

Introduzione a SAS

Gianluca Della Vedova

Univ. Milano-Bicocca
<http://gianluca.dellavedova.org>

16 maggio 2016, revisione 0189fde

- Ufficio U14-2041
- <http://gianluca.dellavedova.org>
- gianluca.dellavedova@unimib.it

Finestre

- Editor (F5)
- Log (F6)
- Output (F7)
- Icona Esegui (F8)

Altre finestre

- Explorer
- Viewtable
- Risultati

Data set

- righe = osservazioni
- colonne = variabili

Data Step

```
DATA nome;  
/*  
  elenco variabili (con formato)  
  come inserire i dati  
*/  
RUN;
```

Data Step: dati nel programma

```
DATA nomipersone;  
  INPUT nome$ cognome$ altezza peso;  
  DATALINES;  
    Mario Rossi 178 69  
    Lucia Bianchi 170 57  
    Andrea Verdi 169 69  
  ;  
RUN;
```

Valori mancanti

- Rappresentati da un punto .
- Inseriti come spazi o punti

Data Step: Importazione

- Da file di dati grezzi (.txt .csv .dat)
- Da wizard di importazione
- Da viewtable

Gianluca Della Vedova Introduzione a SAS

Data set

- nome di data set ≤ 32 caratteri
- lettere e cifre
- cifre in fondo
- nome di variabili ≤ 32 caratteri

Gianluca Della Vedova Introduzione a SAS

Filesystem

- Ogni data set corrisponde ad un file
- Una directory (cartella) contenente data set = **libreria**
- LIBNAME nomelibreria directory;
- data set permanente = in una libreria
- data set temporaneo = libreria WORK = no libreria
- nome libreria ≤ 8 caratteri

Gianluca Della Vedova Introduzione a SAS

Filesystem

- Ogni data set corrisponde ad un file
- Dati: data set oppure file di dati grezzi

Gianluca Della Vedova Introduzione a SAS

Data Step: da dati grezzi

```
DATA nomipersone;
  INFILE 'E:\anagrafe.txt';
  INPUT nome$ cognome$ altezza peso;
RUN;
```

Gianluca Della Vedova Introduzione a SAS

Data Step: da dati grezzi 2

```
LIBNAME esercizi "E:\libreriaSAS";

DATA esercizi.nomipersone;
  INFILE 'E:\anagrafe.txt';
  INPUT nome$ cognome$ altezza peso;
RUN;

Dati separati da spazi
```

Gianluca Della Vedova Introduzione a SAS

Data Step: da dati grezzi 2

```
LIBNAME esercizi "E:\libreriaSAS";

DATA esercizi.nomipersone;
  INFILE 'E:\anagrafe.txt' LRECL=2000;
  INPUT nome$ cognome$ altezza peso;
RUN;

Se la riga contiene > 256 caratteri.
```

Gianluca Della Vedova Introduzione a SAS

Data Step - lettura da file

```
DATA nome;
  INFILE nomefile DLM=', ' DSD;
  INPUT nome$ cognome$ altezza peso;
RUN;

Dati separati da virgola
DSD: campi alfanumerici racchiusi da virgolette
```

Gianluca Della Vedova Introduzione a SAS

Data Step - lettura da file

```
DATA nome;  
  INFILE nomefile DLM='09'x;  
  INPUT nome$ cognome$ altezza peso;  
RUN;
```

Dati separati da tabulazione

Data Step - lettura da file

- FIRSTOBS: da quale riga iniziare
- OBS: quante righe leggere
- MISSOVER: variabili non assegnate=MISSING
- TRUNCOVER: dati a fondo riga = scartati

Formato a colonne

```
DATA nome;  
  INFILE nomefile;  
  INPUT nome$ 1-10 cognome$ 11-20  
         altezza 21-25 peso 28-30;  
RUN;
```

- Il cognome inizia alla colonna n. 11 e termina alla colonna n. 20.
- Righello nel log
- Gli spazi non separano.

Formati Alfanumerici

: \$20.

Rappresenta il numero massimo di caratteri di un campo (in questo caso 20). Il dato termina con il primo spazio.

\$20.
Esattamente 20 caratteri.

Formati Numerici

10.3
Numero totale di caratteri/cifre
.
Numero di cifre decimali

Formati Date

- DDMMYY8.
- DATE7.
- Le date sono rappresentate in un formato interno, come numero di giorni dall'1/1/1960.
- Quindi una data è un numero
- Differenza fra date

Funzioni su numeri

- SUM(var1 var2 var3), MIN(var1 var2 var3), MAX(var1 var2 var3), MEAN(var1 var2 var3),
- SUM(of var1-var3)
- funzioni su variabili numeriche.
- Attenzione ai valori mancanti.

Funzioni su numeri

- concatena due stringhe: CAT(var1, var2)
- estrae sottostringa: SUBSTR(var, inizio, lunghezza)
- converte in formato numerico: INPUT(var, informat)
- trova una sottostringa: INDEX(var, sottostringa)
- calcola la lunghezza: LENGTH(var)
- trasforma in maiuscolo: UPCASE(var)

Altre funzioni

- MONTH(data)
- YEAR(data), DAY(data)
- Sono funzioni che agiscono sulle date
- '01Mar2000'd

Gianluca Della Vedova Introduzione a SAS

Formati

```
DATA nome;  
  INFILE nomefile;  
  INPUT nome$ +1 cognome$ @30  
         (altezza peso) COMMA7. ;  
RUN;
```

- +1 salta un carattere
- le parentesi raggruppano variabili
- COMMA7. legge numeri con virgole come separatore migliaia
- DOLLAR7.2 come COMMA, ma con un dollaro all'inizio
- @30 va a colonna 30

Gianluca Della Vedova Introduzione a SAS

Formati

- @'code:' vai alla parte di dato grezzo che è uguale a code:
- / vai alla riga successiva dei dati
- @@ per avere più di una osservazione in una riga di dati

Gianluca Della Vedova Introduzione a SAS

Stampa

```
PROC PRINT;  
RUN;
```

- Stampa il contenuto dell'ultimo data set creato.

Gianluca Della Vedova Introduzione a SAS

Stampa

```
PROC PRINT DATA=prova;  
  VAR nome;  
RUN;
```

- Stampa solo le variabili specificate

Gianluca Della Vedova Introduzione a SAS

Stampa

```
PROC PRINT;  
  VAR nome;  
  ID cognome;  
  TITLE 'elenco delle persone';  
RUN;
```

- Usa la variabile cognome al posto di obs.
- Definisce il titolo.

Gianluca Della Vedova Introduzione a SAS

Stampa

```
PROC PRINT;  
  VAR nome;  
  ID cognome;  
  TITLE 'elenco delle persone';  
  WHERE peso > 80;  
RUN;
```

- WHERE definisce su quali osservazioni agire

Gianluca Della Vedova Introduzione a SAS

Etichette di stampa

```
LABEL variabile='Etichetta';
```

L'etichetta della variabile deve essere dentro il DATA step

Gianluca Della Vedova Introduzione a SAS

Formati di stampa

- FORMAT variabile 16.;
- FORMAT variabile DATE7.;
- FORMAT variabile 8.2;
- La FORMAT deve essere dentro la PROC PRINT

Gianluca Della Vedova Introduzione a SAS

Formati di stampa

```
PROC PRINT;
  VAR nome;
  ID cognome;
  TITLE 'elenco delle persone';
  WHERE peso > 80;
  FORMAT (cognome nome) $30.;
RUN;
```

Gianluca Della Vedova Introduzione a SAS

Riepilogo data set

```
PROC CONTENTS DATA=data set;
RUN;
```

Gianluca Della Vedova Introduzione a SAS

Nuovi formati

```
PROC FORMAT;
  VALUE formato 0='Rosso'
                1='Giallo'
                2='Blu';
RUN;
```

- Adesso formato. è utilizzabile in una istruzione format.

Gianluca Della Vedova Introduzione a SAS

Nuovi formati

```
PROC FORMAT;
  VALUE $formato2 'RED'='Rosso'
                 'YELLOW'='Giallo'
                 'BLUE'='Blu';
RUN;
```

Gianluca Della Vedova Introduzione a SAS

Nuovi formati

```
PROC PRINT DATA=vario;
  FORMAT data DATE7.;
  FORMAT colore formato.;
RUN;
```

Gianluca Della Vedova Introduzione a SAS

Data Step - copia

```
DATA nome;
  SET data set originale;
  KEEP variabili da tenere;
RUN;
```

```
DATA nome;
  SET data set originale;
  DROP variabili da eliminare;
RUN;
```

```
DATA nome;
  SET data set originale;
  KEEP variabili da tenere;
  RENAME=(vecchia=nuova);
RUN;
```

Gianluca Della Vedova Introduzione a SAS

Data Step - copia

```
DATA nomipersone;
  SET persone;
  KEEP nome cognome;
RUN;
```

Gianluca Della Vedova Introduzione a SAS

Creazione variabili

```
DATA persone2;
  SET persone;
  rapporto=altezza/peso;
RUN;
```

- Le operazioni fra variabili creano nuove variabili
- Operazioni usuali, + - / * **

Selezione di osservazioni

- IF condizione;
- DATA personealte;
 SET persone;
 IF altezza GT 180;
 RUN;
- Operatori confronto, EQ =, GT >, GE ≥, LT <, LE ≤, NE ≠
- Valori non numerici devono essere racchiusi da apici

Selezione di osservazioni 2

```
DATA personealte;
  SET persone;
  IF altezza LE 180 THEN DELETE;
RUN;
```

- Rimozione di osservazione

Istruzioni condizionali

```
IF condizione THEN DO;
  istruzione 1;
  istruzione 2;
  istruzione 3;
END;
```

- Più semplice se una sola istruzione
- IF condizione THEN istruzione;

Condizioni complesse

```
DATA personealteemagre;
  SET persone;
  IF altezza GE 180 AND peso LE 70;
RUN;
```

- Operatori logici: AND, OR, NOT

Condizioni complesse 2

```
DATA personealteemagre;
  SET persone;
  IF altezza GE 180 AND peso LE 70 THEN
    TIPO = 'A';
  ELSE IF altezza LT 170 THEN
    TIPO = 'B';
  ELSE IF peso GT 90 THEN
    TIPO = 'C';
  RUN;
```

- Cosa succede se un'osservazione non soddisfa alcuna condizione?
- E se ne soddisfa più di una?

Esercizio: BMI

$$BMI = \frac{\text{peso}}{\text{altezza}^2}$$

- $BMI < 18.5 \Rightarrow$ sottopeso
- $18.5 \leq BMI < 25 \Rightarrow$ OK
- $25 \leq BMI < 30 \Rightarrow$ sovrappeso
- $BMI \geq 30 \Rightarrow$ obesità
- Modificare il data set attribuendo ad ogni persona la sua classificazione

Lettura vs. Scrittura

- INFILE vs. FILE
- INPUT vs. PUT

Data Step - copia

```
DATA nomipersone;  
SET persone;  
????  
KEEP nome cognome;  
RUN;
```

- Parte di data step eseguita per ogni osservazione del data set di origine
- Ciclo implicito
- variabili inizialmente mancanti

Gianluca Della Vedova Introduzione a SAS

Retain

- RETAIN mantiene il valore di una variabile per osservazioni diverse
- senza RETAIN: variabile MISSING
- con RETAIN: mantiene valore precedente
- Somma \Rightarrow RETAIN
- $a + (5 * b)$
- $a+1$;

Gianluca Della Vedova Introduzione a SAS

Sequenza di variabili

- temp1 temp2 temp3 temp4 temp5
- temp1-temp5
- Sono variabili individuali

Gianluca Della Vedova Introduzione a SAS

Array

- Elemento array = nome alternativo variabile
- temp[1]
- nome collettivo dell'array + indice
- array temp[5];
- Gli array devono essere **dichiarati** con il numero di elementi.

Gianluca Della Vedova Introduzione a SAS

Array

- temp[1] singolo elemento dell'array
- temp nome array
- array temp[5];
- Associa le variabili temp1-temp5 agli elementi dell'array temp.
- array temp[5] a b c d e;
- Associa le variabili a b c d e agli elementi dell'array temp.

Gianluca Della Vedova Introduzione a SAS

Ciclo

- Ripetere più volte delle istruzioni.
 - numero di volte = contatore = variabile dedicata
 - Ciclo DO
- ```
array temp[5];
do i=1 to 5;
 if temp[i]=. then temp[i]=0;
end;
```

Gianluca Della Vedova Introduzione a SAS

## Gestione variabili

- Il corpo di un data set viene ripetuto per ogni osservazione del data set originario.
- Il contenuto delle variabili vengono distrutte all'inizio dell'esecuzione di ogni osservazione.
- $i+1$ ;
- RETAIN variabile;

Gianluca Della Vedova Introduzione a SAS

## Ordinare un data set

```
PROC SORT DATA=data set;
 BY variabile;
 BY DESCENDING variabile2;
 BY variabile1 DESCENDING variabile2;
RUN;
```

Gianluca Della Vedova Introduzione a SAS

## Ordinare = Partizionare

- Stesso valore = osservazioni consecutive = insieme nella partizione
- FIRST.variabile = prima osservazione dell'insieme
- LAST.variabile = ultima osservazione dell'insieme

Gianluca Della Vedova Introduzione a SAS

## Esempio

```
PROC SORT DATA=nuovometeo;
 BY provincia;
RUN;
DATA primo;
 SET nuovometeo;
 BY provincia;
 IF FIRST.provincia;
RUN;
```

Gianluca Della Vedova Introduzione a SAS

## Raggruppare osservazioni 2

- FIRST.chiave: azzerare i
- Ogni osservazione: scrivere il dato letto nella i-esima posizione di un array, incrementare i
- LAST.chiave: OUTPUT

Gianluca Della Vedova Introduzione a SAS

## Raggruppare osservazioni 3

| Provincia | temp | i | FIRST.provincia | LAST.provincia |
|-----------|------|---|-----------------|----------------|
| Milano    | 23   | 1 | V               |                |
| Milano    | 20   | 2 |                 |                |
| Milano    | 22   | 3 |                 |                |
| Milano    | 12   | 4 |                 |                |
| Milano    | .    | 5 |                 | V              |
| Pavia     | 24   | 1 | V               |                |
| Pavia     | 21   | 2 |                 |                |
| Pavia     | 19   | 3 |                 |                |
| Pavia     | 24   | 4 |                 |                |
| Pavia     | 21   | 5 |                 | V              |

Gianluca Della Vedova Introduzione a SAS

## Uso di first e last

- Sono predicati
- Vengono utilizzati in IF

Gianluca Della Vedova Introduzione a SAS

## Raggruppare osservazioni 1

- Una variabile **chiave**
- Ordinare il data set con BY chiave
- Gestire un contatore i per contare la posizione dell'osservazione corrente all'interno del gruppo.

Gianluca Della Vedova Introduzione a SAS

## Raggruppare osservazioni 3

| Provincia | temp | i | FIRST.provincia | LAST.provincia |
|-----------|------|---|-----------------|----------------|
| Milano    | 23   |   |                 |                |
| Milano,   | 20   |   |                 |                |
| Milano,   | 22   |   |                 |                |
| Milano,   | 12   |   |                 |                |
| Milano,   | .    |   |                 |                |
| Pavia,    | 24   |   |                 |                |
| Pavia,    | 21   |   |                 |                |
| Pavia,    | 19   |   |                 |                |
| Pavia,    | 24   |   |                 |                |
| Pavia,    | 21   |   |                 |                |

Gianluca Della Vedova Introduzione a SAS

## Calcolare statistiche

```
PROC MEANS DATA=data set N MEAN;
 VAR variabile1 variabile2;
run;
```

- Seleziona variabili
- Altrimenti su tutte le variabili numeriche

Gianluca Della Vedova Introduzione a SAS

## Calcolare statistiche

```
PROC MEANS DATA=dataset N MEAN;
 VAR variabile1 variabile2;
 CLASS variabile3;
run;
```

- Stratifica le statistiche

## Calcolare statistiche

```
PROC MEANS DATA=dataset N MEAN;
 VAR variabile1 variabile2;
 CLASS variabile3;
 ID variabile3;
 OUTPUT OUT=nuovodat MEAN=media
 MAXID=massimo;
run;
```

- Risultati in un dataset
- `_TYPE_` e `_FREQ_`

## Calcolare statistiche

```
PROC MEANS DATA=dataset NWAY N MEAN;
 VAR variabile1 variabile2;
 CLASS variabile3;
 ID variabile3;
 OUTPUT OUT=nuovodat
 MAXID(variabile1)=massimo
 MEAN(variabile1 variabile2)=media1 m2;
run;
```

- `NWAY`: Stratificazione massima

## Statistiche

`N`: numero osservazioni con valore non mancante  
`NMISS`: numero osservazioni con valore mancanti  
`NONOBS`: numero osservazioni  
`MEAN`: media  
`MEDIAN`: mediana  
`STDDEV`: deviazione standard  
`MAX`: massimo  
`SUM`: somma  
`ALPHA=.05` `CLM`: intervallo di confidenza

## Fondere due data set

```
DATA nuovods;
 SET vecchiods1 vecchiods2;
run;
```

- Le osservazioni vengono aggiunte sequenzialmente

## Fondere due data set

```
DATA nuovods;
 MERGE vecchiods1 vecchiods2;
 BY comune;
run;
```

- Le osservazioni sono mantenute
- Il campo comune in entrambi i data set guida la fusione.
- `BY` richiede un ordinamento.

## Analisi delle frequenze

```
PROC FREQ DATA=gare;
 TABLES posizioneg;
RUN;
```

## Analisi delle frequenze

```
PROC FREQ DATA=gare;
 TABLES posizioneg*partenzag;
RUN;
```

## Analisi delle frequenze

```
PROC FREQ DATA=votazioni;
 TABLES candidato*seggio;
 WEIGHT voti;
RUN;
```

- WEIGHT: variabile che indica un peso per ogni osservazione
- WEIGHT: variabile quantitativa
- TABLES: variabili qualitative

## Opzioni

```
PROC FREQ DATA=votazioni /CHISQ;
 TABLES candidato*seggio;
 WEIGHT voti;
RUN;
```

- CHISQ:  $\chi^2$
- CL: Intervalli di confidenza
- MEASURES: misure di associazione

## Output Delivery System: ODS

- Le procedure inviano dati all'ODS
- Destinazioni: LISTING (standard output), HTML, PDF, OUTPUT (data set), MARKUP (csv, xml, . . .), DOCUMENT
- Template: tabella, stile

## Output Delivery System

```
ODS TRACE ON;
PROC FREQ DATA=dataset;
 TABLES variabile;
run;
ODS TRACE OFF;
```

- Per avere nel log come ODS interpreta il programma

## Output Delivery System

```
PROC FREQ DATA=dataset;
 TABLES variabile;
 ODS SELECT CrossFreqsTab;
run;
```

- Seleziona elemento
- Evitare nome duplicati (Path)
- ODS EXCLUDE

## Output Delivery System

```
PROC FREQ DATA=dataset;
 TABLES variabile;
 ODS OUTPUT CrossFreqsTab = dataset2;
run;
```

## Output Delivery System

```
ODS PDF FILE = 'c:\Documents...\file.pdf';
PROC FREQ DATA=dataset;
 TABLES variabile;
run;
ODS PDF CLOSE;
```

## Output Delivery System

```
ODS HTML FILE = 'e:\file.html';
PROC FREQ DATA=dataset;
 TABLES variabile;
run;
ODS HTML CLOSE;
```

## Licenza d'uso

Quest'opera è distribuita con Licenza Creative Commons Attribuzione

- Condividi allo stesso modo 4.0 Internazionale

<http://creativecommons.org/licenses/by-sa/4.0/>.

La versione più recente, con i sorgenti per modificare l'opera si trova a

<http://gianluca.dellavedova.org> e

[https://github.com/gdv/lab\\_statistico-informatico](https://github.com/gdv/lab_statistico-informatico).