Name: _

1 Objective: Compare simulated confidence intervals (CI) coverage with nominal 95% CI coverage

This lab is based on the paper *Approximate is Better than "Exact" for Interval Estimation of Binomial Proportions* by Alan Agresti and Brent A. Coull (The American Statistician by American Statistical Association, May 1998, Vol. 52, No. 2, p.119-126)

The coverage probability of a confidence interval (CI) is the actual probability that the interval contains the true value of statistics interest (e.g., mean, standard deviation etc.). Coverage probability that we will use is

$$C_n(p) = C(n, p, \text{method}) = P(p \text{ in CI}, p \text{ true population proportion}) = \sum_{k=0}^{n} I(k, p)\binom{n}{k} p^k (1-p)^{n-k}$$

where $I(k,p) = \begin{cases} 1 & \text{if the interval contains } p \\ 0 & \text{if otherwise.} \end{cases}$

The function C is supposed to equal to nominal confidence level of 95%. We will consider two types of point estimation:

$$\rho = \begin{cases} \widehat{p} \coloneqq \frac{s}{n} & \text{If ordinary sample proportion} \\ \widetilde{p} \coloneqq \frac{s+2}{n+4} & \text{If adjusted Wald approximation} \end{cases}$$

where $\widetilde{p} = \frac{s+z^2/2}{n+z^2} \approx \frac{s+2}{n+4}$ and $z = 1.96$ is the 95% CI cut off of the normal distribution.

Assume S follows binomial distribution $S \sim \text{Bin}(n, p)$ (e.g., S = number of success ($H \coloneqq$ heads) in n sample), then $\bar{S} = E[S] = np$ and $\text{Var}(S) = np(1-p)$. Binomial confidence interval is given by

$$\left(\rho - z^* \sqrt{\frac{\rho(1-\rho)}{n}}, \rho + z^* \sqrt{\frac{\rho(1-\rho)}{n}}\right)$$

based on CLT for large enough n.

We'll look at two cases:

[A] in unknown variable case using T method for three populations

- [1] Caucy
- [2] exponential
- [3] normal

[B] in proportion case using Z method and two estimators (H is for head in a coin toss)

- [1] $\hat{p}$ = phat = ordinary sample proportion : #H/#sam = S/n
- [2] $\tilde{p}$ = pcurl = (#H + 2)/(#sam + 4) = $S + 2/n + 4$

2 Key Words

`rancau`, `ranexp`, `rannor`, `ranbin`, coverage probability.

3 Turn In

In each case, simulate 10,000 95% CI of sample size `nc` and report in table form observed percentage of 10,000 that cover after you've done case A and case B.

Table 1: Observed Coverage Percentage `cv_hat`, `cv_curl`

[A1]	Cauchy	`nc` = 8	______________	**[A2]**	Exponential	`nc` = 8	______________
[A3]	Normal	`nc` = 8	______________				
[B1]	phat	`nc` = 12	______________	**[B2]**	pcurl	`nc` = 12	______________
[B1]	phat	`nc` = 17	______________	**[B2]**	pcurl	`nc` = 17	______________
[B1]	phat	`nc` = 18	______________	**[B2]**	pcurl	`nc` = 18	______________

4 Case [A]: T Method

GET SAMPLES: `nr` students (rows) each take `nc` samples (columns). Unlike last lab, we do all processing in first data step.

```
DATA samples;
        seed = 1234875;
        nr = 10; nc = 8;                        /* WHEN WORKS, SET nr = 10000 */
        array x[20];                            /* USE 20 TO ALLOW nc <= 20 */
        tstar = abs( tinv(.025, nc-1) );        /* 95% CONF FACTOR FROM T TABLE */
        mpop = 0 ;                              /* = 1 if EXP'L, = 0 if NORM & CAUCHY */
        do r = 1 to nr;
                do c = 1 to nc;
                        x[c] = rancau(seed);    /* USE RANCAU, RANEXP & RANNOR */
                        mn = mean(of x[*]);     /* GET SAMPLE MEAN AND STD */
                        s2 = var(of x[*]);
                        t = (mn - mpop)/ sqrt( s2 / nc );       /* GET T-STT */
                        if abs(t) <= tstar then cv = 1;
                                else cv = 0;
                end;
                output;
        end;
        keep x1-x8 mn s2 t cv tstar;
run;
```

Note: CI covers `mpop` if and only if `abs( mn - mpop ) <= tstar * sd err` if and only if `abs(t) <= tstar`. Next we do logical test (the "if" condition above) to see which data rows produced Cl's that covered.

Run below code only for `nr` = 10 case to see your data.

```
PROC print data = samples;
run;
```

To find proportion of CIs that cover, find mean of `cv`.

```
PROC means data = samples n mean stderr;
        var cv;
run;
```

Look at distribution of `t`: should be T with df = 7: should be symmetric bell-shaped (explain why?).

```
PROC gchart data = samples;
        vbar t;
run;
```

When above is running, CHANGE `nr` to 10,000 and use `rancau`, `ranexp`, and `rannor`, with appropriate `mpop`, to fill out Table 1, part [A].

5 Case [B]: Proportions - Z Method

Code below is quite close to code in part [A], so you don't have to retype.

```
DATA propns;
        seed = 123475;
        nr = 10; nc = 12;                    /* WHEN WORKS, SET nr = 10000, nc = 12, 17, 18 */
        array x[20];                          /* USE 20 TO ALLOW nc <= 20 */
        zstar = abs( probit(.025) );         /* 95% CONF FACTOR FROM Z TABLE */
        ppop = .5 ;                           /* POPULATION PROPORTION */
        do r = 1 to nr;
                do c = 1 to nc;
                        x[c] = ranbin(seed, 1, ppop);
                        nh = sum(of x[*]);          /* GET number of heads */
                        phat = nh / nc;             /* ordinary sample proportion and std error */
                        se_phat = sqrt( phat * (1-phat) / nc );
                        pcurl = (nh + 2)/(nc + 4);  /* 'add 2' sample proportion and std error */
                        se_pcurl = sqrt( pcurl * (1-pcurl) / (nc + 4) );
                        zhat = (phat - ppop)/se_phat;              /* z_stt's */
                        zcurl = (pcurl - ppop)/se_pcurl;
                        if abs(zhat) <= zstar then cv_hat = 1;
                                else cv_hat = 0;
                        if abs(zcurl) <= zstar then cv_curl = 1;
                                else cv_curl = 0;
                end;
                output;
        end;
        keep phat se_phat pcurl se_pcurl zhat zcurl cv_hat cv_curl;
run;
```

Note: CI covers `ppop` if and only if `abs( sam_propn - ppop ) <= zstar * sd_err` if and only if `abs(z) <= zstar`.

Run below code only for `nr` = 10 case to see your data.

```
PROC print data = propns;
run;
```

To find proportion of CI's that cover, find mean of `cv`

```
PROC means data = propns n mean stderr;
       var cv_hat cv_curl;
run;
```

Look at distribution of `z`: should be nearly standard normal (explain why?).

```
PROC gchart data = propns;
       vbar zhat zcurl;
run;
```

When above is running, change `nr` to 10,000 use `nc` = 12, 17 and 18 to fill out Table 1, part [B].