Basic Statistical and Modeling Procedures Using SAS

One-Sample Tests

The statistical procedures illustrated in this handout use two datasets. The first, Pulse, has information collected in a classroom setting, where students were asked to take their pulse two times. Half the class was asked to run in place between the two readings and the other group was asked to stay seated between the two readings. The raw data for this study are contained in a file called pulse.csv. The other dataset we use is a dataset called Employee.sas7bdat. It is a SAS dataset that contains information about salaries in a mythical company.

Read in the pulse data and create a temporary SAS dataset for the examples:

data pulse;
  infile "pulse.csv" firstobs=2 delimiter="," missover;
  input pulse1 pulse2 ran smokes sex height weight activity;
  label pulse1 = "Resting pulse, rate per minute"
    pulse2 = "Second pulse, rate per minute";
run;

Create and assign formats to variables:

proc format;
  value sexfmt 1="Male" 2="Female";
  value yesnofmt 1="Yes" 2="No";
  value actfmt 1="Low" 2="Medium" 3="High";
run;

proc print data=pulse (obs=25) label;
  format sex sexfmt. ran smokes yesnofmt. activity actfmt.;
run;

Descriptive Statistics:

proc means data=pulse;
  run;

<table>
<thead>
<tr>
<th>Variable</th>
<th>Label</th>
<th>N</th>
<th>Mean</th>
<th>Std Dev</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>pulse1</td>
<td>Resting pulse, rate per minute</td>
<td>92</td>
<td>72.8695652</td>
<td>11.0087052</td>
<td>48.0000000</td>
<td>100.0000000</td>
</tr>
<tr>
<td>pulse2</td>
<td>Second pulse, rate per minute</td>
<td>92</td>
<td>80.00000000</td>
<td>17.0937943</td>
<td>50.0000000</td>
<td>140.0000000</td>
</tr>
<tr>
<td>ran</td>
<td></td>
<td>92</td>
<td>1.6195652</td>
<td>0.4881540</td>
<td>1.0000000</td>
<td>2.0000000</td>
</tr>
<tr>
<td>smokes</td>
<td></td>
<td>92</td>
<td>1.6956522</td>
<td>0.4626519</td>
<td>1.0000000</td>
<td>2.0000000</td>
</tr>
<tr>
<td>sex</td>
<td></td>
<td>92</td>
<td>1.3804348</td>
<td>0.4881540</td>
<td>1.0000000</td>
<td>2.0000000</td>
</tr>
<tr>
<td>height</td>
<td></td>
<td>92</td>
<td>68.7391304</td>
<td>3.6520943</td>
<td>61.0000000</td>
<td>75.0000000</td>
</tr>
<tr>
<td>weight</td>
<td></td>
<td>92</td>
<td>145.1521739</td>
<td>23.7393978</td>
<td>95.0000000</td>
<td>215.0000000</td>
</tr>
<tr>
<td>activity</td>
<td></td>
<td>92</td>
<td>2.1195652</td>
<td>0.5711448</td>
<td>1.0000000</td>
<td>3.0000000</td>
</tr>
</tbody>
</table>
Binomial Confidence Intervals and Tests for Binary Variables:

If you have a categorical variable with only two levels, you can use the `binomial` option to request a 95% confidence interval for the proportion in the first level of the variable. In the PULSE data set, SMOKES=1 indicates those who were smokers, and SMOKES=2 indicates non-smokers. Use the `(p=)` option to specify the null hypothesis proportion that you wish to test for the first level of the variable. In the commands below, we test hypotheses for the proportion of SMOKES=1 (i.e., proportion of smokers) in the population. By default SAS produces an asymptotic test of the null hypothesis:

\[ H_0: \text{proportion of smokers} = 0.25 \]
\[ H_A: \text{proportion of smokers} \neq 0.25 \]

```sas
proc freq data = pulse;
    tables  smokes / binomial(p=.25);
run;
```

```
smokes
Cumulative    Cumulative
smokes    Frequency     Percent     Frequency      Percent
-----------------------------------------------------------
 1          28       30.43            28        30.43
 2          64       69.57            92       100.00
```

Binomial Proportion for smokes = 1
-------------------------------
Proportion                0.3043
ASE                       0.0480
95% Lower Conf Limit      0.2103
95% Upper Conf Limit      0.3984

Exact Conf Limits
95% Lower Conf Limit      0.2127
95% Upper Conf Limit      0.4090

Test of HO: Proportion = 0.25

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ASE under HO</td>
<td>0.0451</td>
</tr>
<tr>
<td>Z</td>
<td>1.2039</td>
</tr>
<tr>
<td>One-sided Pr &gt; Z</td>
<td>0.1143</td>
</tr>
<tr>
<td>Two-sided Pr &gt;</td>
<td>Z</td>
</tr>
</tbody>
</table>

Sample Size = 92

If you wish to obtain an exact binomial test of the null hypothesis, use the `exact` statement. If you include the `mc` option for large datasets, you will get a Monte Carlo p-value.

```sas
proc freq data = pulse;
    tables  smokes / binomial(p=.25);
    exact binomial / mc;
run;
```
This results in an exact test of the null hypothesis, in addition to the default asymptotic test, the exact test results for both a one-sided and two-sided alternative hypothesis are shown.

```
Binomial Proportion for smokes = 1
-----------------------------------
Proportion (P)               0.3043
ASE                          0.0480
95% Lower Conf Limit         0.2103
95% Upper Conf Limit         0.3984

Exact Conf Limits
95% Lower Conf Limit         0.2127
95% Upper Conf Limit         0.4090

Test of H0: Proportion = 0.25
ASE under HO                 0.0451
Z                             1.2039
One-sided Pr > Z             0.1143
Two-sided Pr > |Z|            0.2286

Exact Test
One-sided Pr >= P            0.1399
Two-sided = 2 * One-sided    0.2797
Sample Size = 92
```

**Chi-square Goodness of Fit Tests for Categorical Variables:**

Use the chisq option in the tables statement to get a chi-square goodness of fit test, which can be used for categorical variables with two or more levels. By default SAS assumes that you wish to test the null hypothesis that the proportion of cases is equal in all categories. In the variable `ACTIVITY`, a value of 1 indicates a low level of activity, a value of 2 is a medium level of activity, and a value of 3 indicates a high level of activity.

```
proc freq data = pulse;
  tables activity / chisq;
run;
```

```
activity
Cumulative
activity    Frequency     Percent      Frequency      Percent
-------------------------------------------------------
1           10       10.87            10        10.87
2           61       66.30            71        77.17
3           21       22.83            92       100.00

Chi-Square Test
for Equal Proportions
---------------------
Chi-Square    46.9783
DF             2
Pr > ChiSq     <.0001

Sample Size = 92
```
If you wish to specify your own proportions, use the testp = option in the tables statement. This option allows you to specify any proportions that you wish to test for each level of the variable in the tables statement, as long as the sum of the proportions equals 1.0. In the example below we test the null hypothesis:

\[ H_0: P_1 = 0.20, P_2 = 0.50, P_3 = 0.30 \]

```sas
proc freq data = pulse;
   tables activity /chisq testp = ( .20 , .50, .30 );
run;
```

The FREQ Procedure

<table>
<thead>
<tr>
<th>activity</th>
<th>Frequency</th>
<th>Percent</th>
<th>Test</th>
<th>Cumulative Frequency</th>
<th>Cumulative Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>10.87</td>
<td>20.00</td>
<td>10</td>
<td>10.87</td>
</tr>
<tr>
<td>2</td>
<td>61</td>
<td>66.30</td>
<td>50.00</td>
<td>71</td>
<td>77.17</td>
</tr>
<tr>
<td>3</td>
<td>21</td>
<td>22.83</td>
<td>30.00</td>
<td>92</td>
<td>100.00</td>
</tr>
</tbody>
</table>

Chi-Square Test for Specified Proportions

- Chi-Square: 10.3043
- DF: 2
- Pr > ChiSq: 0.0058

Sample Size = 92

You may also specify percentages to test, rather than proportions, as long as they add up to 100 percent:

```sas
proc freq data = pulse;
   tables activity /chisq testp = ( 20 , 50, 30 );
run;
```

One-Sample test for a continuous variable:

You can use Proc Univariate to carry out a one-sample t-test to test the population mean against any null hypothesis value you specify by using mu0= option. The default, if no value of mu0 is specified is that mu0 = 0. In the commands below, we test:

\[ H_0: \mu_0 = 72 \]
\[ H_A: \mu_0 \neq 72 \]

Note that SAS also provides the non-parametric Sign test and Wilcoxon signed rank test.
proc univariate data=pulse mu0=72;
  var pulse1;
  histogram / normal (mu=est sigma=est);
  qqplot /normal (mu=est sigma=est);
run;

Selected output from Proc Univariate:

<table>
<thead>
<tr>
<th>Test</th>
<th>Statistic</th>
<th>p Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student's t</td>
<td>0.757635</td>
<td>0.4506</td>
</tr>
<tr>
<td>Sign</td>
<td>-3</td>
<td>0.5900</td>
</tr>
<tr>
<td>Signed Rank</td>
<td>96.5</td>
<td>0.6797</td>
</tr>
</tbody>
</table>

Equivalently, we can carry out a one-sample t-test in Proc Ttest by specifying the H0= option:

proc ttest data=pulse H0=72 ;
  var pulse1;
run;

Variable: pulse1 (Resting pulse, rate per minute)

<table>
<thead>
<tr>
<th>N</th>
<th>Mean</th>
<th>Std Dev</th>
<th>Std Err</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>92</td>
<td>72.8696</td>
<td>11.0087</td>
<td>1.1477</td>
<td>48.0000</td>
<td>100.0</td>
</tr>
</tbody>
</table>

Mean           95% CL Mean    Std Dev  95% CL Std Dev
72.8696         70.5897     75.1494     11.0087  9.6155   12.8779

DF  t Value    Pr > |t|
91  0.76       0.4506

Paired Samples t-test:

If you wish to compare the means of two variables that are paired (i.e. correlated), you can use a paired sample t-test for continuous variables. To do this use Proc ttest with a paired statement, to get a paired samples t-test:

proc ttest data=pulse;
  paired pulse2*pulse1;
run;
The paired t-test can be carried out for each level of RAN. The commands and results of these commands are shown below:

```plaintext
proc sort data=pulse;
  by ran;
run;
proc ttest data=pulse;
  paired pulse2*pulse1;
  by ran;
run;
```

---

**ran=1**

---

The paired t-test can be carried out for each level of RAN. The commands and results of these commands are shown below:

```plaintext
proc sort data=pulse;
  by ran;
run;
proc ttest data=pulse;
  paired pulse2*pulse1;
  by ran;
run;
```

---

**ran=2**

---

The paired t-test can be carried out for each level of RAN. The commands and results of these commands are shown below:

```plaintext
proc sort data=pulse;
  by ran;
run;
proc ttest data=pulse;
  paired pulse2*pulse1;
  by ran;
run;
```
Independent samples t-tests

An independent samples t-test can be used to compare the means in two independent groups of observations.:

```sas
proc ttest data=sasdata2.employee2;
class gender;
var salary;
run;
```

The output from this procedure is shown below:

```
The TTEST Procedure
Variable:  salary  (Current Salary)
gender           N        Mean     Std Dev     Std Err     Minimum     Maximum
f              216     26031.9      7558.0       514.3     15750.0     58125.0
m              258     41441.8     19499.2      1214.0     19650.0      135000
Diff (1-2)   -15409.9     15265.9      1407.9

gender       Method              Mean      95% CL Mean       Std Dev     95% CL Std Dev
f                             26031.9    25018.3  27045.6     7558.0     6906.2   8346.8
m                             41441.8    39051.2  43832.4    19499.2    17949.3  21344.3
Diff (1-2)   Pooled             -15409.9    -18176.4 -12643.3    15265.9    14351.1  16306.1
Diff (1-2)   Satterthwaite     -15409.9    -18003.0 -12816.7

Method          Variances        DF    t Value    Pr > |t|
Pooled           Equal           472     -10.95    <.0001
Satterthwaite    Unequal      344.26   -11.69    <.0001

Equality of Variances

Method          Num DF    Den DF    F Value    Pr > F
Folded F               257     215       6.66    <.0001
```

If you want to check on the distribution of Salary for males and females, you can use Proc Univariate.

```sas
proc univariate data=sasdata2.employee2;
var salary;
class gender;
histogram;
run;
```
Because it looks like salary is highly skewed, you might want to use a log transformation of salary to compare the two genders. Proc ttest has the `dist=lognormal` option to accomplish this:

```sas
proc ttest data=sasdata2.employee2 dist=lognormal;
   class gender;
   var salary ;
run;
```

The output from this procedure shows that the geometric mean and coefficient of variation are reported, rather than the arithmetic mean and standard deviation.
To get an independent samples t-test within each job category, use a BY statement, after sorting by jobcat.

```sas
proc sort data=sasdata2.employee;
    by jobcat;
run;
proc ttest data=sasdata2.employee;
    by jobcat;
    class gender;
    var salary;
run;
```

**Wilcoxon rank sum test:**

If you are unwilling to assume normality for your continuous test variable or the sample size is too small for you to appeal to the central-limit-theorem, you may want to use non-parametric tests. The Wilcoxon rank sum test (also known as the Mann-Whitney test) is the non-parametric analog of the independent sample t test.

```sas
/*NON-PARAMETRIC TEST: WILCOXON/MANN-WHITNEY TEST*/
proc npar1way data=sasdata2.employee wilcoxon;
    class gender;
    var salary;
run;
```

A Monte-Carlo approximation of the exact p-value can be obtained for the Wilcoxon test by using an exact statement, as shown below:

```sas
proc npar1way data=sasdata2.employee wilcoxon;
    class gender;
    var salary;
    exact wilcoxon / mc;
run;
```

**Correlation**

Proc corr can be used to calculate correlations for several variables:

```sas
proc corr data=sasdata2.employee;
    var salary salbegin educ;
run;
```

The CORR Procedure

3 Variables:  salary  salbegin  educ

<table>
<thead>
<tr>
<th>Variable</th>
<th>N</th>
<th>Mean</th>
<th>Std Dev</th>
<th>Sum</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>salary</td>
<td>474</td>
<td>34420</td>
<td>17076</td>
<td>16314875</td>
<td>15750</td>
<td>135000</td>
</tr>
<tr>
<td>salbegin</td>
<td>474</td>
<td>17016</td>
<td>7871</td>
<td>8065625</td>
<td>9000</td>
<td>79980</td>
</tr>
<tr>
<td>educ</td>
<td>474</td>
<td>13.49156</td>
<td>2.88485</td>
<td>6395</td>
<td>8.00000</td>
<td>21.00000</td>
</tr>
</tbody>
</table>
Prob > |r| under H0: Rho=0

<table>
<thead>
<tr>
<th></th>
<th>salary</th>
<th>salbegin</th>
<th>educ</th>
</tr>
</thead>
<tbody>
<tr>
<td>salary</td>
<td>1.00000</td>
<td>0.88012</td>
<td>0.66056</td>
</tr>
<tr>
<td>Current Salary</td>
<td>&lt;0.0001</td>
<td>&lt;0.0001</td>
<td></td>
</tr>
<tr>
<td>salbegin</td>
<td>0.88012</td>
<td>1.00000</td>
<td>0.63320</td>
</tr>
<tr>
<td>Beginning Salary</td>
<td>&lt;0.0001</td>
<td>&lt;0.0001</td>
<td></td>
</tr>
<tr>
<td>educ</td>
<td>0.66056</td>
<td>0.63320</td>
<td>1.00000</td>
</tr>
<tr>
<td>Educational Level (years)</td>
<td>&lt;0.0001</td>
<td>&lt;0.0001</td>
<td></td>
</tr>
</tbody>
</table>

**Linear regression**

You can fit a linear regression model using Proc Reg:

```latex
ods graphics on;
proc reg data=sasdata2.employee2;
  model salary = salbegin educ jobdum2 jobdum3 prevexp female;
run; quit;
ods graphics off;
```

Note that the output dataset that we created, REGDAT, has all the original observations and variables in it, plus the new variables Predict, Resid, and Rstudent.

Output from the linear regression model is shown below:

```
The REG Procedure
Model: MODEL1
Dependent Variable: salary Current Salary
```

```
Number of Observations Read 474
Number of Observations Used 474
```

```
Analysis of Variance

<table>
<thead>
<tr>
<th>Source</th>
<th>DF</th>
<th>Squares</th>
<th>Mean Square</th>
<th>F Value</th>
<th>Pr &gt; F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>6</td>
<td>1.15239E11</td>
<td>19206503793</td>
<td>395.52</td>
<td>&lt;0.0001</td>
</tr>
<tr>
<td>Error</td>
<td>467</td>
<td>22677472676</td>
<td>48559899</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Corrected Total</td>
<td>473</td>
<td>1.379165E11</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

```
Root MSE 6968.49328 R-Square 0.8356
Dependent Mean 34420 Adj R-Sq 0.8335
Coeff Var 20.24573
```

```
Parameter Estimates
```

| Variable   | Label                        | DF | Estimate   | Error  | t Value | Pr > |t| |
|------------|------------------------------|----|------------|--------|---------|-------|------|
| Intercept  | Intercept                    | 1  | 5333.10875 | 2337.45787 | 2.28  | 0.0230 |
| salbegin   | Beginning Salary             | 1  | 1.31359    | 0.07433 | 17.67 | <0.0001|
| educ       | Educational Level (years)    | 1  | 548.90277  | 163.27562 | 3.36  | 0.0008 |
| jobdum2    |                              | 1  | 6764.00748 | 1666.58592 | 4.06  | <0.0001|
| jobdum3    |                              | 1  | 11389      | 1394.92854 | 8.16  | <0.0001|
| prevexp    | Previous Experience (months) | 1  | -21.98825  | 3.64720  | -6.03 | <0.0001|
| female     |                              | 1  | -2122.17197| 775.86768 | -2.74 | 0.0065 |
We note that the distribution of the residuals is highly skewed. This is an indication that we may want to use a transformation of the dependent variable.

The variance of the residuals is highly heteroskedastic; we note that there is much more variability of residuals for large predicted values, making a megaphone-like appearance in the graph.

We may want to transform salary using the natural log. The commands below show how Logsalary can be created to be used in the regression. Note that to create a new variable, we need
to use a data step. Submit these commands and check the residuals from this new regression model.

```sas
data temp;
  set sasdata2.employee2;
  logsalary = log(salary);
run;

ods graphics on;
proc reg data=temp;
  model logsalary = salbegin educ jobdum2 jobdum3 prevexp female;
  output out=regdat2 p=predict r=resid rstudent=rstudent;
run; quit;
ods graphics off;
```

The REG Procedure
Model: MODEL1
Dependent Variable: logsalary

<table>
<thead>
<tr>
<th>Source</th>
<th>DF</th>
<th>Sum of Squares</th>
<th>Mean Square</th>
<th>F Value</th>
<th>Pr &gt; F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>6</td>
<td>61.64142</td>
<td>10.27357</td>
<td>368.12</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>Error</td>
<td>467</td>
<td>13.03320</td>
<td>0.02791</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Corrected Total</td>
<td>473</td>
<td>74.67462</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Root MSE 0.16706
Dependent Mean 10.35679
R-Square 0.8255
Adj R-Sq 0.8232
Coef Var 1.61303

Parameter Estimates

| Variable     | Label                  | DF  | Parameter Estimate | Standard Error | t Value | Pr > |t| |
|--------------|------------------------|-----|--------------------|----------------|---------|------|---|
| Intercept    | Intercept              | 1   | 9.66675            | 0.05604        | 172.51  | <.0001|
| salbegin     | Beginning Salary       | 1   | 0.00002304         | 0.00000178     | 12.93   | <.0001|
| educ         | Educational Level (years) | 1 | 0.02592            | 0.00391        | 6.62    | <.0001|
| jobdum2      |                        | 1   | 0.24880            | 0.03995        | 6.23    | <.0001|
| jobdum3      |                        | 1   | 0.28225            | 0.03344        | 8.44    | <.0001|
| prevexp      | Previous Experience (months) | 1 | -0.00063575        | 0.00008744     | -7.27   | <.0001|
| female       |                        | 1   | -0.12070           | 0.01860        | -6.49   | <.0001|

The distribution of the residuals appears to be much more normal after the log transformation was applied.

The variance of the residuals appears to be much more constant across all predicted values after applying the log transformation to the dependent variable. We no longer appear to have heteroskedasticity of the residuals.
Cross-tabulations

You can carry out a Pearson Chi-square test of independence using Proc Freq. This procedure is extremely versatile and flexible, and has many options available.

```sas
proc freq data=sasdata2.employee2;
   tables gender*jobcat / chisq;
run
```
The FREQ Procedure

Table of gender by jobcat

<table>
<thead>
<tr>
<th>Sex</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>363</td>
<td>27</td>
<td>84</td>
<td>474</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Category</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>206</td>
<td>0</td>
<td>10</td>
<td>216</td>
</tr>
<tr>
<td></td>
<td>43.46</td>
<td>0.00</td>
<td>2.11</td>
<td>45.57</td>
</tr>
<tr>
<td></td>
<td>95.37</td>
<td>0.00</td>
<td>4.63</td>
<td></td>
</tr>
<tr>
<td></td>
<td>56.75</td>
<td>0.00</td>
<td>11.90</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Category</th>
<th>157</th>
<th>27</th>
<th>74</th>
<th>258</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>33.12</td>
<td>5.70</td>
<td>15.61</td>
<td>54.43</td>
</tr>
<tr>
<td></td>
<td>60.85</td>
<td>10.47</td>
<td>28.68</td>
<td></td>
</tr>
<tr>
<td></td>
<td>43.25</td>
<td>100.00</td>
<td>88.10</td>
<td></td>
</tr>
</tbody>
</table>

You can get an exact test for this by using an Exact statement. In this case, we requested Fisher’s exact test, but exact p-values for other statistics can be requested:

```sas
proc freq data=sasdata2.employee;
tables gender*jobcat / chisq;
extact fisher;
run;
```

In the output below, be sure to read the last p-value at the bottom of the output for Fisher’s exact test.

```
Fisher's Exact Test

Table Probability (P)  2.854E-22
Pr <= P  5.756E-21

Sample Size = 474
```

If your problem is large, you may wish to get a Monte Carlo simulation for the p-value, based on 10,000 tables. To do this use the following syntax. Seed=0 will use a random seed for the process based on the clock time when you run the procedure.
```sas
proc freq data=sasdata2.employee;
  tables gender*jobcat / chisq;
  exact fisher / mc seed=0;
run;
```

Partial output from this procedure is shown below:

```
The FREQ Procedure

Statistics for Table of gender by jobcat

Fisher's Exact Test

----------------------------------
Table Probability (P)    2.854E-22

Monte Carlo Estimate for the Exact Test

Pr <= P                     0.0000
99% Lower Conf Limit        0.0000
99% Upper Conf Limit        4.604E-04
Number of Samples            10000
Initial Seed             445615001

Sample Size = 474
```

Each time the procedure is run using this syntax, you will get different answers. If you wish to get the same result, simply use the Initial Seed value reported by SAS in the output in your Exact statement.

```sas
proc freq data=sasdata2.employee;
  tables gender*jobcat / chisq;
  exact fisher / mc seed=445615001;
run;
```

**McNemar’s test for paired categorical data:**

If you wish to compare the proportions in a 2 by 2 table for paired data, you can use McNemar’s test, by specifying the **agree** option in Proc Freq. Before running the McNemar’s test, we recode PULSE1 and PULSE2 into two categorical variables HIPULSE1 and HIPULSE2, as shown below:

```sas
data newpulse;
  set pulse;
  if pulse1 > 80 then hipulse1 = 1;
  if pulse1 > 0 and pulse1 <= 89 then hipulse1 = 0;
  if pulse2 > 80 then hipulse2 = 1;
  if pulse2 > 0 and pulse2 <= 89 then hipulse2 = 0;
run;
```
We can now carry out McNemar’s test of symmetry to see if the proportion of participants with a high value of PULSE1 is different than the proportion of participants with a high value of PULSE2.

```
proc freq data=newpulse;
  tables hipulse1 * hipulse2 / agree;
run;
```

Table of hipulse1 by hipulse2

```
                  hipulse1
          Frequency| hipulse2
          Percent |     0|     1| Total
Row Pct  |       0|     1|  Total
Col Pct  |  71   |  21  |  92    
---------+----------+--------+---------
0       |    69 |    13 |    82  |
        |  75.00|  14.13|  89.13 |
        |  84.15|  15.85|       |
        |  97.18|  61.90|       |
---------+----------+--------+---------
1       |      2 |     8 |     10|
        |  2.17 |  8.70 |  10.87 |
        |  20.00|  80.00|       |
        |  2.82 | 38.10 |       |
---------+----------+--------+---------
Total   |  71   |  21   |  92    
        |  77.17| 22.83 | 100.00 
```

Statistics for Table of hipulse1 by hipulse2

```
McNemar's Test
------------------
Statistic (S) 8.0667
DF              1
Pr > S          0.0045
Sample Size = 92
```
Logistic regression

If the outcome is coded as 0,1 and you wish to predict the probability of a 1, use the descending option for Proc Logistic.

data afifi;
  set sasdata2.afifi;
  if survive=3 then died=1;
  if survive=1 then died=0;
run;

proc logistic data=afifi descending;
  model died = map1 shockdum sex / risklimits;
  units map1 = 1 10 shockdum = 1 sex=1;
run;

Data Set WORK.AFIFI
Response Variable died
Number of Response Levels 2
Model binary logit
Optimization Technique Fisher's scoring

Number of Observations Read 113
Number of Observations Used 113

Response Profile

<table>
<thead>
<tr>
<th>Ordered Value</th>
<th>Total died</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>43</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>70</td>
</tr>
</tbody>
</table>

Probability modeled is died=1.

Model Convergence Status

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

Model Fit Statistics

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Intercept Only</th>
<th>Intercept and Covariates</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIC</td>
<td>152.137</td>
<td>127.874</td>
</tr>
<tr>
<td>SC</td>
<td>154.864</td>
<td>138.784</td>
</tr>
<tr>
<td>-2 Log L</td>
<td>150.137</td>
<td>119.874</td>
</tr>
</tbody>
</table>

Testing Global Null Hypothesis: BETA=0

<table>
<thead>
<tr>
<th>Test</th>
<th>Chi-Square</th>
<th>DF</th>
<th>Pr &gt; ChiSq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Likelihood Ratio</td>
<td>30.2028</td>
<td>3</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>Score</td>
<td>26.1922</td>
<td>3</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>Wald</td>
<td>20.3328</td>
<td>3</td>
<td>0.0001</td>
</tr>
</tbody>
</table>
Analysis of Maximum Likelihood Estimates

<table>
<thead>
<tr>
<th>Parameter</th>
<th>DF</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>Chi-Square</th>
<th>Pr &gt; ChiSq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1</td>
<td>-0.9571</td>
<td>1.2827</td>
<td>0.5568</td>
<td>0.4556</td>
</tr>
<tr>
<td>MAP1</td>
<td>1</td>
<td>-0.0285</td>
<td>0.0114</td>
<td>6.220</td>
<td>0.0126</td>
</tr>
<tr>
<td>SHOCKDUM</td>
<td>1</td>
<td>1.8999</td>
<td>0.6694</td>
<td>8.0540</td>
<td>0.0045</td>
</tr>
<tr>
<td>SEX</td>
<td>1</td>
<td>0.6760</td>
<td>0.4450</td>
<td>2.3082</td>
<td>0.1287</td>
</tr>
</tbody>
</table>

Association of Predicted Probabilities and Observed Responses

| Percent Concordant | 79.1 | Somers' D | 0.586 |
| Percent Discordant | 20.5 | Gamma     | 0.588 |
| Percent Tied      | 0.4  | Tau-a     | 0.279 |
| Pairs             | 3010 | c         | 0.793 |

Odds Ratio Estimates and Wald Confidence Intervals

<table>
<thead>
<tr>
<th>Effect</th>
<th>Unit</th>
<th>Estimate</th>
<th>95% Confidence Limits</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAP1</td>
<td>1.0000</td>
<td>0.972</td>
<td>0.950 0.994</td>
</tr>
<tr>
<td>SHOCKDUM</td>
<td>1.0000</td>
<td>6.685</td>
<td>1.800 24.827</td>
</tr>
<tr>
<td>SEX</td>
<td>1.0000</td>
<td>1.966</td>
<td>0.822 4.703</td>
</tr>
</tbody>
</table>

To get graphical output, include the plots = option in the SAS code. We also request odds ratios for a 1-unit and for 10 units increase in MAP1. The oddsratio plot will not be produced unless the risklimits option is specified at the end of the model statement.

```sas
ods graphics on;
proc logistic data=afifi descending PLOTS(ONLY) = (effect oddsratio);
    model died = map1 shockdum sex / risklimits;
    units map1 = 1 10 shockdum = 1 sex=1;
run;
ods graphics off;
```
Generalized Linear Model for Count Data

If the outcome is a count variable, you may want to fit a generalized linear model using Proc Genmod. To use this procedure, you must include an option in the model statement specifying the distribution to use. In this example we are modeling the number of home runs that a major league baseball player will get in a season as a function of his salary. We first use a Poisson regression, in which we specify the log of the number of times at bat as the offset (so that we are really modeling the Poisson rate). In the Poisson distribution, the variance is equal to the mean. If we have an appropriate model, we expect the scaled deviance divided by the degrees of freedom to equal approximately 1.0, which is not the case in this example.

```plaintext
proc genmod data=baseball;
   class league division;
   model no_home = salary / dist=poisson offset=log_atbat;
   estimate "Effect of 100k salary increase" salary 100 / est;
   output out=Pfitdata p=predict resraw=resraw reschi=reschi;
run;
```

The GENMOD Procedure

Model Information

Data Set WORK.BASEBALL
Distribution Poisson
Link Function Log
Dependent Variable no_home
Offset Variable log_atbat

Number of Observations Read 322
Number of Observations Used 263
Missing Values 59

Class Level Information

Class Levels Values
league 2 American National
division 2 East West

Criteria For Assessing Goodness Of Fit

<table>
<thead>
<tr>
<th>Criterion</th>
<th>DF</th>
<th>Value</th>
<th>Value/DF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deviance</td>
<td>261</td>
<td>1187.5050</td>
<td>4.5498</td>
</tr>
<tr>
<td>Scaled Deviance</td>
<td>261</td>
<td>1187.5050</td>
<td>4.5498</td>
</tr>
<tr>
<td>Pearson Chi-Square</td>
<td>261</td>
<td>1074.2680</td>
<td>4.1160</td>
</tr>
<tr>
<td>Scaled Pearson X2</td>
<td>261</td>
<td>1074.2680</td>
<td>4.1160</td>
</tr>
<tr>
<td>Log Likelihood</td>
<td></td>
<td>4853.9066</td>
<td></td>
</tr>
<tr>
<td>Full Log Likelihood</td>
<td></td>
<td>-1110.3858</td>
<td></td>
</tr>
<tr>
<td>AIC (smaller is better)</td>
<td></td>
<td>2224.7716</td>
<td></td>
</tr>
<tr>
<td>AICC (smaller is better)</td>
<td></td>
<td>2224.8178</td>
<td></td>
</tr>
<tr>
<td>BIC (smaller is better)</td>
<td></td>
<td>2231.9159</td>
<td></td>
</tr>
</tbody>
</table>

Algorithm converged.
Analysis Of Maximum Likelihood Parameter Estimates

<table>
<thead>
<tr>
<th>Parameter</th>
<th>DF</th>
<th>Estimate</th>
<th>Error</th>
<th>Limits</th>
<th>Chi-Square</th>
<th>Pr &gt; ChiSq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1</td>
<td>-3.6957</td>
<td>0.0291</td>
<td>-3.7527 - 3.6387</td>
<td>16152.5</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>salary</td>
<td>1</td>
<td>0.0002</td>
<td>0.0000</td>
<td>0.0002 - 0.0003</td>
<td>49.78</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>Scale</td>
<td>0</td>
<td>1.0000</td>
<td>0.0000</td>
<td>1.0000 - 1.0000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

NOTE: The scale parameter was held fixed.

Contrast Estimate Results

<table>
<thead>
<tr>
<th>Label</th>
<th>Mean Estimate</th>
<th>Mean Confidence Limits</th>
<th>L'Beta Estimate</th>
<th>Standard Error</th>
<th>Alpha</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effect of 100k salary increase</td>
<td>1.0244</td>
<td>1.0176 - 1.0313</td>
<td>0.0241</td>
<td>0.0034</td>
<td>0.05</td>
</tr>
</tbody>
</table>

The effect of a 100k increase in salary is estimated to be about a 2.4% increase in home run production (95% CI = 1.8% to 3.1% increase).

We look at the distribution of the raw residuals vs. the predicted value. If the Poisson distribution is appropriate, we expect the spread of the residuals to be a function of the mean (which is approximated by the predicted value). This in fact seems to be true, as seen in the graph below:

```
proc sgplot data=fitdata;
  scatter y=resraw x=predict;
run;
```

Here, we use some SAS code to create groups based on the predicted value (i.e., an approximation to the mean of the conditional distribution). We then look at the distribution of the mean of the predicted value in each interval, and the variance of the raw residuals. We see that the mean of the distribution is in all cases less than the variance of the raw residuals. This is another indication that the Poisson distribution is not the best choice for this problem.
data Pfitdata2;
  set Pfitdata;
  if 0<= predict <5 then group=1;
  if 5<= predict <10 then group=2;
  if 10<= predict <15 then group=3;
  if 15<= predict <20 then group=4;
  if 20<= predict then group=5;
run;
proc format;
  value grpfmt 1="0 to 4.9" 2="5 to 9.9" 3="10 to 14.9"
       4="15 to 19.9" 5="20 to Max";
run;
proc means data=Pfitdata2 n min max mean std var;
  class group;
  var predict resraw;
  format group grpfmt. ;
run;

<table>
<thead>
<tr>
<th>group</th>
<th>N Obs</th>
<th>Variable</th>
<th>N</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Mean</th>
<th>Std Dev</th>
<th>Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 to 4.9</td>
<td>13</td>
<td>predict</td>
<td>13</td>
<td>3.2578916</td>
<td>4.9121568</td>
<td>4.1868213</td>
<td>0.4976284</td>
<td>0.2476340</td>
</tr>
<tr>
<td></td>
<td></td>
<td>resraw</td>
<td>13</td>
<td>-3.6719335</td>
<td>7.9021423</td>
<td>0.6593325</td>
<td>3.4251530</td>
<td>11.7316732</td>
</tr>
<tr>
<td>5 to 9.9</td>
<td>95</td>
<td>predict</td>
<td>95</td>
<td>5.0315488</td>
<td>9.9696055</td>
<td>7.4209071</td>
<td>1.4917644</td>
<td>2.2253609</td>
</tr>
<tr>
<td></td>
<td></td>
<td>resraw</td>
<td>95</td>
<td>-7.9171285</td>
<td>14.8172419</td>
<td>-0.2209071</td>
<td>4.6511984</td>
<td>21.6336467</td>
</tr>
<tr>
<td>10 to 14.9</td>
<td>81</td>
<td>predict</td>
<td>81</td>
<td>10.1005542</td>
<td>14.9827472</td>
<td>12.5494485</td>
<td>1.5378169</td>
<td>2.3648810</td>
</tr>
<tr>
<td></td>
<td></td>
<td>resraw</td>
<td>81</td>
<td>-13.9169213</td>
<td>20.8129702</td>
<td>-0.1912922</td>
<td>8.1655403</td>
<td>66.6760486</td>
</tr>
<tr>
<td>15 to 19.9</td>
<td>60</td>
<td>predict</td>
<td>60</td>
<td>15.0114917</td>
<td>19.9416355</td>
<td>16.9885310</td>
<td>1.4100640</td>
<td>1.9882804</td>
</tr>
<tr>
<td></td>
<td></td>
<td>resraw</td>
<td>60</td>
<td>-15.4846933</td>
<td>20.2883934</td>
<td>-0.0218643</td>
<td>9.1883062</td>
<td>84.4249700</td>
</tr>
<tr>
<td>20 to Max</td>
<td>14</td>
<td>predict</td>
<td>14</td>
<td>20.3345690</td>
<td>27.4575979</td>
<td>22.4834298</td>
<td>2.2746223</td>
<td>5.1739064</td>
</tr>
<tr>
<td></td>
<td></td>
<td>resraw</td>
<td>14</td>
<td>-20.3773828</td>
<td>14.1047560</td>
<td>-0.1262870</td>
<td>9.7491252</td>
<td>95.0454412</td>
</tr>
</tbody>
</table>

We now change the distribution to a negative binomial.

ods graphics on;
proc genmod data=baseball plots = (predicted(clm));
  class league division;
  model no_home = salary / dist=negbin offset=log_atbat;
  output out=nbfitdata p=predict resraw=resraw reschi=reschi;
  estimate "Effect of 100k salary increase" salary 100 / est;
run;
ods graphics off;

Model Information
Data Set WORK.BASEBALL
Distribution Negative Binomial
Link Function Log
Dependent Variable no_home
Offset Variable log_atbat
Number of Observations Read         322
Number of Observations Used         263
Missing Values                       59

Class Level Information

Class       Levels       Values
league      2            American National
            2            East West

division    2

Parameter Information

Parameter    Effect
Prm1         Intercept
Prm2         salary

Criteria For Assessing Goodness Of Fit

Criterion                     DF           Value        Value/DF
Deviance                     261        296.8518          1.1374
Scaled Deviance              261        296.8518          1.1374
Pearson Chi-Square           261        217.5268          0.8334
Scaled Pearson X2            261        217.5268          0.8334
Log Likelihood               5104.3910
Full Log Likelihood          -859.9014
AIC (smaller is better)      1725.8028
AICC (smaller is better)     1725.8954
BIC (smaller is better)      1736.5192

Algorithm converged.

Analysis Of Maximum Likelihood Parameter Estimates

<table>
<thead>
<tr>
<th>Parameter</th>
<th>DF</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>Wald 95% Confidence Limits</th>
<th>Wald Chi-Square</th>
<th>Pr &gt; ChiSq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1</td>
<td>-3.7020</td>
<td>0.0650</td>
<td>-3.8294 -3.5745</td>
<td>3241.64</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>salary</td>
<td>1</td>
<td>0.0003</td>
<td>0.0001</td>
<td>0.0001 0.0004</td>
<td>7.82</td>
<td>0.0052</td>
</tr>
<tr>
<td>Dispersion</td>
<td>1</td>
<td>0.3480</td>
<td>0.0407</td>
<td>0.2768 0.4375</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

NOTE: The negative binomial dispersion parameter was estimated by maximum likelihood.

<table>
<thead>
<tr>
<th>Label</th>
<th>Mean Estimate</th>
<th>Confidence Limits</th>
<th>Mean L'Beta Estimate</th>
<th>Error</th>
<th>Alpha</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effect of 100k salary increase</td>
<td>1.0253</td>
<td>1.0075 1.0434</td>
<td>0.0250 0.0089</td>
<td>0.05</td>
<td></td>
</tr>
</tbody>
</table>
We now see that the scaled deviance divided by df is approximately 1.0, which is an improvement over the previous model.

In this model, the predicted effect of a 100k increase in salary is predicted to be about a 2.5% increase in home run production, with a wider Confidence Interval (CI = 0.75% to 4.3%).

We also look at the predicted values and their respective 95% Confidence intervals. Notice that the smaller residuals have smaller estimated CI, as we expect when fitting this type of model.