Introduction to SAS

Part 2

Goals

- Data manipulation
- Important procedures
- Saving SAS results in a data set

Original CHDAGE data set

Obs	id	age	chd
1	1	20	0
2	2	23	0
3	3	24	0
4	4	25	0
5	5	25	1
Etc.			

Data manipulation

```
libname sdat 'C:\ERHS642';

data chdage; set sdat.chdage;
  * Create a categorical age variable with 4
categories *;
        if 20<=age<=30 then a=1;
  else if 30< age<=40 then a=2;
  else if 40< age<=50 then a=3;
  else if 50< age<=70 then a=4;
```

Continued on next slide

Data manipulation

Continued from previous slide

```
* Create the square root of age *;
  sqrt_age=sqrt(age);
  * Create the natural log of age *;
  ln_age = log(age);
  * Create age squared *;
  age_sq=age**2;

run;
```

CHDAGE data set with original and new variables

```
proc print data=chdage; run;
```

Obs	id	age	chd	a	sqrt_age	ln_age	age_sq
1	1	20	0	1	4.47214	2.99573	400
2	2	23	0	1	4.79583	3.13549	529
3	3	24	0	1	4.89898	3.17805	576
4	4	25	0	1	5.00000	3.21888	625
5	5	25	1	1	5.00000	3.21888	625

CHDAGE data set with only new variables

```
proc print data=chdage;
   var a sqrt_age ln_age age_sq;
run;
```

Obs	a	sqrt_age	ln_age	age_sq
1	1	4.47214	2.99573	400
2	1	4.79583	3.13549	529
3	1	4.89898	3.17805	576
4	1	5.00000	3.21888	625
5	1	5.00000	3.21888	625

Commonly used procedures

Creating descriptive statistics
- proc means
- proc univariate

Creating frequency tables
- proc freq

Logistic regression
- proc logistic

Creating graphs
- proc gplot

proc means, default output

```
proc means data=chdage; var age; run;
```

Analysis Variable : age age				
N	Mean	Std Dev	Minimum	Maximum
100	44.38000	11.7213265	20.00000	69.00000

If the var statement is omitted, results are shown for all variables in the data set

proc means, user-specified output

```
proc means data=chdage
   n mean min median max std;
var age; run;
```

Other options are possible

Analysis Variable : age age					
N	Mean	Min	Median	Max	Std Dev
100	44.3800	20.00	44.0000	69.00	11.7213265

proc univariate, basic stats

proc univariate data=chdage; var age; run;

Basic Statistical Measures			
Location		Variability	
Mean	44.38000	Std Deviation	11.72133
Median	44.00000	Variance	137.38949
Mode	30.00000	Range	49.00000
		Interquartile Range	20.50000

proc univariate, more basic stats

Moments			
N	100	Sum Weights	100
Mean	44.38	Sum Observations	4438
Std Deviation	11.7213	Variance	137.389
Skewness	-0.00814	Kurtosis	-0.98353
Uncorrected SS	210560	Corrected SS	13601.56
Coeff Variation	26.411	Std Error Mean	1.17213

proc univariate, basic tests

Tests for Location: Mu0=0				
Test	Statistic		p Value	
Student's t	t	37.86261	Pr > |t|	<.0001
Sign	M	50	Pr >= |M|	<.0001
Signed Rank	S	2525	Pr >= |S|	<.0001

proc univariate, quantiles

Quantiles	
Quantile	Estimate
100% Max	69.0
99%	67.0
95%	62.5
90%	59.5
75% Q3	55.0
50% Median	44.0
25% Q1	34.5
10%	29.5
5%	25.5
1%	21.5
0% Min	20.0

proc univariate, highest and lowest observations

Extreme Observations			
Lowest		Highest	
Value	Obs	Value	Obs
20	1	63	96
23	2	64	97
24	3	64	98
25	5	65	99
25	4	69	100

proc freq, table for chd

```
proc sort data=chdage; by descending chd; run;
proc freq data=chdage order=data;
        tables chd a a*chd;
run;
```

chd				
chd	Frequency	Percent	Cumulative Frequency	Cumulative Percent
1	43	43.00	43	43.00
0	57	57.00	100	100.00

proc freq, table for chd and a

```
proc sort data=chdage;
        by descending chd;
run;
proc freq data=chdage order=data;
        tables chd a;
run;
```

proc freq, table for chd and a

chd				
chd	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	57	57.00	57	57.00
1	43	43.00	100	100.00

a	Frequency	Percent	Cumulative Frequency	Cumulative Percent
1	16	16.00	16	16.00
2	23	23.00	39	39.00
3	28	28.00	67	67.00
4	33	33.00	100	100.00

proc freq, table for chd by a

```
proc sort data=chdage;
        by descending chd;
run;
proc freq data=chdage order=data;
        tables chd a a*chd;
run;
```

proc freq, table for chd by a

Frequency
Percent
Row Pct
Col Pct

Table of a by chd

a	chd(chd) 1	0	Total
1	2	14	16
	2.00	14.00	16.00
	12.50	87.50	
	4.65	24.56	
2	5	18	23
	5.00	18.00	23.00
	21.74	78.26	
	11.63	31.58	
3	11	17	28
	11.00	17.00	28.00
	39.29	60.71	
	25.58	29.82	
4	25	8	33
	25.00	8.00	33.00
	75.76	24.24	
	58.14	14.04	
Total	43	57	100
	43.00	57.00	100.00

proc freq, table for chd, a and chd by a

```
proc sort data=chdage;
        by descending chd;
run;

proc freq data=chdage order=data;
 tables chd a a*chd;
run;
```

proc logistic, basic information

```
proc logistic descending data=chdage;
  model chd=age;
run;
```

Model Information		
Data Set	WORK.CHDAGE	
Response Variable	chd	chd
Number of Response Levels	2	
Model	binary logit	
Optimization Technique	Fisher's scoring	

proc logistic, # of observations, response profile and prob. modeled

Number of Observations Read	100
Number of Observations Used	100

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

Probability modeled is chd=1.

proc logistic, convergence status and fit statistics

Model Convergence Status
Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	138.663	111.353
SC	141.268	116.563
-2 Log L	136.663	107.353

proc logistic, global tests

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	29.3099	1	<.0001
Score	26.3989	1	<.0001
Wald	21.2541	1	<.0001

proc logistic, estimated coefficients

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-5.3095	1.1337	21.9350	<.0001
age	1	0.1109	0.0241	21.2541	<.0001

proc logistic, estimated odds ratios

Odds Ratio Estimates			
Effect	Point Estimate	95% Wald Confidence Limits	
age	1.117	1.066	1.171

proc logistic, observed/predicted

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	79.0	Somers' D	0.600
Percent Discordant	19.0	Gamma	0.612
Percent Tied	2.0	Tau-a	0.297
Pairs	2451	c	0.800

proc gplot, preparation

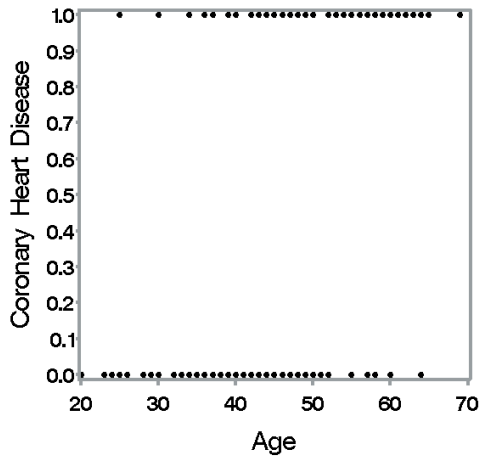
```
axis1 minor=none label=(f=swiss h=2.5 'Age');

axis2 minor=none label=(f=swiss h=2.5 a=90 'Coronary
                                    Heart Disease');
goptions FTEXT=swissb HTEXT=2.0 HSIZE=6 in
         VSIZE=6 in;
symbol1 c=black v=dot;
```

proc gplot

```
proc gplot data=chdage;
        plot chd*age/haxis=axis1 vaxis=axis2;
run; quit;
```

proc gplot



Saving SAS output in a data set

- There are two options
  - Output statements
  - ODS
- Here, we'll discuss output statements
- We may discuss ODS later in the semester

Example 1

- Save the probabilities, $\hat{\pi}$, predicted by the logistic regression model
- Add them to an existing graph

Example 1, saving the probabilities

```
proc logistic descending data=chdage;
  model chd=age;
  output out=pdat p=pihat;
run;
```

Variable name

Data set name

```
proc print data=pdat;
   var age CHD pihat;
run;
```

Obs	age	chd	pihat
1	20	0	0.04348
2	23	0	0.05962
Etc.			

Example 1, adding the probabilities to an existing graph, preparation

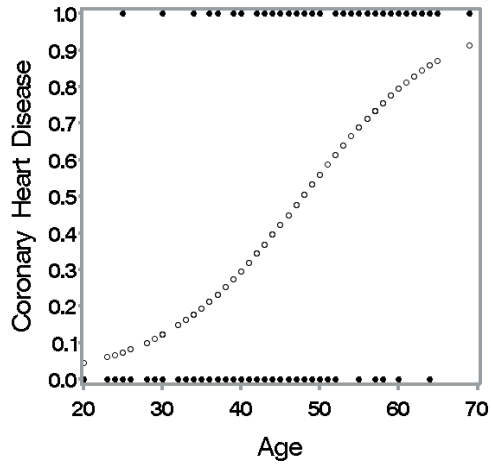
```
axis1 minor=none label=(f=swiss h=2.5 'Age');

axis2 minor=none label=(f=swiss h=2.5 a=90 'Coronary
                                    Heart Disease');
goptions FTEXT=swissb HTEXT=2.0 HSIZE=6 in
                VSIZE=6 in;
symbol1 c=black v=dot;
symbol2 c=black v=circle;
```

Example 1, adding the probabilities to an existing graph, proc gplot

```
proc gplot data=pdat;
  plot (chd pihat)*age/overlay haxis=axis1 vaxis=axis2;
run; quit;
```

Example 1 cont.



Logistic regression predicts the proportion with CHD for each age

Example 2

- Create 4 age categories
- Determine the proportion with CHD in each age category
- Save the proportions in a data set
- Add the proportions to an existing graph

(Hopefully this example will help you better understand pihat ($\hat{\pi}$))

Example 2, create 4 age categories

```
libname sdat 'C:\ERHS642';

data chdage; set sdat.chdage;
* Create a categorical age variable with 4 categories *;
        if 20<=age<=30 then a=1;
  else if 30< age<=40 then a=2;
  else if 40< age<=50 then a=3;
  else if 50< age<=70 then a=4;
run;
```

Example 2, determine the proportion with CHD in each age category

Note: For a 0/1 variable like CHD, the mean equals the proportion with CHD

```
proc sort data=chdage; by a; run;

proc means noprint mean data=chdage;
  by a;
  var chd;
  output out=s_means mean=proportion;
run;
```

Contains 4 proportions (proportion with CHD in each age category) but no actual ages

Data set name

Variable name

Example 2, add ages to saved data set

```
data s_means;
  set s_means;
        if a=1 then age=25;
  else if a=2 then age=35;
  else if a=3 then age=45;
  else if a=4 then age=60;
  drop _type_ _freq_ a;
run;
proc print data=s_means; run;
```

Age interval midpoints

Nuisance variables

Obs	proportion	age
1	0.12500	25
2	0.21739	35
3	0.39286	45
4	0.75758	60

Example 2, merge new data set with data set containing pihat values

```
data plotdat;
  merge pdat s_means;
  by age;
run;

proc print data=plotdat;
  var id age chd pihat proportion;
run;
```

Example 2, add the proportions to the graph, preparation

```
axis1 minor=none label=(f=swiss h=2.5 'Age');

axis2 minor=none label=(f=swiss h=2.5 a=90 'Coronary
                                    Heart Disease');
goptions FTEXT=swissb HTEXT=2.0 HSIZE=6 in
                VSIZE=6 in;
symbol1 c=black v=dot;
symbol2 c=black v=circle;
symbol3 c=black v=star h=2;
```

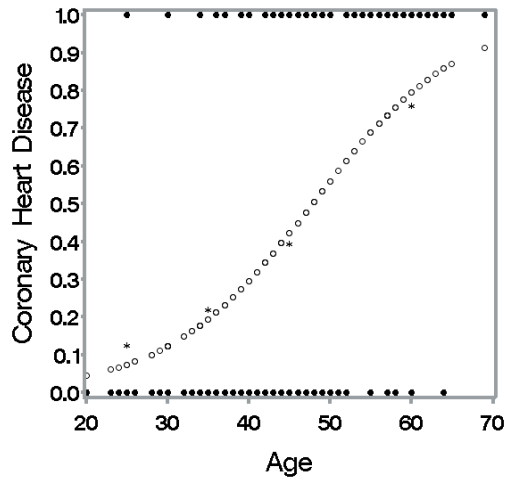
Example, add the proportions to the graph

```
proc gplot data=plotdat;

  plot (chd pihat prop)*age
          /overlay haxis=axis1 vaxis=axis2;

run; quit;
```

Example, add the proportions to the graph



- Logistic regression predicts the probability ($\hat{\pi}$) of CHD for each age
- We can approximate this by calculating the proportion with CHD in different age categories

Complete program

```
libname sdat 'C:\ERHS642';

data chdage; set sdat.chdage;
  * Create a categorical age variable with 4
categories *;
        if 20<=age<=30 then a=1;
  else if 30< age<=40 then a=2;
  else if 40< age<=50 then a=3;
  else if 50< age<=70 then a=4;
```

Complete program, cont.

```
* Create the square root of age *;
 sqrt_age=sqrt(age);

 * Create the natural log of age *;
 ln_age = log(age);

 * Create age squared *;
 age_sq=age**2;
  run;
```

Complete program, cont.

```
proc print data=chdage; run;

proc means data=chdage; var age; run;

proc means data=chdage
        n mean min median max std; var age; run;

proc univariate data=chdage; var age; run;
```

Complete program, cont.

```
proc freq data=chdage; tables chd a a*chd; run;

proc logistic descending data=chdage;
  model chd=age;
run;

proc logistic descending data=chdage;
  model chd=age;
  output out=pdat p=pihat;
run;

proc print data=pdat; var age CHD pihat; run;
```

Complete program, cont.

```
proc sort data=chdage; by a; run;

proc means noprint mean data=chdage;
  by a;
  var chd;
  output out=s_means mean=proportion; run;

proc print data=s_means; run;
```

Complete program, cont.

```
data s_means;   set s_means;
          if a=1 then age=25;
  else if a=2 then age=35;
  else if a=3 then age=45;
  else if a=4 then age=60;
  drop _type_ _freq_ a;
run;

proc print data=s_means; run;
data plotdat; merge pdat s_means; by age; run;
proc print data=plotdat; var id age chd pihat prop; run;
```

Complete program, cont.

```
axis1 minor=none label=(f=swiss h=2.5 'Age');
axis2 minor=none label=(f=swiss h=2.5 a=90
                        'Coronary Heart Disease');

goptions FTEXT=swissb HTEXT=2.0 HSIZE=6 in
VSIZE=6 in;

symbol1 c=black v=dot;
symbol2 c=black v=circle;
symbol3 c=black v=star h=2;
```

Complete program, cont.

```
proc gplot data=plotdat;

  plot (chd pihat prop)*age
        /overlay haxis=axis1 vaxis=axis2;

run; quit;
```