STA9750 – Lecture II

OUTLINE

1. Analysis of survey data
2. Basic confidence intervals and hypothesis tests:
    a. One-sample
    b. Two-sample
    c. One-way ANOVA

1. ANALYSIS OF SURVEY DATA

If we read in our Google forms data using the SAS input wizard, the variable that indicates your preferred % of classes where we cover R is treated as numerical. If we want to transform the variable to categorical, we can do it in at least the following 2 ways.

The first one is creating a new variable using IF and THEN statements (as we did in Lecture I):

```
DATA survey2;
      SET survey;
      LENGTH Rcat $ 3;
      IF R = 0.25 THEN Rcat ='25%';
      IF R = 0.30 THEN Rcat ='30%';
      IF R = 0.35 THEN Rcat='35%';
      IF R = 0.40 THEN Rcat='40%';
;
```

Another option is using the function PUT, which converts variables to categorical variables:

```
DATA survey3;
      SET survey;
      Rcat = PUT(R,4.2)
;
```

The 4.2 indicates that we want a variable whose length is 4 characters and has 2 decimal points. In this case, the new variable will look just like the old one, but the way that SAS processes the variable will be different.

You can check out the variable types with PROC CONTENTS:

```
PROC CONTENTS data=survey3;
RUN;
```

You can convert categorical variables that look numerical to actual numerical variables using the command INPUT. The example below shows how to do it.

```
DATA test;
      INPUT V1 $ 1-4;
      DATALINES;
1.34
2.25
3.35
;

DATA test2;
      SET test;
      v2 = INPUT(v1,4.);
;
```

After this, we can analyze the dataset with the tools we learned last time:

```
PROC MEANS data=survey3;
RUN;

PROC FREQ data=survey3;
      TABLES Rcat;
RUN;

PROC SGPLOT data=survey3;
      HBAR Rcat;
RUN;

PROC SGPLOT data=survey3;
      VBAR Experience;
RUN;
```

2. SOME CONFIDENCE INTERVALS AND HYPOTHESIS TESTS

Let's go back to the pollution dataset. Recall we have 3 numerical variables, POP, SO2, and TEMP.

Confidence intervals

You can get intervals for the numerical variables assuming normality with PROC UNIVARIATE. If our dataset is called pollution, we can get 95% intervals with the following command:

```
PROC UNIVARIATE data=pollution cibasic;
RUN;
```

If you want to change the confidence level to, say, 99%, you can accomplish that by doing:

```
PROC UNIVARIATE data=pollution cibasic(alpha=0.01);
RUN;
```

As you have probably noticed, PROC UNIVARIATE gives us too much info! The intervals are buried… Somewhere. You can tell PROC UNIVARIATE that you only want the intervals as follows:

```
ods select BasicIntervals;
PROC UNIVARIATE data=pollution cibasic(alpha=0.01);
RUN;
```

One sample t-tests

Doing one-sample t-tests is straightforward. For example, suppose that we want to test whether the average temperature in the US is equal to 70, assuming that the data are a representative sample from the population of US cities (WARNING: it possibly isn't!).

```
PROC TTEST data=pollution sides=2 alpha=0.05  H0=70;
      VAR TEMP;
RUN;
```

If you want to do one-sided testing, you only have to change the value of sides.

If your alternative hypothesis is TEMP < 70:

```
PROC TTEST data=pollution sides=L alpha=0.05  H0=70;
      VAR TEMP;
RUN;
```

And if your alternative hypothesis is TEMP > 70:

```
PROC TTEST data=pollution sides=U alpha=0.05  H0=70;
      VAR TEMP;
RUN;
```

Two-sample t-tests

This is quite similar to the one sample test. For example, if we construct a variable that classifies cities as big if they have more than 500k people and not big otherwise, we can compare SO2 levels between "big" and "not big" as follows:

```
DATA pollution2;
      SET pollution;
      LENGTH BIG $ 3 SIZE $ 6 ;
      IF POP > 500 THEN BIG='Yes';
      IF POP =< 500 THEN BIG='No';
      IF POP >= 750 THEN SIZE='big';
      IF POP < 750 and SIZE <= 300 THEN SIZE='medium';
      IF POP < 300 THEN SIZE = 'small';
;

PROC TTEST data=pollution2 sides=2 alpha=0.05;
      class BIG;
      var SO2;
RUN;
```

One-way ANOVA

When we created pollution2, we also defined a variable that is called SIZE which catalogues cities as big, medium, or small. We can run a one-way ANOVA as follows:

```
PROC ANOVA data=pollution2;
      class SIZE;
      model SO2 = SIZE;
RUN;
```