R lang Cheat Sheet

Chapter 9

1) Population of one sample with variance known (Z-test)

- **Test Statistic(Z-Test)**
- `Calculate Standard error (SE): - sigma/sqrt(n)`
- `Z- test = (xbar - mu0)/SE`
- **If Lower Tail :-**
  - `Pval = pnorm(z)`

- **If Upper Tail :-**
  - `Pval = pnorm(z,lower.tail=FALSE)`

- **If Two - Tail :-**
  - IF Z value is Positive :-
  - `Pval = 2 * pnorm(z,lower.tail = FALSE)`
  - IF Z value is Negative :-
  - `Pval = 2 * pnorm(z)`
- Reject H0 if P Value < alpha
- Accept H0 if PValue > alpha

- Using Critical value method :-
  - Lower-Tail :-
    - `alpha = .05`
    - `z.alpha = qnorm(1-alpha)`
    - `-z.alpha`

  - Upper-Tail :-
    - `alpha = .05`
    - `z.alpha = qnorm(1-alpha)`
    - `z.alpha`

  - Two-Tail :-
    - `alpha = .05`
    - `z.alpha = qnorm(1-alpha/2)`
    - `c(-z.alpha,z.alpha)`

- Find Confidence intervals :-
    - `E(margin of error) = qnorm(alpha)*SE`
    - `CI = Xbar + c(-E,E)`

- Type II error:-
    - **If Lower-Tail :-**

```
(1) q = qnorm(alpha, mean=mu0, sd=se)

(2) mu = 9950              # assumed actual mean
    beta = pnorm(q, mean=mu, sd=sem, lower.tail=FALSE)
    Power of test = 1-beta
```

- **If Upper -Tail :-**

```
(3) q = qnorm(alpha, mean=mu0, sd=se, lower.tail=FALSE)

(4) mu = 9950              # assumed actual mean
    pnorm(q, mean = mu, sd=sem)
    Power of test = 1-beta
```

- **If Two -Tail :-**

```
1)  i = c(alpha/2, 1-alpha/2)
2)  q = qnorm(i, mean=mu0, sd=sem)

3)  mu = 15.1              # assumed actual mean
4)  p = pnorm(q, mean=mu, sd=sem)
```

2) Population of one sample when variance is unknown (T-test)

- **Test Statistic(T-Test)**
- `Calculate Standard error (SE): - sd/sqrt(n)`
- `T- test =  (xbar - mu0)/SE`
- If Lower Tail :-
    - `Pval = pt(t,df=n-1)`

- If Upper Tail :-
    - `Pval = pt(z,lower.tail=FALSE,df=n-1)`

- If Two - Tail :-
    - IF T value is Positive :-
    - `Pval = 2 * pt(z,lower.tail = FALSE,df=n-1)`

- IF T value is Negative :-
  - `Pval = 2 * pt(z,df=n-1)`
- Reject H0 if P Value < alpha
- Accept H0 if PValue > alpha

- Using Critical value method :-
  - Lower-Tail :-
    - `alpha = .05`
    - `z.alpha = qt(1-alpha,df= n-1)`
    - `-z.alpha`
  - Upper-Tail :-
    - `alpha = .05`
    - `z.alpha = qt(1-alpha,df= n-1)`
    - `z.alpha`
  - Two-Tail :-
    - `alpha = .05`
    - `z.alpha = qt(1-alpha/2,df= n-1)`
    - `c(-z.alpha,z.alpha)`
  - Find Confidence intervals :-
    - `E(margin of error) = qt(alpha,df=n-1)*SE`
    - `CI = Xbar + c(-E,E)`

- Type II error:-
  - If Lower-Tail :-

```
1) q = mu0 + qt(alpha, df=n-1) * SE

2) mu = 9950              # assumed actual mean
   P = pt((q - mu)/SE, df=n-1, lower.tail=FALSE)
```

  - If Upper -Tail :-

```
1) q = mu0 + qt(alpha, df=n-1,lower.tail=FALSE) * SE

2) mu = 9950      # assumed actual mean
   P = pt((q - mu)/SE, df=n-1)
```

  - If Two -Tail :-

```
1) i = c(alpha/2, 1-alpha/2)
2) q = mu0 + qt(i, df=n-1) * SE

3) mu = 15.1              # assumed actual mean
   P = pt((q - mu)/SE, df=n-1)
```

3) Population Proportion of one sample

- **Test Statistic**
- `Calculate Standard error (SE): - sqrt(p0*(1-p0)/n)`
- `Z- test = (Pbar-p0)/SE`
- If Lower Tail :-
  - `Pval = pnorm(z)`

- If Upper Tail :-
  - `Pval = pnorm(z,lower.tail=FALSE)`

- If Two - Tail :-
  - IF Z value is Positive :-
  - `Pval = 2 * pnorm(z,lower.tail = FALSE)`
  - IF Z value is Negative :-
  - `Pval = 2 * pnorm(z)`
- Reject H0 if P Value < alpha
- Accept H0 if PValue > alpha

- Using Critical value method :-
- Lower-Tail :-
  - `alpha = .05`
  - `z.alpha = qnorm(1-alpha)`
  - `-z.alpha`

- Upper-Tail :-
  - `alpha = .05`
  - `z.alpha = qnorm(1-alpha)`
  - `z.alpha`

- Two-Tail :-
  - `alpha = .05`
  - `z.alpha = qnorm(1-alpha/2)`
  - `c(-z.alpha,z.alpha)`

- Find Confidence intervals :-
  - `E(margin of error) = qnorm(alpha)*SE`
  - `CI = Xbar + c(-E,E)`

Chapter 10

4) Population of two samples when variance is known

- **Test Statistic(Z-Test)**
- `Calculate Standard error (SE):    sqrt(sigma1^2/n1+sigma2^2/n2)`
- `PE(Point Estimate) = xbar1 - xbar2`
- `Z- test =  PE/SE`
- If Lower Tail :-
  - `Pval = pnorm(z)`

- If Upper Tail :-
  - `Pval = pnorm(z,lower.tail=FALSE)`

- If Two - Tail :-
  - IF Z value is Positive :-
  - `Pval = 2 * pnorm(z,lower.tail = FALSE)`
  - IF Z value is Negative :-
  - `Pval = 2 * pnorm(z)`
  - Reject H0 if P Value < alpha
  - Accept H0 if PValue > alpha

- Using Critical value method :-
  - Lower-Tail :-
    - `alpha = .05`
    - `z.alpha = qnorm(alpha)`
    - `-z.alpha`
  - Upper-Tail :-
    - `alpha = .05`

- `z.alpha = qnorm(1-alpha)`
- `z.alpha`

- Two-Tail :-
  - `alpha = .05`
  - `z.alpha = qnorm(1-alpha/2)`
  - `c(-z.alpha,z.alpha)`

- Find Confidence intervals :-
  - `E(margin of error) = qnorm(alpha)*SE`
  - `CI = PE+ c(-E,E)`

5) Population of two samples when variance is unknown

- **Test Statistic(T-Test)**
- `Calculate Standard error (SE):`
  - `sqrt(s1^2/n1+s2^2/n2)`
- `Calculate DF :`
  - `df = (s1^2/n1 + s2^2/n2)^2/(1/(n1-1)*(s1^2/n1)^2 + 1/(n2-1)*(s2^2/n2)^2)`
  - `df = as.integer((df))`

- `PE(Point Estimate) = (xbar1 - xbar2)`
- `T- test =  PE/SE`
- If Lower Tail :-
  - `Pval = pt(z,df=df)`

- If Upper Tail :-
  - `Pval = pt(z,lower.tail=FALSE,df=df)`

- If Two - Tail :-
  - IF Z value is Positive :-
  - `Pval = 2 * pt(z,lower.tail = FALSE,df=df)`
  - IF Z value is Negative :-
  - `Pval = 2 * pt(z,df=df)`
- Reject H0 if P Value < alpha
- Accept H0 if PValue > alpha

- Using Critical value method :-
  - Lower-Tail :-
    - `alpha = .05`
    - `z.alpha = qt(alpha,df=df)`
    - `-z.alpha`
  - Upper-Tail :-
    - `alpha = .05`
    - `z.alpha = qt(1-alpha,df=df)`
    - `z.alpha`
  - Two-Tail :-
    - `alpha = .05`
    - `z.alpha = qt(1-alpha/2,df=df)`
    - `c(-z.alpha,z.alpha)`
  - Find Confidence intervals :-
    - `E(margin of error) = qt(alpha,df=df)*SE`
    - `CI = PE+ c(-E,E)`

6) Population proportion of two samples

- **Test Statistic(Z-Test)**
- `Calculate Standard error (SE):`
    - `sqrt(p1*(1-p1)/n1+p2*(1-p2)/n2)`
- `PE(Point Estimate) = pbar1 - pbar2`
- `Z- test = PE/SE`
- If Lower Tail :-
  - `Pval = pnorm(z)`
- If Upper Tail :-
  - `Pval = pnorm(z,lower.tail=FALSE)`
- If Two - Tail :-
  - IF Z value is Positive :-
  - `Pval = 2 * pnorm(z,lower.tail = FALSE)`
  - IF Z value is Negative :-
  - `Pval = 2 * pnorm(z)`
  - Reject H0 if P Value < alpha

- Accept H0 if PValue > alpha

- Using Critical value method :-
  - Lower-Tail :-
    - `alpha = .05`
    - `z.alpha = qnorm(alpha)`
    - `-z.alpha`
  - Upper-Tail :-
    - `alpha = .05`
    - `z.alpha = qnorm(1-alpha)`
    - `z.alpha`
  - Two-Tail :-
    - `alpha = .05`
    - `z.alpha = qnorm(1-alpha/2)`
    - `c(-z.alpha,z.alpha)`
  - Find Confidence intervals :-
    - `E(margin of error) = qnorm(alpha/2)*SE`
    - `CI = PE+ c(-E,E)`

Chapter 11

7) Inferences about Variance

- Confidence Interval for sample variance
    - `s_sq = sample variance`
    - `n = number of samples`
    - `alpha = significance level`

    - `chq = qchisq(c(1-alpha/2,alpha/2), df = n-1)`
    - `CI = (n-1)*s_sq/chq`

- Hypothesis testing for variance
  - `sigma_sq = Hypothesized Variance`
  - `s_sq = Sample Variance`
  - `n = Number of Samples`
  - `alpha = Significance Level`

  - `z = (n-1)*s_sq/sigma_sq`

  - If Lower Tail :-
    - `Pval = pchisq(z,df=n-1)`

  - If Upper Tail :-
    - `Pval = pchisq(z,lower.tail=FALSE,df=n-1)`

  - If Two - Tail :-
    - `Pval = pchisq(z,df=n-1)`

  - Reject H0 if P Value < alpha
  - Accept H0 if PValue > alpha

- Find Confidence intervals :-
  - `Chsq = qchisq(c(1-alpha/2,alpha/2),df=n-1)`
  - `CI = (n-1)*s_sq/chsq`

8) Inferences about Variances

- F-Test :-
  - `n1 = number of samples in first category`
  - `s1 = variance of samples in first category`
  - `n2 = number of samples in second category`
  - `s2 = variance of samples in Second category`
  - `alpha = Significance Level`
  - `f = s1/s2`

`For Upper Tail :-`

  - `pval = pf(f,df1 = n1-1,df2 = n2-1,lower.tail = FALSE)`

`For Lower Tail :-`

  - `pval = pf(f,df1 = n1-1,df2 = n2-1)`

Chapter 12

9) Goodness of Fit

- `real_per = Probabilities given`
- `obs_freq = observed frequencies`
- `alpha = Significance Level`
- `chisq.test(obs_freq, p = real_per)`

10) Test of Independence

A test of independence addresses the question of whether the beer preference (light, regular, or dark) is independent of the gender of the beer drinker (male, female). The hypotheses for this test of independence are:

H_0: Beer preference is independent of the gender of the beer drinker
H_a: Beer preference is not independent of the gender of the beer drinker

- `data = matrix(c(20,30,20,30,60,25,10,15,30),nrow= 3, ncol = 3)`
- `rownames(data) = c("M","F","L")`
- `colnames(data) = c("L","R","D")`
- `data`
- `alpha = 0.05`
- `chisq.test(data)`