices indexed by rows and columns
 - The dimension of the matrix is defined by the number of rows and columns. An m by n matrix has mn elements with m rows and n columns

- ✧ SAS/IML allows data analysts/programmers to conduct more analyses outside the standard procedures in SAS/STAT

- ✧ Useful for simulation studies

- ✧ A good platform for integrating programs from other softwares such as R (**SAS/IML Studio**)

Data Processing

- ✧ SAS datasets can be imported directly into SAS/IML
- ✧ Several data manipulations can also be obtained from using SAS/IML including data import, export, sorting, and imputation

Data Import to SAS/IML

SAS Data Step:

```
data one;  
input x y z;  
cards;  
1 2 3  
4 5 6;  
run;
```

SAS/IML Step

```
proc IML;  
use one;  
read all var {x y} into a;  
read all var {x y} into b where (z = 3);  
quit;
```

Data Export from SAS/IML

- ✧ A SAS data set can be created from a matrix using the create and append statements
- ✧ The columns of the matrix become the SAS data set variables and the rows of the matrix become the observations
 - E.g., a $n \times m$ matrix creates a SAS data set with m variables and n observations

Data Export from IML

Example:

```
proc iml;  
  a = {1 2 3 4 , 5 6 7 8, 9 10 11 12};  
  varnames = {x1 x2 x3 x4};  
  create newname from a [colname = varnames];  
  append from a;
```

Basic Operators

- ✧ **Elementwise Addition '+':** $c = a + b;$
Creates matrix $c = \begin{Bmatrix} 2 & 3, \\ 4 & 5 \end{Bmatrix};$
- ✧ **Elementwise Subtraction '-':** $d = a - b;$
Creates matrix $d = \begin{Bmatrix} 0 & 1, \\ 2 & 3 \end{Bmatrix};$
- ✧ **Elementwise Multiplication '#':** $e = a \# b;$
Creates matrix $e = \begin{Bmatrix} 1 & 2, \\ 3 & 4 \end{Bmatrix};$

Basic Operators

✧ **Elementwise Division ‘/’:** $f=a / b;$

Creates matrix $f=\begin{Bmatrix} 1 & 2, \\ 3 & 4 \end{Bmatrix};$

✧ **Element-wise Power ‘##’:** $g=a ## 2;$

Creates matrix: $g=\begin{Bmatrix} 1 & 4, \\ 9 & 16 \end{Bmatrix};$

✧ **Elementwise square root ‘sqrt’:** $h=sqrt (g);$

Creates matrix: $h=\begin{Bmatrix} 3 & 1; \\ 4 & 2 \end{Bmatrix};$

Basic Operators

- ✧ Column or Row sum using subscripts operator '[+]': $s=a[+,]$ or $t=a[, +]$;

Creates column sum of a : $c=\{4\ 6\}$;

Creates row sum of a: $c=\{3,7\}$;

- ✧ Other frequently used operators includes

Exponential: EXP

Natural Log: LN

Matrix Multiplication

Multiplication Rules

- The first matrix must have the same number of columns as the second matrix has rows
- The new matrix has the same number of rows as the first matrix and the same number of columns as the second matrix.

✧ Example:

```
proc IML;  
a = {1 2, 3 4};  
b = {1 2};  
c = b * a;  
quit
```

Creates the matrix: $c = \{7 \ 10\}$;

Examples

- ✧ Descriptive Analysis
- ✧ Multiple Linear Regression

Acknowledgements

- ✧ Canadian Institute of Health Research (CIHR) -
 - Vanier Graduate Scholarship

- ✧ SPH Population Health Data Laboratory