# Using Stata for One Sample Tests

All of the one sample problems we have discussed so far can be solved in Stata via either (a) statistical calculator functions, where you provide Stata with the necessary summary statistics for means, standard deviations, and sample sizes; these commands end with an i, where the i stands for "immediate" (but other commands also sometimes end with an i) (b) modules that directly analyze raw data; or (c) both. Some of these solutions require, or may be easier to solve, if you first add the Stataquest menus and commands; see

http://www.stata.com/support/faqs/res/quest7.html

The commands shown below can all be generated via Stata's pulldown menus if you prefer to use them.

---

## A. Single Sample Tests Case I: Sampling distribution of $\overline{X}$, Normal parent population (i.e. X is normally distributed), σ is known.

Problem. A manufacture of steel rods considers that the manufacturing process is working properly if the mean length of the rods is 8.6. The standard deviation of these rods always runs about 0.3 inches. Suppose a random sample of size n = 36 yields an average length of 8.7 inches. Should the manufacturer conclude the process is working properly or improperly? Use the .05 level of significance.

Solution. I don't know of any Stata routine that will do this by directly analyzing raw data. However, the `ztesti` command (which is installed with Stataquest) will do this when you have the summary statistics. Enter

```
ztesti 36 8.7 .3 8.6, level(95)
```

where the 5 parameters are the sample size N, the sample mean, the known population standard deviation, the hypothesized mean (under $H_0$), and the desired CI level (e.g. 95 for 95% confidence interval, 99 for 99% c.i.) The Stata results (which match up perfectly with our earlier analysis) are

```
. ztesti 36 8.7 .3 8.6, level(95)

                                                      Number of obs =       36

------------------------------------------------------------------------------
Variable |       Mean    Std. Err.        z     P>|z|       [95% Conf. Interval]
---------+--------------------------------------------------------------------
       x |        8.7         .05       174    0.0000       8.602002    8.797998
------------------------------------------------------------------------------

                            Ho: mean(x) = 8.6

      Ha: mean < 8.6          Ha: mean ~= 8.6            Ha: mean > 8.6
        z =   2.0000              z =   2.0000              z =   2.0000
    P < z =   0.9772        P > |z| =   0.0455          P > z =   0.0228
```

As is typical with many Stata commands, the output gives you the probability levels for both possible 1-tailed alternatives as well as for the 2-tailed alternative. In this case, the 2-tailed probability is .0455, which is less than .05, so we reject the null. Also, the 95% confidence interval does not include the hypothesized value of 8.6, so reject the null.

## B. Case II: Sampling distribution for the binomial parameter p.

Problem. The mayor contends that 25% of the city's employees are black. Various left-wing and right-wing critics have claimed that the mayor is either exaggerating or understating the number of black employees. A random sample of 120 employees contains 18 blacks. Test the mayor's claim at the .01 level of significance.

Exact Solution. We've shown how to get approximate solutions using the normal approximation to the binomial. However, with Stata, there is no need to rely on an approximation, as the `bitesti` and `bitest` commands can give you the exact answer, i.e. Stata is smart enough to work with the binomial distribution directly. Using the statistical calculator function `bitesti`, the format is

```
bitesti 120 18 .25
```

where the parameters are the number of trials, the observed number of successes, and the predicted probability of success. Stata gives you

```
. bitesti 120 18 .25

        N   Observed k   Expected k   Assumed p   Observed p
------------------------------------------------------------
      120         18           30       0.25000      0.15000

  Pr(k >= 18)            = 0.997208  (one-sided test)
  Pr(k <= 18)            = 0.005645  (one-sided test)
  Pr(k <= 18 or k >= 43) = 0.011020  (two-sided test)
```

For a two-tailed test, this result is significant at the .011 level. Since that is slightly more than .01, you stick with the null (barely). If the alternative hypotheses were $H_0$: $p < .25$, you would reject the null using the .01 level of significance, since .005645 is less than .01. To get the default Clopper-Pearson (aka "exact") 99% confidence interval, use the `cii` command:

```
cii 120 18, level(99)
```

where the parameters are the number of trials and the number of successes. The `level` parameter indicates that you want the 99% c.i.; the default is the 95% c.i.

```
. cii 120 18, level(99)

                                                          -- Binomial Exact --
    Variable |        Obs        Mean    Std. Err.       [99% Conf. Interval]
-------------+---------------------------------------------------------------
             |        120         .15     .032596        .0771953    .2517566
```

Again, the mayor's claim of .25 falls within the interval, so the null is not rejected.

While widely used, the Clopper-Pearson CI is criticized by some for being too conservative, i.e. it can produces confidence intervals that are very wide. Many therefore prefer Wilson's Confidence Interval or one of the other options (Jeffries, Agresti-Coull) that Stata offers. If you want Wilson's C.I., add the `wilson` parameter:

```
. cii 120 18, level(99) wilson

                                                         ------ Wilson ------
    Variable |        Obs        Mean    Std. Err.       [99% Conf. Interval]
-------------+---------------------------------------------------------------
             |        120         .15     .032596        .0845733    .2521024
```

If you had the raw data, you would have 102 cases coded 0 on a variable called `black` (meaning 102 employees were not black), and 18 cases coded 1 on the variable `black` (meaning 18 employees were black.). The `bitest` command analyzes the raw data:

```
. bitest black=.25

    Variable |        N   Observed k   Expected k   Assumed p   Observed p
-------------+------------------------------------------------------------
       black |      120         18           30       0.25000      0.15000

  Pr(k >= 18)            = 0.997208  (one-sided test)
  Pr(k <= 18)            = 0.005645  (one-sided test)
  Pr(k <= 18 or k >= 43) = 0.011020  (two-sided test)
```

The parameters are variable name = hypothesized probability of success. Likewise, to get the so-called exact (Clopper-Pearson) confidence interval, use the `ci` command with the `binomial` parameter:

```
. ci black, binomial level(99)

                                                         -- Binomial Exact --
    Variable |        Obs        Mean    Std. Err.       [99% Conf. Interval]
-------------+---------------------------------------------------------------
       black |        120         .15     .032596        .0771953    .2517566
```

To get the Wilson interval instead,

```
. ci black, binomial level(99) wilson

                                                         ------ Wilson ------
    Variable |        Obs        Mean    Std. Err.       [99% Conf. Interval]
-------------+---------------------------------------------------------------
       black |        120         .15     .032596        .0845733    .2521024
```

Note that we get identical results whether we use the immediate or raw data versions of the commands.

Normal Approximation to the Binomial Solution. You can also get the normal approximation to the binomial using the `prtesti` and `prtest` commands. The `prtesti` format is

```
prtesti 120 18 .25, count level(99)
```

where the parameters are N (the number of trials), the observed number of successes, and the predicted probability of success. If you didn't include the count parameter, you would say .15 instead of 18, i.e. you'd give the observed probability of success rather than the observed number of successes. The confidence interval is only approximate (and in this case wrong, because it does not include .25). The output is

```
. prtesti 120 18 .25, count level(99)

One-sample test of proportion                      x: Number of obs =      120

------------------------------------------------------------------------------
    Variable |       Mean   Std. Err.                     [99% Conf. Interval]
-------------+----------------------------------------------------------------
           x |        .15    .032596                      .0660382    .2339618
------------------------------------------------------------------------------

                          Ho: proportion(x) = .25

       Ha: x < .25              Ha: x != .25              Ha: x > .25
        z = -2.530                z = -2.530               z = -2.530
    P < z =  0.0057         P > |z| =  0.0114          P > z =  0.9943
```

Note that Stata does <u>not</u> do the correction for continuity by default. The `bintesti` command with the `normal` option, which comes with Stataquest, lets you make the correction by hand, i.e. you can add or subtract .5 from the observed number of successes as is appropriate.

```
. bintesti 120 18.5 .25 , normal level(99)

  Variable |      Obs  Proportion   Std. Error
  ---------+---------------------------------
         x |      120    .1541667    .0329645

           Ho:     p =
                   z = -2.42
            Pr > |z| = 0.0153
              99% CI = (0.0693,0.2391)
```

Incidentally, in this case, you actually come slightly closer to the correct result by NOT making the correction for continuity. After having tried several problems, my impression (possibly wrong) is that, when $p_0$ is close to .5, the correction improves accuracy, but as $p_0$ gets further and further away from .5 it may actually do more harm than good. The moral is that, if it is a very close call, you probably want to get the exact solution rather than rely on the normal approximation. Indeed, the Stata documentation for `prtest` says "Researchers are advised to use `bitest` when possible, especially for small samples."

With raw data, to get the normal approximation (without correction for continuity – I'm not sure how you would do the correction for continuity using raw data), use the `prtest` command with parameters varname=predicted probability of success.

```
. prtest black = .25, level(99)

One-sample test of proportion                  black: Number of obs =      120

------------------------------------------------------------------------------
    Variable |       Mean   Std. Err.                     [99% Conf. Interval]
-------------+----------------------------------------------------------------
       black |        .15    .032596                      .0660382    .2339618
------------------------------------------------------------------------------

                        Ho: proportion(black) = .25

     Ha: black < .25         Ha: black != .25          Ha: black > .25
          z = -2.530              z = -2.530               z = -2.530
      P < z = 0.0057         P > |z| = 0.0114           P > z = 0.9943
```

C. Case III: Sampling distribution of $\overline{X}$, normal parent population, σ unknown.

Problem. The Deans contend that the average graduate student makes $8,000 a year. Zealous administration budget cutters contend that the students are being paid more than that, while the Graduate Student Union contends that the figure is less. A random sample of 6 students has an average income (measured in thousands of dollars) of 6.5 and a sample variance of 2. Using both confidence intervals and significance tests, test the Deans' claim at the .10 and .02 levels of significance.

Solution. Use the `ttesti` or the `ttest` command. If you just have the summary statistics and you want the 90% confidence interval, enter the command

```
ttesti 6 6.5 1.414213562 8, level(90)
```

The parameters are N, the sample mean, the sample sd, the predicted mean, and the CI level. The results are

```
. ttesti 6 6.5 1.414213562 8, level(90)

One-sample t test

------------------------------------------------------------------------------
         |     Obs        Mean    Std. Err.   Std. Dev.   [90% Conf. Interval]
---------+--------------------------------------------------------------------
       x |       6         6.5    .5773503    1.414214    5.336611    7.663389
------------------------------------------------------------------------------
Degrees of freedom: 5

                              Ho: mean(x) = 8

     Ha: mean < 8              Ha: mean != 8              Ha: mean > 8
       t =  -2.5981              t =  -2.5981               t =  -2.5981
   P < t =   0.0242          P > |t| =   0.0484          P > t =   0.9758
```

To get the 98% c.i., just change the `level` parameter:

```
. ttesti 6 6.5 1.414213562 8, level(98)

One-sample t test

------------------------------------------------------------------------------
         |     Obs        Mean    Std. Err.   Std. Dev.   [98% Conf. Interval]
---------+--------------------------------------------------------------------
       x |       6         6.5    .5773503    1.414214    4.557257    8.442743
------------------------------------------------------------------------------
Degrees of freedom: 5

                               Ho: mean(x) = 8

     Ha: mean < 8              Ha: mean != 8              Ha: mean > 8
       t =  -2.5981              t =  -2.5981               t =  -2.5981
   P < t =   0.0242          P > |t| =   0.0484           P > t =   0.97588
```

If you just want the confidence interval, use the `cii` command, where the parameters are N, the sample mean, the sample sd, and the CI level.

```
. cii 6 6.5 1.414213562, level(90)

    Variable |        Obs        Mean    Std. Err.       [90% Conf. Interval]
-------------+---------------------------------------------------------------
             |          6         6.5    .5773503        5.336611    7.663389
```

If you are analyzing the original raw data, use the `ttest` command. In this example, `pay` is the variable containing salary information, and 8 is the hypothesized mean of pay. The `level(90)` parameter gives us the 90% confidence interval.

```
. ttest pay = 8, level(90)

One-sample t test

------------------------------------------------------------------------------
Variable |     Obs        Mean    Std. Err.   Std. Dev.   [90% Conf. Interval]
---------+--------------------------------------------------------------------
     pay |       6         6.5    .5773503    1.414214    5.336611    7.663389
------------------------------------------------------------------------------
Degrees of freedom: 5

                              Ho: mean(pay) = 8

     Ha: mean < 8              Ha: mean != 8              Ha: mean > 8
       t =  -2.5981              t =  -2.5981               t =  -2.5981
   P < t =   0.0242          P > |t| =   0.0484           P > t =   0.9758
```

To get the 98% c.i.,

```
. ttest pay = 8, level(98)

One-sample t test

------------------------------------------------------------------------------
Variable |     Obs        Mean    Std. Err.   Std. Dev.   [98% Conf. Interval]
---------+--------------------------------------------------------------------
     pay |       6         6.5    .5773503    1.414214    4.557257    8.442743
------------------------------------------------------------------------------
Degrees of freedom: 5

                              Ho: mean(pay) = 8

     Ha: mean < 8               Ha: mean != 8              Ha: mean > 8
       t =  -2.5981                t =  -2.5981              t =  -2.5981
   P < t =   0.0242          P > |t| =   0.0484          P > t =   0.9758
```

If you just wanted to get the 90% confidence interval without the t-test, use the `ci` command specifying whatever level you want:

```
. ci pay, level(90)

    Variable |        Obs        Mean    Std. Err.       [90% Conf. Interval]
-------------+---------------------------------------------------------------
         pay |          6         6.5    .5773503        5.336611    7.663389
```

For the 98% c.i.,

```
. ci pay, level(98)

    Variable |        Obs        Mean    Std. Err.       [98% Conf. Interval]
-------------+---------------------------------------------------------------
         pay |          6         6.5    .5773503        4.557257    8.442743
```

As before, the output from `ttest` and `ttesti` is pretty much identical; if you did see any differences, it would be because of rounding error in the summary statistics.