duplicates
duplicates
duplicates
duplicates
duplicates
duplicates
duplicates
duplicates
duplicates
dealing with duplicates
duplicates
duplicates

You will learn how to

- Detect duplicates
- Summarize duplicates
- Output duplicates

Sample data set: TEST

Obs	id	x
1	104	11
2	102	12
3	102	22
4	103	11
5	101	11
6	105	13
7	105	23
8	106	12
9	106	22
10	105	33
11	107	12
12	107	22
13	108	11
14	109	11

Do I have duplicates?

Detecting duplicates with PROC SQL

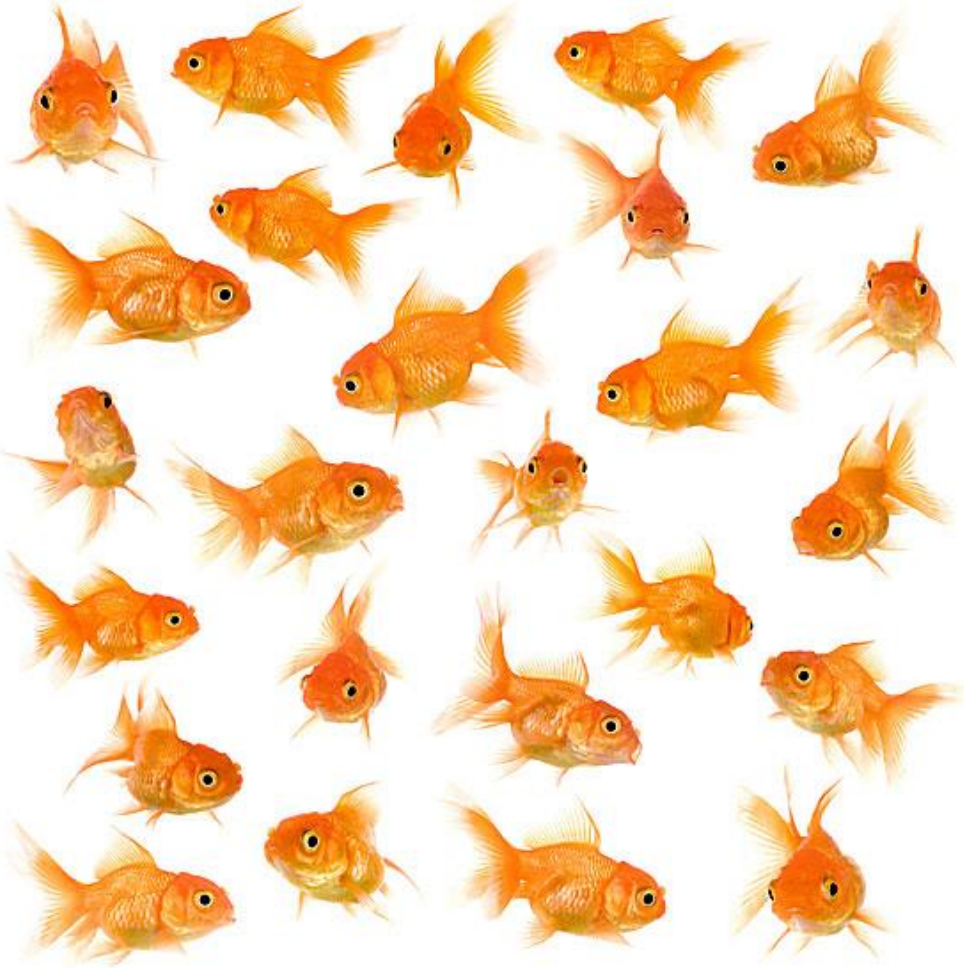
```
proc sql;
select count(distinct id) as Ndistinct,
       count(*) as Nobs
from test;
quit;
```

Ndistinct	Nobs
9	14

Detecting duplicates with PROC SQL

- One query
- NDISTINCT = NOBS [no duplicates]
- NDISTINCT ≠ NOBS [**duplicates**]

How many are there?



Summarizing duplicates with PROC FREQ

```
proc freq data=test;
tables id/out=freqout /* noprint */ ;
run;
```

id	Frequency	Percent	Cumulative Frequency	Cumulative Percent
101	1	7.14	1	7.14
102	2	14.29	3	21.43
103	1	7.14	4	28.57
104	1	7.14	5	35.71
105	3	21.43	8	57.14
106	2	14.29	10	71.43
107	2	14.29	12	85.71
108	1	7.14	13	92.86
109	1	7.14	14	100.00

Summarizing duplicates with PROC FREQ

```
proc print data=freqout;
run;
```

Obs	id	COUNT	PERCENT
1	101	1	7.1429
2	102	2	14.2857
3	103	1	7.1429
4	104	1	7.1429
5	105	3	21.4286
6	106	2	14.2857
7	107	2	14.2857
8	108	1	7.1429
9	109	1	7.1429

COUNT is a categorical variable that classifies each ID as single [1], double [2], triple [3], etc.

Summarizing duplicates with PROC FREQ

```
proc freq data=freqout;
tables count;
run;
```

Frequency Count				
COUNT	Frequency	Percent	Cumulative Frequency	Cumulative Percent
1	5	55.56	5	55.56
2	3	33.33	8	88.89
3	1	11.11	9	100.00

Summarizing duplicates with PROC FREQ

- Two steps
- Counts unique/duplicates
- Data-driven

How do I separate them?

SAS 9.3

Outputting duplicates with PROC SORT

```
proc sort data=test nouniquekeys
          uniqueout=singles out=dups;
by id;
run;
```

singles

Obs	id	x
1	101	11
2	103	11
3	104	11
4	108	11
5	109	11

dups

Obs	id	x
1	102	12
2	102	22
3	105	13
4	105	23
5	105	33
6	106	12
7	106	22
8	107	12
9	107	22

Outputting duplicates with PROC SORT

- One step
- **NOUNIQUEKEYS** deletes unique obs
- **UNIQUEOUT=***dsname* stores unique obs
- **OUT=***dsname* stores other [**duplicate**] obs

Conclusion

- **Detect** duplicates with PROC SQL
- **Summarize** duplicates with PROC FREQ
- **Output** duplicates with PROC SORT

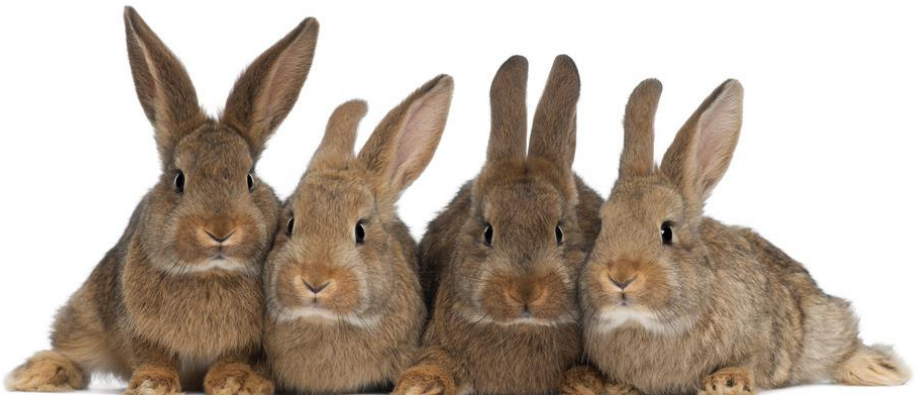
Macro

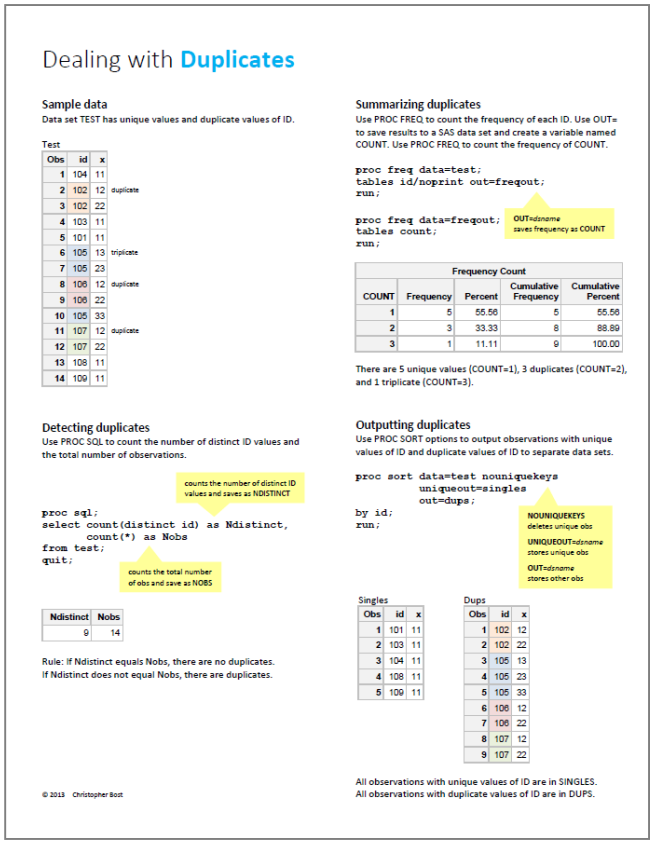
```
%macro dupcheck(inputds=,
                    var=,
                uniqueds=singles,
                   dupds=dups);

*Detecting duplicates with PROC SQL;
proc sql;
title "Checking for duplicates of &var in
&inputds";
select count(distinct &var) as Ndistinct,
       count(*) as Nobs
into :Ndistinct,:Nobs
from &inputds;
quit;

%if %sysevalf(&Ndistinct/&Nobs) ne 1
  %then %do;

*Summarizing duplicates with PROC FREQ;
proc freq data=&inputds;
tables &var/out=freqout noprint;
run;

proc freq data=freqout;
tables count;
title "FREQ of singles, doubles, etc. of
&var in data set &inputds";
run;

*Outputting duplicates with PROC SORT;
proc sort data=&inputds
          nouniquekeys
          uniqueout=&uniqueds
          out=&dupds;
by &var;
run;

%end;

%mend dupcheck;


/* Sample macro call */
%dupcheck(inputds=test,var=id)
```

Feedback

Questions and comments are valued and encouraged.

christopher.bost@mdrc.org
chrisbost@gmail.com





Dealing with **Duplicates**

Sample data

Data set TEST has unique values and duplicate values of ID.

Test

Obs	id	x	
1	104	11	
2	102	12	duplicate
3	102	22	
4	103	11	
5	101	11	
6	105	13	triplicate
7	105	23	
8	106	12	duplicate
9	106	22	
10	105	33	
11	107	12	duplicate
12	107	22	
13	108	11	
14	109	11	

Detecting duplicates

Use PROC SQL to count the number of distinct ID values and the total number of observations.

```
proc sql;
select count(distinct id) as Ndistinct,
       count(*) as Nobs
from test;
quit;
```

counts the number of distinct ID values and saves as NDISTINCT

counts the total number of obs and save as NOBS

Ndistinct	Nobs
9	14

Rule: If Ndistinct equals Nobs, there are no duplicates.
If Ndistinct does not equal Nobs, there are duplicates.

Summarizing duplicates

Use PROC FREQ to count the frequency of each ID. Use OUT= to save results to a SAS data set and create a variable named COUNT. Use PROC FREQ to count the frequency of COUNT.

```
proc freq data=test;
tables id/noprint out=freqout;
run;

proc freq data=freqout;
tables count;
run;
```

OUT=*dsname* saves frequency as COUNT

Frequency Count				
COUNT	Frequency	Percent	Cumulative Frequency	Cumulative Percent
1	5	55.56	5	55.56
2	3	33.33	8	88.89
3	1	11.11	9	100.00

There are 5 unique values (COUNT=1), 3 duplicates (COUNT=2), and 1 triplicate (COUNT=3).

Outputting duplicates

Use PROC SORT options to output observations with unique values of ID and duplicate values of ID to separate data sets.

```
proc sort data=test nouniquekeys
          uniqueout=singles
          out=dups;
by id;
run;
```

NOUNIQUEKEYS deletes unique obs

UNIQUEOUT=*dsname* stores unique obs

OUT=*dsname* stores other obs

Singles

Obs	id	x
1	101	11
2	103	11
3	104	11
4	108	11
5	109	11

Dups

Obs	id	x
1	102	12
2	102	22
3	105	13
4	105	23
5	105	33
6	106	12
7	106	22
8	107	12
9	107	22

All observations with unique values of ID are in SINGLES.
All observations with duplicate values of ID are in DUPS.