SAS® GLOBAL FORUM 2016

IMAGINE. CREATE. INNOVATE.

Effective ways of handling various file types and importing techniques using SAS®9.4

Divya Dadi
Rahul Jhaver

#SASGF

Effective ways of handling various file types and importing techniques using SAS®9.4

Divya Dadi and Rahul Jhaver

MS in MIS, SAS® and OSU Data Mining Certificate, Oklahoma State University

Introduction

Data-driven decision making is critical for any organization to thrive in this fiercely competitive world. The decision-making process has to be accurate and fast in order to stay a step ahead of the competition. One major problem organizations face is huge data load times in loading or processing the data. Reducing the data loading time can help organizations perform faster analysis and thereby respond quickly. In this paper, we compared the methods that can import data of a particular file type in the shortest possible time, and thereby increase the efficiency of decision making. SAS® takes input from various file types (such as XLS, CSV, XLSX, ACCESS, and TXT) and converts that input into SAS data sets. To perform this task, SAS® provides multiple solutions (such as the IMPORT procedure, the INFILE statement, and the LIBNAME engine) to import the data. We observed the processing times taken by each method for different file types with a data set containing 65,535 observations and 11 variables. We executed the procedure multiple times to check for variation in processing time. From these tests, we recorded the minimum processing time for the combination of procedure and file type. From our analysis of processing times taken by each importing technique, we observed that the shortest processing times for CSV and TXT files, XLS and XLSX files, and ACCESS files are the INFILE statement, the LIBNAME engine, and PROC IMPORT, respectively.

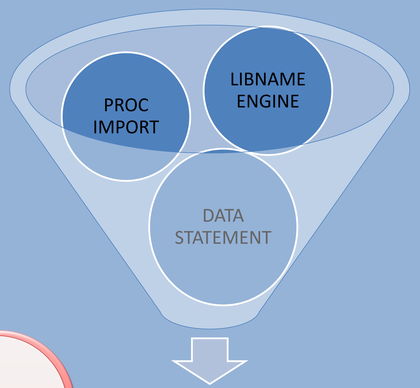
Importing Methods and File Types

Each importing method has different processing times for different file types. When we have large datasets containing millions of records, reducing the processing times becomes of utmost important. Thus finding an optimal method for a particular file type reduces the processing time while importing the data.

- Figure 1 shows the most common file types used in organizations.
- Figure 2 shows the major techniques used for importing files into a SAS dataset.

Fig 1: File types

Fig 2: Major Importing Methods



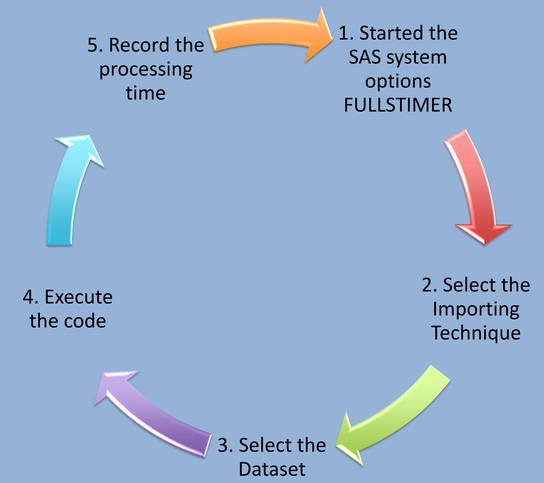
Analysis of processing times while importing the data

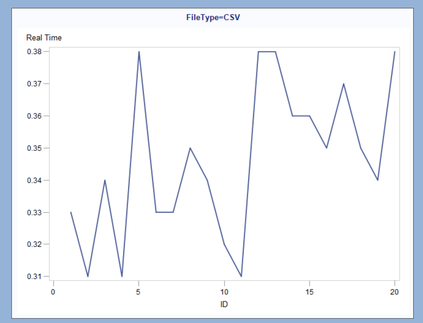
Steps considered for analysis of processing times while importing the data:

- Executed the code individually in a separate SAS session for each importing technique.
- Included SAS code which is essential for importing data.
- Ran the code on same system multiple times in order to check the variations in processing times. This can handle the resource availability of the system. The CPU time and Real time are noted further.
- Figure 3 shows the process followed to record processing time.
- Data used for import has 65,535 rows and 11 variables.

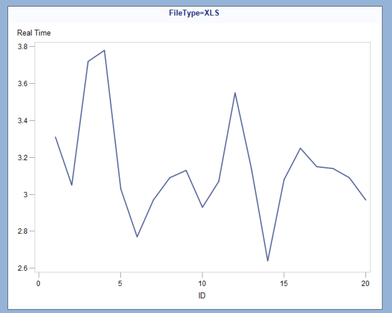
The graphs below depict the variations observed in *Processing Times* (for 20 observations) for multiple file types.

Fig 3: Procedure for recording the processing times

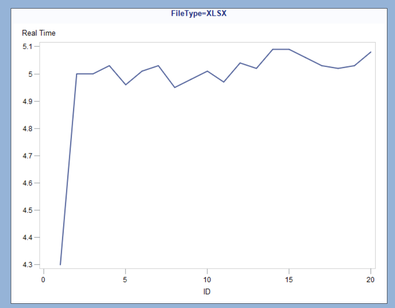




4 (a) File type “CSV”



4 (b) File type “XLS”



4 (c) File type “XLSX”

Fig 4: Processing times observed for different file types

Effective ways of handling various file types and importing techniques using SAS®9.4

Divya Dadi and Rahul Jhaver

MS in MIS, SAS® and OSU Data Mining Certificate, Oklahoma State University

- The variations in the processing times between the type of file and the number of runs are considerably small. Figure 4 depicts the same.
- The statistical significance of the variation in processing times is tested by performing ANOVA test on the data.
- From ANOVA test results, the processing times are proved to be statistically different for different file types.
- Below are the results of the Tukey Test performed on the data:

FileType	Real_Time LSMEAN	LSMEAN Number
ACCESS	0.59100000	1
CSV	0.34600000	2
DATA-CSV	0.29050000	3
DATA-TXT	0.32850000	4
LIBNAME-XLS	0.97600000	5
LIBNAME-XLSX	2.21550000	6
TXT	0.38800000	7
XLS	3.14300000	8
XLSX	4.98500000	9

Least Squares Means for effect FileType
Pr > |t| for H0: LSMean(i)=LSMean(j)
Dependent Variable: Real_Time

i/j	1	2	3	4	5	6	7	8	9
1		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
2	<.0001		0.8919	1.0000	<.0001	<.0001	0.9779	<.0001	<.0001
3	<.0001	0.8919		0.9883	<.0001	<.0001	0.2489	<.0001	<.0001
4	<.0001	1.0000	0.9883		<.0001	<.0001	0.8477	<.0001	<.0001
5	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001
6	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001
7	<.0001	0.9779	0.2489	0.8477	<.0001	<.0001		<.0001	<.0001
8	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001
9	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	

FileType	User_CPU_Time LSMEAN	LSMEAN Number
ACCESS	0.27650000	1
CSV	0.16550000	2
DATA-CSV	0.13050000	3
DATA-TXT	0.13900000	4
LIBNAME-XLS	0.95150000	5
LIBNAME-XLSX	2.46300000	6
TXT	0.16200000	7
XLS	0.15850000	8
XLSX	4.39750000	9

Least Squares Means for effect FileType
Pr > |t| for H0: LSMean(i)=LSMean(j)
Dependent Variable: User_CPU_Time

i/j	1	2	3	4	5	6	7	8	9
1		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
2	<.0001		0.5757	0.8628	<.0001	<.0001	1.0000	1.0000	<.0001
3	<.0001	0.5757		0.9999	<.0001	<.0001	0.7077	0.8222	<.0001
4	<.0001	0.8628	0.9999		<.0001	<.0001	0.9343	0.9750	<.0001
5	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001
6	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001
7	<.0001	1.0000	0.7077	0.9343	<.0001	<.0001		1.0000	<.0001
8	<.0001	1.0000	0.8222	0.9750	<.0001	<.0001	1.0000		<.0001
9	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	

Fig 5: Tukey Test Results

Trade Offs for Importing Techniques

- When trying to find the effective ways of importing data through different techniques, there are certain trade-offs which are to be considered, as shown in figure6.

Functionality	Trade off
Using INFILE statement with DATA step	Leads to increase the Programmer's time
Using PROC IMPORT	May compromise with the data type and the format of the variables
LIBNAME	Cannot import any file other than xls or xslx

Fig 6: Trade offs for Importing Methods

RESULTS

- From the Tukey test results it is seen that the variation in the processing times among multiple file types are significantly different.
- The average of the processing times for 20 runs is considered to formulate the crosstab shown in figure 7. It shows the different run times for each method to import multiple file types.
- From our analysis, the processing times taken by each importing technique, we have obtained the results as shown in figure 8.

Fig 7: Cross Tab

File type	INFILE Statement		PROC IMPORT		LIBNAME	
	Real Time	CPU Time	Real Time	CPU Time	Real Time	CPU Time
CSV	0.14	0.15	0.32	0.19	NA	NA
XLS	NA	NA	3.74	0.83	0.92	0.90
XLSX	NA	NA	5.32	4.39	2.16	2.69
TXT	0.31	0.08	0.33	0.15	NA	NA
ACCESS	NA	NA	0.67	0.31	NA	NA

Fig 8: Results

FILE TYPE	IMPORTING Technique
CSV	INFILE Statement
XLS	LIBNAME Engine
XLSX	LIBNAME Engine
TXT	INFILE Statement
ACCESS	PROC IMPORT

REFERENCES

SAS® 9.4 Language Reference

Primary Documentation for PROC IMPORT:

Primary Documentation for INFILE Statement:

De-Mystifying the SAS® LIBNAME Engine in Microsoft Excel: A Practical Guide

Acknowledgement

We thank Dr. Goutam Chakraborty, Professor, Department of Marketing and founder of SAS® and OSU Data Mining Certificate Program - Oklahoma State University for his support throughout the research.

SAS® GLOBAL FORUM 2016

IMAGINE. CREATE. INNOVATE.

LAS VEGAS | APRIL 18-21

#SASGF