Shorten your SAS Code with Character Functions

Emily K.Q. Sisson, M.A.

Boston University Clinical and Translational Science Institute
Boston University School of Public Health Data Coordinating Center

Manipulating Character Strings

Concatenate character strings without removing leading or trailing blanks

```
Old = X1 || X2 || X3 || X4;
New = CAT(OF X1 – X4);
```

X1	X2	X3	X4	Old	New
This	is	a	sentence	This is a sentence	This is a sentence

Manipulating Character Strings

Concatenate character strings and remove trailing blanks

```
Old = TRIM(X1) || TRIM(X2) || TRIM(X3) ||
      TRIM(X4);
New = CATT(OF X1 – X4);
```

X1	X2	X3	X4	Old	New
This	is	a	sentence	This is asentence	This is asentence

Manipulating Character Strings

Concatenate character strings and remove leading **and** trailing blanks

```
Old = TRIM(LEFT(X1)) || TRIM(LEFT(X2)) ||
      TRIM(LEFT(X3)) || TRIM(LEFT(X4));
New = CATS(OF X1 – X4);
```

X1	X2	X3	X4	Old	New
This	is	a	sentence	Thisisasentence	Thisisasentence

Manipulating Character Strings

Concatenate character strings, remove leading and trailing blanks, and insert separator

```
Old = TRIM(LEFT(X1)) || " " || TRIM(LEFT(X2))
      || " " || TRIM(LEFT(X3)) || " " ||
      TRIM(LEFT(X4));
New = CATX(" ", X1 – X4);
```

X1	X2	X3	X4	Old	New
This	is	a	sentence	This is a sentence	This is a sentence

Manipulating Character Strings

Note: default length for variables created using CAT functions is different from the length that is obtained when using the concatenation operator (||)

```
data names;
        input last : $25. first : $15. @@;
        full_old = trim(last) || ', ' || trim(left(first));
        full_new = catx(', ', last, first);
datalines;
Sisson Emily Palmisano Joe Coleman Sharon
;
run;
```

Manipulating Character Strings

Results:

Obs	last	first	full_old	full_new
1	Sisson	Emily	Sisson, Emily	Sisson, Emily
2	Palmisano	Joe	Palmisano, Joe	Palmisano, Joe
3	Coleman	Sharon	Coleman, Sharon	Coleman, Sharon

Alphabetic List of Variables and Attributes			
#	Variable	Type	Len
2	first	Char	15
4	full_new	Char	200
3	full_old	Char	42
1	last	Char	25

Search for Variable Values

The next few examples will consider a dataset housing answers to a 10-question survey:

id	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10
1234	Y	Y	Y	Y	Y	Y	Y	Y	N	N
2345	N	N	N	N	N	N	N	N	N	N
3456	N	N	N	N	N	N	N	N	N	Y
4567	N	N	N	N	Y	N	N	N	N	N
5678	Y	N	N	Y	N	N	Y	N	N	Y

Search for Variable Values

Using the survey dataset, create a new dataset containing observations where at least one answer was ‘Y’.

Method 1: Brute Force:

```
data yes;
set survey;
if a1 = 'Y' or a2 = 'Y' or a3 = 'Y' or a4 =
'Y' or a5 = 'Y' or a6 = 'Y' or a7 = 'Y' or
a8 = 'Y' or a9 = 'Y' or a10 = 'Y';
run;
```

Search for Variable Values

Using the survey dataset, create a new dataset containing observations where at least one answer was 'Y'.

Method 2: Conventional Array/Do Loop:

```
data yes;
      set survey;
      array a(10);
      do i = 1 to 10 until (a(i) = 'Y');
      end;
      if i < 11;
run;
```

Search for Variable Values

CAT and FIND Functions:

```
data yes;
   set survey;
   if find(cat(of a1-a10), 'Y') > 0;
run;
```

id	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	cat(of a1-a10)
1234	Y	Y	Y	Y	Y	Y	Y	Y	N	N	YYYYYYYYNN
2345	N	N	N	N	N	N	N	N	N	N	NNNNNNNNNN
3456	N	N	N	N	N	N	N	N	N	Y	NNNNNNNNNY
4567	N	N	N	N	Y	N	N	N	N	N	NNNNYNNNNN
5678	Y	N	N	Y	N	N	Y	N	N	Y	YNNYNNYNNY

Search for Variable Values

Using the survey dataset, create a new dataset containing observations where at least TWO answers were 'Y'.

Conventional Array/Do Loop:

```
data yes;
set survey;
        array a(10);
        do i = 1 to 10;
        found_Y = sum(found_Y, (a(i) = 'Y'));
        end;
        if found_Y >= 2;
        drop i found_Y;
run;
```

Search for Variable Values

CAT and COUNTC Functions:

```
data yes;
   set survey;
   if countc(cat(of a1-a10), 'Y') >= 2;
run;
```

id	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	cat(of a1-a10)
1234	Y	Y	Y	Y	Y	Y	Y	Y	N	N	YYYYYYYYNN
2345	N	N	N	N	N	N	N	N	N	N	NNNNNNNNNN
3456	N	N	N	N	N	N	N	N	N	Y	NNNNNNNNNY
4567	N	N	N	N	Y	N	N	N	N	N	NNNNYNNNNN
5678	Y	N	N	Y	N	N	Y	N	N	Y	YNNYNNYNNY

Search for Variable Values

Character functions can also come in handy for numeric data! Consider a dataset housing answers to a 10-question survey:

id	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10
1234	5	5	3	3	2	1	1	1	1	1
2345	0	0	0	0	0	0	0	5	5	5
3456	1	1	2	2	4	0	0	0	0	0
4567	5	5	5	5	5	5	5	5	5	5
5678	1	0	1	0	1	0	5	5	5	3

Task: Using the numeric survey dataset, create a new dataset containing observations where at least one answer was 5.

Search for Variable Values

CAT and FIND Functions:

```
data five;
   set surveynum;
   if find(cat(of a:), '5');
run;
```

id	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	cat(of a:)
1234	5	5	3	3	2	1	1	1	1	1	5533211111
2345	0	0	0	0	0	0	0	5	5	5	0000000555
3456	1	1	2	2	4	0	0	0	0	0	1122400000
4567	5	5	5	5	5	5	5	5	5	5	5555555555
5678	1	0	1	0	1	0	5	5	5	3	1010105553

Practical Application: Diagnosis Codes

Each hospital admission can have up to five three-character diagnosis codes. Find patients with Diabetes (any DX code of 250).

Patient	DX1	DX2	DX3	DX4	DX5
001	025	022			
002	682	401	244	493	
003	592	401	493		
004	428	493	780	V43	250
005	250				
006	414	V45	401	250	

Practical Application: Diagnosis Codes

Why does

```
find(cat(of DX1-DX5), '250')
```

not work in this situation?

Patient	DX1	DX2	DX3	DX4	DX5
001	025	022			
002	682	401	244	493	
003	592	401	493		
004	428	493	780	V43	250
005	250				
006	414	V45	401	250	

Practical Application: Diagnosis Codes

Why does

```
find(cat(of DX1-DX5), '250')
```

not work in this situation?

Patient	DX1	DX2	DX3	DX4	DX5	cat(of DX1-DX5)
001	025	022				0**250**22
002	682	401	244	493		682401244493
003	592	401	493			592401493
004	428	493	780	V43	250	428493780V43**250**
005	250					**250**
006	414	V45	401	250		414V45401**250**

Practical Application: Diagnosis Codes

Solution: Use CATX to insert a separator!

```
find(catx('*',of DX1 – DX5), '250')
```

Patient	DX1	DX2	DX3	DX4	DX5	catx('*', of DX1-DX5)
001	025	022				025*022
002	682	401	244	493		682*401*244*493
003	592	401	493			592*401*493
004	428	493	780	V43	250	428*493*780*V43***250**
005	250					**250**
006	414	V45	401	250		414*V45*401***250**

Practical Application: Diagnosis Codes

Of course, ICD codes are usually a little more complicated than all this. The **first** 3 digits indicate a disease, and 0-2 digits may follow for a further-refined diagnosis…

Patient	DX1	DX2	DX3	DX4	DX5
001	25001	V180			
002	5680	78039	3250	49390	53081
003	4270	4111	4019	36250	
004	78659	25000	49320		
005	49392	4660	2449		
006	34839	2765	40493	4280	

Practical Application: Diagnosis Codes

Why do these statements not work to find Asthma (starting with 493) or Diabetes (starting with 250)?

```
diabetes = (find(catx('*',of dx1-dx5),'*250') gt 0);
asthma = (find(catx('*',of dx1-dx5),'*493') gt 0);
```

Patient	DX1	DX2	DX3	DX4	DX5
001	25001	V180			
002	5680	78039	3250	49390	53081
003	4270	4111	4019	36250	
004	78659	25000	49320		
005	49392	4660	2449		
006	34839	2765	40493	4280	

Practical Application: Diagnosis Codes

Why do these statements not work to find Asthma (starting with 493) or Diabetes (starting with 250)?

```
diabetes = (find(catx('*',of dx1-dx5),'*250') gt 0);
asthma = (find(catx('*',of dx1-dx5),'*493') gt 0);
```

Patient	DX1	DX2	DX3	DX4	DX5	catx('*',of dx1-dx5)
001	25001	V180				25001*V180
002	5680	78039	3250	49390	53081	5680*78039*3250*493390*53081
003	4270	4111	4019	36250		4270*4111*4019*36250
004	78659	25000	49320			78659*25000*49320
005	49392	4660	2449			49392*4660*2449
006	34839	2765	40493	4280		34839*2765*40493*4280

Practical Application: Diagnosis Codes

Solution:

```
diabetes = (find(catx('*', '*', of dx1-dx5),'*250') gt 0);
asthma = (find(catx('*', '*', of dx1-dx5),'*493') gt 0);
```

Patient	DX1	DX2	DX3	DX4	DX5	catx('*','*',of dx1-dx5)
001	25001	V180				***250**01*V180
002	5680	78039	3250	49390	53081	*5680*78039*3250***493**390*53081
003	4270	4111	4019	36250		*4270*4111*4019*36250
004	78659	25000	49320			*78659***250**00***493**20
005	49392	4660	2449			***493**92*4660*2449
006	34839	2765	40493	4280		*34839*2765*40493*4280

Character Functions Used: Summary

- CAT Operators (CAT, CATT, CATS, CATX)
  - Useful in evaluating multiple character strings simultaneously
- FIND
  - Allows you to identify the first position in a string that contains the specified search term
- COUNTC
  - Counts number of characters in a string that contain the specified search character
- Others:
  - http://support.sas.com/publishing/pubcat/chaps/59343.pdf

Thank you!

Emily K.Q. Sisson
Statistical Manager
Data Coordinating Center
Boston University School of Public Health
T: 617.638.5869
eq@bu.edu

Using ODS (Output Delivery System) Layout to Enhance Reporting

Joseph Palmisano, MA, MPH

BOSTON UNIVERSITY

Boston University Clinical and Translational Science Institute
Boston University School of Public Health Data Coordinating Center

Output Delivery System

- Create datasets from procedure output
- Generate high quality graphs
- Direct output to non-listing destinations
- Control format and style of output

ODS Destinations

Category	Destinations	Results
SAS Formatted	DOCUMENT	ODS document
	LISTING	SAS output listing
	OUTPUT	SAS data set
Third-Party Formatted	HTML	HTML file for online viewing
	MARKUP	Markup language tagsets
	PRINTER	Printable output in one of three different formats: PCL, PDF, or PS (PostScript)
	RTF	Output written in Rich Text Format for use with Microsoft Word 2000

Directing Output with ODS

- Syntax

```
ods <destination> file="<pathname>\<filename>";
<SAS Procedure>;
ods <destination> close;
```

- Examples
  - ```
    ods rtf file="C:\Example.rtf";
    <SAS Procedure>;
    ods rtf close;
    ```
  - ```
    ods pdf file="C:\Example.pdf";
    <SAS Procedure>;
    ods pdf close;
    ```

Helpful Options for Directing Output

- Columns =
  - Specify number of columns on page
  - HTML, RTF and Printer (PDF) destinations
  - Titles and footnotes unaffected, maintain normal positions
  - Graphs resize, tables wrap to fit

- Startpage =
  - Control page breaks
  - RTF and Printer (PDF) destinations
  - Only one title per page

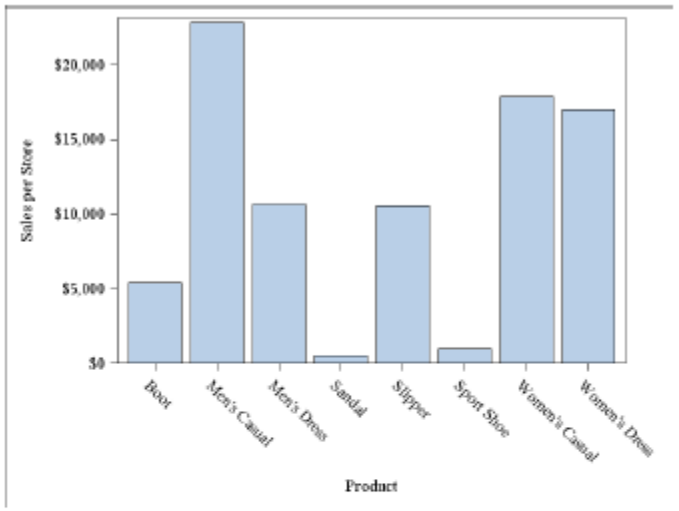
Helpful Options: Columns =

```
options orientation=landscape nodate nonumber;
ods pdf file="<pathname>\<filename>" columns=2;
Title "New York Shoe Sales";
proc print data = NYShoes noobs label;
        var Product Sales;
run;

proc sgplot data = NYShoes;
        where product ne 'Total';
        yaxis label="Sales per Store";
        vbar product / response=sales_store;
        title;
run;
ods pdf close;
```

New York Shoe Sales

Product	Total Sales
Boot	$97,151
Men's Casual	$456,985
Men's Dress	$191,755
Sandal	$554
Slipper	$252,758
Sport Shoe	$22,190
Women's Casual	$178,842
Women's Dress	$288,972
Total	$1,489,207



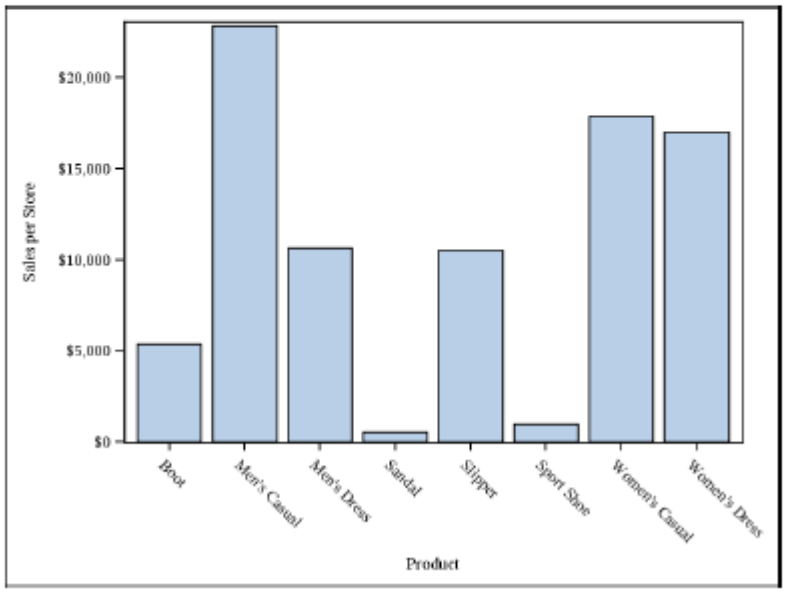
Helpful Options: Startpage =

```
options orientation=portrait nodate nonumber;
ods pdf file="<pathname>\<filename>" startpage=no;
Title "New York Shoe Sales";
proc print data = NYShoes noobs label;
        var Product Sales;
run;

proc sgplot data = NYShoes;
        where product ne 'Total';
        yaxis label="Sales per Store";
        vbar product / response=sales_store;
        title;
run;
ods pdf close;
```

New York Shoe Sales

Product	Total Sales
Boot	$97,151
Men's Casual	$456,905
Men's Dress	$191,755
Sandal	$554
Slipper	$252,758
Sport Shoe	$22,190
Women's Casual	$178,842
Women's Dress	$288,972
Total	$1,489,207



ODS Layout

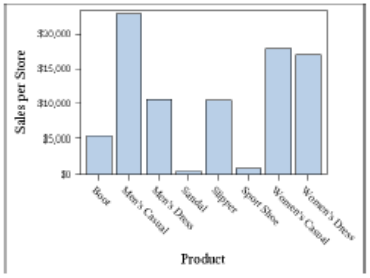
- Absolute Layout
  - Explicitly specify location and dimensions of output regions
  - Printer (PDF) destinations only
  - Restricted to one page per layout
  - Good for static reports or where precise layout is required

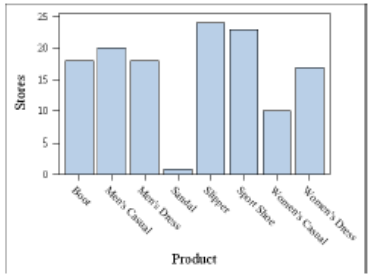
- Gridded Layout
  - Specify gridded output dimensions with columns= and rows=
  - HTML and Printer (PDF) destinations
  - Good for reports where dynamic sizing is required

New York Shoe Sales

Product	Total Sales
Boot	$97,151
Men's Casual	$456,985
Men's Dress	$191,755
Sandal	$554
Slipper	$252,758
Sport Shoe	$22,190
Women's Casual	$178,842
Women's Dress	$288,972
Total	$1,489,207

This report summarizes shoe sale data for the New York subsidiary for the 2012 calendar year.Mean's casual shoes accounted for the highest total sales as well as the highest total sales per store.Sandals accounted for the lowest total sales as well as the lowest total sales per store.Sport shoe sales were among the lowest in terms of total sales per store,though they are also among the most commonly available (sold in 23 stores).





Absolute Layout: Structure

```
ods pdf file="<pathname>\<filename>.pdf";
/*Layout Container*/
ods layout start x=0.5in y=0.5in width=7.5in height=9in;
/*Top Left Region Container*/
ods region x=0in y=0in width=3.5in height=4.25in;
/*Top Right Region Container*/
ods region x=4in y=0in width=3.5in height=4.25in;
/*Bottom Left Region Container*/
ods region x=0in y=4.0in width=3.25in height=4.25in;
/*Bottom Right Region Container*/
ods region x=3.5in y=4.0in width=3.25in height=4.25in;
ods layout end;
ods pdf close;
```

Absolute Layout: Detail (Top Left)

```
options orientation=portrait nodate nonumber;
ods pdf file="<pathname>\<filename>.pdf";
Title "New York Shoe Sales";
/*Layout Container*/
ods layout start x=0.5in y=0.5in width=7.5in height=9in;
/*Top Left Region Container*/
ods region x=0in y=0in width=3.5in height=4.25in;
proc print data = NYShoes noobs label;
       var Product Sales;
run;
```

Absolute Layout: Detail (Top Right)

```
/*Top Right Region Container*/
ods region x=4in y=0in width=3.5in height=4.25in;
ods pdf text = "This report summarizes shoe sale data for the
New York subsidiary for the 2012 calendar year. Men's casual
shoes accounted for the highest total sales as well as the
highest total sales per store. Sandals accounted for the lowest
total sales as well as the lowest total sales per store. Sport
shoe sales were among the lowest in terms of total sales per
store, though they are also among the most commonly available
(sold in 23 stores).";
```

Absolute Layout: Detail (Bottom Left)

```
/*Bottom Left Region Container*/
ods region x=0in y=4.0in width=3.25in height=4.25in;;
proc sgplot data = NYShoes;
        where product ne 'Total';
        yaxis label="Sales per Store";
        vbar product / response=sales_store;
        title;
run;
```

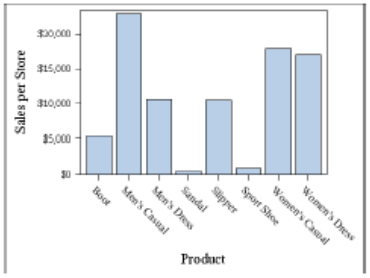
Absolute Layout: Detail (Bottom Right)

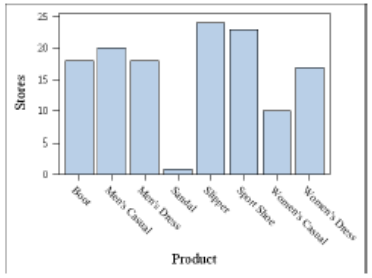
```
/*Bottom Right Region Container*/
ods region x=3.5in y=4.0in width=3.25in height=4.25in;;
proc sgplot data = NYShoes;
        where product ne 'Total';
        yaxis label="Stores";
        vbar product / response=stores;
        title;
run;
ods layout end;
ods pdf close;
```

New York Shoe Sales

Product	Total Sales
Boot	$97,151
Men's Casual	$456,985
Men's Dress	$191,755
Sandal	$554
Slipper	$252,758
Sport Shoe	$22,190
Women's Casual	$178,842
Women's Dress	$288,972
Total	$1,489,207

This report summarizes shoe sale data for the New York subsidiary for the 2012 calendar year.Mean's casual shoes accounted for the highest total sales as well as the highest total sales per store.Sandals accounted for the lowest total sales as well as the lowest total sales per store.Sport shoe sales were among the lowest in terms of total sales per store,though they are also among the most commonly available (sold in 23 stores).





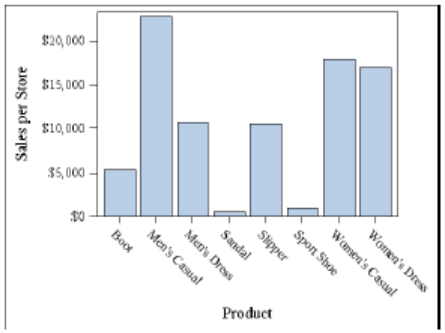
Gridded Layout: Structure

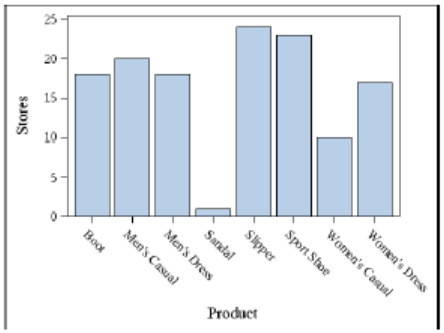
```
ods pdf file="<pathname>\<filename>.pdf";
/*Layout Container*/
ods layout start columns=2 rows=2
       /*column_widths=(3.5in 3.5in)*/;
/*Top Left Region Container*/
ods region width=3.5in;
/*Top Right Region Container*/
ods region width=3.5in;
/*Bottom Left Region Container*/
ods region width=3.5in;
/*Bottom Right Region Container*/
ods region width=3.5in;
ods layout end;
ods pdf close;
```

New York Shoe Sales

Product	Total Sales
Boot	$97,151
Men's Casual	$456,985
Men's Dress	$191,755
Sandal	$554
Slipper	$252,758
Sport Shoe	$22,190
Women's Casual	$178,842
Women's Dress	$288,972
Total	$1,489,207

This report summarizes shoe sale data for the New York subsidiary for the 2012 calendar year.Mean's casual shoes accounted for the highest total sales as well as the highest total sales per store.Sandals accounted for the lowest total sales as well as the lowest total sales per store.Sport shoe sales were among the lowest in terms of total sales per store,though they are also among the most commonly available (sold in 23 stores).





Resources

- SAS(R) 9.2 Output Delivery System: User's Guide
  - http://support.sas.com/documentation/cdl/en/odsug/61723/HTML/default/viewer.htm#a002291014.htm
- SAS(R) 9 ODS Tip Sheet
  - http://support.sas.com/rnd/base/ods/scratch/ods-tips.pdf
- Breaking New Ground with SAS® 9.2 ODS Layout Enhancements
  - http://support.sas.com/resources/papers/proceedings09/043-2009.pdf
- ODS LAYOUT to Create Publication-Quality PDF Reports of STD Surveillance Data
  - http://support.sas.com/resources/papers/proceedings10/216-2010.pdf

Applied Logistic Regression: SAS Coding Tips to Enhance Interpretation of Modeling and Output

Sharon Coleman MS, MPH

BOSTON UNIVERSITY

Boston University Clinical and Translational Science Institute
Boston University School of Public Health Data Coordinating Center

Analysis of Categorical or Event Outcomes

- Outcome (dependent variable): dichotomous (yes/no, event)
- Study design: either comparing groups on dichotomous outcome, or association with dichotomous outcome; Cross-sectional or Cohort or Case-Control or RC Trial
- The independent variable(s) can be measurement or categorical

A Little Refresher

- The logistic model is based on the odds of an event occurring:

  Based on the odds or probability, $p_x/(1-p_x)$

  $\log(\text{odds}(x)) = \log(p_x/1-p_x) = \beta_0 + \beta x$

  $\text{logit}(P) = \beta_0 + \beta x$

  The Odds Ratio (x1 vs x2) $= \dfrac{p_{x_1}/(1-p_{x_1})}{p_{x_2}/(1-p_{x_2})}$

The Odds Ratio

	Sharon misses	yoga class	
Cherry Garcia Ice Cream	Yes (1)	No (0)	Total
Yes (1)	7	3	10
No (0)	4	6	10
Total	11	9	20

OR of Cherry Garcia

- The probability of Sharon having Cherry Garcia when she skips yoga is 7/11 or 0.63 . The odds from this probability are (p/ 1-p) or 0.63/0.36= 1.75

- The probability of Sharon having CG when she attends class 3/9 or 0.33. The odds from this probability are 0.33/0.66= 0.50

- OR= 1.75/ 0.5 = 3.50

A Simple SAS Example

- Perhaps we want to look at the association between CHD and age
- CHD is coded as 1=yes 0=no
- Age is a measurement Independent variable

The "SIMPLE" option gives summary statistics
The (event = '1') is an efficient way of specifying the probability modeled (vs. using the descending option)

```
proc logistic data=chdage simple;
model chd (event='1') = age;run;
```

Be sure your output is modeling the event of interest

Response Profile		
Ordered Value	chd	Total Frequency
1	0	57
2	1	43

Probability modeled is chd=1.

The SAS Units Statement

- **The SAS UNITS statement - When we have a continuous IV often the value of a 1 unit change in x will not be very biologically interesting or clinically meaningful**

```
proc logistic data=chdage simple;
model chd (event='1') = age/ clodds =
wald;
units age= sd 5 10;
run;
```

The SAS Units Statement

- **The SAS UNITS statement - When we have a continuous IV often the value of a 1 unit change in x will not be very biologically interesting or clinically meaningful**

```
proc logistic data=chdage simple;
model chd (event='1') = age/ clodds =
wald;
units age= sd 5 10;
run;
```

Output from the Units Statement

Descriptive Statistics for Continuous Variables					
Variable	**chd**	**Mean**	**Standard Deviation**	**Minimum**	**Maximum**
age	0	39.175439	10.201755	20.000000	64.000000
	1	51.279070	9.979325	25.000000	69.000000
	Total	44.380000	**11.721327**	20.000000	69.000000

Odds Ratio Estimates and Wald Confidence Intervals				
Effect	**Unit**	**Estimate**	**95% Confidence Limits**	
Age (1 SD)	**11.7213**	3.670	2.112	6.378
age	1.0000	1.117	1.066	1.171
age	5.0000	1.741	1.376	2.204
age	10.0000	3.032	1.892	4.859

Reference Group Coding

Let us consider data from a study designed to identify risk factors associated with giving birth to a baby weighing less than 2500 grams (LBW)

Race is categorized as White=1 Black=2 and Other=3

We will consider Indicator variables for Race

RACE		
WHITE	0	0
BLACK	1	0
OTHER	0	1

(White=1 , Black=2, Other=3 codes in dataset)

What are the Odds of having a Low Birth Weight Baby Based on Race?

```
proc logistic data=work.lbw simple;
class race/ param=ref ref=first;
model low (event='1')= LWT RACE ;
run;
```

White Coded as 1 Becomes our Reference Group

Class Level Information			
Class	**Value**	**Design Variables**	
RACE	**1=white**	0	0
	2=Black	1	0
	3=Other	0	1

Odds Ratio Estimates			
Effect	**Point Estimate**	**95% Wald Confidence Limits**	
LWT	0.985	0.973	0.997
RACE 2 vs 1	2.948	1.133	7.672
RACE 3 vs 1	1.617	0.804	3.253

SAS defaults to Deviation from Means Coding

- If the parameter statements are not used- sas will default to deviation from the means coding

- This expresses effect as the deviation of the group mean from the overall mean.

Class Level Information			
Class	Value	Design Variables	
RACE	1	-1	-1
	2	1	0
	3	0	1

Back to some math

- Due to the S shape of the logistic distribution one must use the natural log (ln) to account for the shape

- $\ln(P)=\beta_0 + \beta x$

- The odds ratio is equal to *exp*(*B*)

- So we may need to exponentiate our beta coefficient or slope to get the odds ratio

exp(1.08)=2.94 and conversely ln(2.94)=1.08

Example with Proc Genmod

```
proc genmod data=work.lbw
descending;
class race (ref='1')/ param=ref;
model low = lwt race /dist=bin type3
link=logit ;
estimate "Black vs White" race  1 0
/ exp;
run;
```

White=1	0	0	0
Black=2	0	1	0
Other=3	0	0	1

Analysis Of Maximum Likelihood Parameter Estimates

Parameter		DF	Estimate		Wald 95% Confidence Limits		Wald Chi-Square	Pr > ChiSq
Intercept		1	0.8058	0.8452	-0.8507	2.4622	0.91	0.3404
LWT		1	-0.0152	0.0064	-0.0278	-0.0026	5.59	0.0181
RACE	2	1	**1.0811**	0.4881	0.1245	2.0376	4.91	0.0268
RACE	3	1	0.4806	0.3567	-0.2185	1.1797	1.82	0.1778
Scale		0	1.0000	0.0000	1.0000	1.0000		

Contrast Estimate Results

Label	Mean Estimate	Mean Confidence Limits		L'Beta Estimate	Standard Error	Alpha	L'Beta Confidence Limits		Chi-Square	Pr > ChiSq
Black vs White	0.7467	0.5311	0.8847	1.0811	0.4881	0.05	0.1245	2.0376	4.91	0.0268
Exp(Black vs White)				**2.9478**	**1.4387**	**0.05**	**1.1326**	**7.6724**		

Resources

- **Applied Logistic Regression , Hosmer, Lemeshow**
- **Boston Area SAS Users Group**
- **www.basug.org**
- **THANK YOU !**