

**Lab 2: Two-way tables**  
**STT 422: Summer, 2004**  
**Vince Melfi**

All the tests that are described in Chapter 9 of the text (and much more) are implemented in **proc freq** in SAS. In this lab we'll learn how to use **proc freq** to perform tests in a variety of settings.

## 2.1 Chi-square tests

Exercise 9.31 in the text presents data collected to determine whether contacting people by phone or letter before sending them a survey will increase the response rate. Specifically, one group of people received a letter before getting the survey; one group received a phone call before receiving the survey; and one group didn't receive any information before the survey arrived. For this study, a response was defined as returning the survey within 2 weeks. The data are contained in the file U:\msu\course\stt\422\summer04\survey.dat. For reference, the contents of the file are given next.

```
01 yes letter 171
02 no letter 220
03 yes phone 146
04 no phone 68
05 yes none 118
06 no none 455
```

First comes the record number (1 through 6), then response status (yes or no) then contact method (letter, phone, none), then the number in the group. For example, there were 171 people who responded and received a preliminary letter; there were 220 people who didn't respond and received a preliminary letter.

The following SAS code reads in the data, drops the useless variable **record** and prints the raw data. One new aspect: In the **input** statement you see two dollar signs \$. These come after the names of variables that are "character" rather than numerical, in our case **response** and **contact**. If the \$ were omitted, SAS would expect a number, and would give an error message.

```
data survey;
  infile 'U:\msu\course\stt\422\summer04\survey.dat';
  input record response $ contact $ number;
  drop record;

proc print data = survey;

run;
```

Since the data contains the numbers in each group rather than a separate observation for each subject, we'll need to use the **weight** statement in **proc freq**. For example, the following SAS code will compute the frequencies for each variable **response** and **contact**.

```
proc freq data = survey;
  weight number;

run;
```

1. What proportion of subjects responded?
2. What proportion of subjects received a phone call?

We want to create a two-way table and see whether contacting the subject before the survey arrives is related to whether the subject responds. To create the table we use the **tables** statement. To perform the chi-square test of association we use the **chisq** option.

```
proc freq data = survey;
  weight number;
  tables contact * response / chisq;

run;
```

### 2.1.1 Explanation and questions

1. Each cell of the two-way table contains the count in the cell, the percentage of the whole dataset in the cell, the percentage of that row's data in the cell, and the percentage of that column's data in the cell. For example, the upper left cell represents those who received a letter but didn't respond. The number 220 tells us that there were 220 people in this category. The percentage 18.68 tells us that these 220 people are 18.68% of all people in the study. The percentage 56.27 tells us that 56.27% of those who received a letter didn't respond. The percentage 29.61 tells us that 29.61% of those who didn't respond received a letter.
2. What percentage of those who were not contacted responded? What proportion of those who were contacted by phone responded? Does it seem that contacting the subjects increases response?
3. After the table comes the output from the statistical tests. We'll concentrate on the first line labeled "Chi-Square." The other lines present other statistical methods for analyzing these data. What is the value of the Chi-Square statistic? Would you reject the null hypothesis of independence at the level  $\alpha = 0.01$ ?

## 2.2 Analyzing “raw” data

In the previous example we needed to use the **weight** statement in **proc freq** because the data consisted of the cell counts rather than raw data. Here we’ll analyze raw data; in the process we’ll learn a bit more about reading data from external files, and about controlling the output of **proc freq**.

The “individuals” data, described in the Data Appendix of the text, has records from a Bureau of Labor Statistics survey on 55,899 people from March, 2000. Variables are **id** (from 1 to 55899), **age**, **education** (from 1 through 6, with 1 representing “did not finish high school” and 6 representing “postgraduate degree.”), **gender** (1 = male and 2 = female), **income**, and **job** (categorizing by type of job). We’ll investigate the relationship between **gender** and **education**.

The data are in the file `U:\msu\course\stt\422\summer04\Text\Appendix\INDIVIDUALS.TXT`. Here are the first few lines of that file.

ID	AGE	EDUC	SEX	EARN	JOB
1	25	2	2	7234	5
2	25	5	1	37413	5
3	25	4	2	29500	5

We have to be a bit careful in reading the data for two reasons. First, we need to tell SAS to skip the first line, since it doesn’t contain data. That’s easy. In addition, the file uses tabs rather than spaces for separation. If we try to read the data as we have previously, SAS will generate error messages, because it is expecting spaces rather than tabs. This can also be fixed, although the fix is rather obtuse. Here is the code we need to read in the data.

```
data individuals;
infile 'U:\msu\course\stt\422\summer04\Text\Appendix\INDIVIDUALS.TXT'
      DLM='09'x firstobs = 2;
input id age education $ gender $ income job $;
cards;
```

The **firstobs = 2** statement tells SAS that the first observation is on line 2. The **DLM='09'x** statement tells SAS that the file is tab-delimited rather than space-delimited.

Once the data are read into SAS, we can use **proc freq** as before:

```
proc freq data = individuals;
tables education*gender / chisq nocol nocum nopercnt norow expected;

run;
```

Note that we don’t need a **weight** statement. Note also the options after **chisq**. These control the appearance of the table. The options beginning with **no** tell SAS what not to include. The **expected** option tells SAS to print the expected counts. Look at the output to see what you get.

## 2.3 Fisher's Exact Test

For 2 by 2 tables SAS computes Fisher's Exact Test automatically when the **chisq** option is used. The following example uses data on the diet (high fat or low fat) and coronary heart disease status of 23 people. Note that the cell counts are low enough that the chi-square test may not be valid.

```
data fatdiet;
  input diet $ chd $ count;
  cards;
  lowfat nochd 6
  lowfat chd 2
  highfat nochd 4
  highfat chd 11
  ;

proc print data = fatdiet;

proc freq data = fatdiet;
  weight count;
tables diet*chd/chisq;

run;
```