Institute of Actuaries of India

Subject CS1-Actuarial Statistics (Paper B)

December 2022 Examination

INDICATIVE SOLUTION

Introduction
The indicative solution has been written by the Examiners with the aim of helping candidates. The solutions given are only indicative. It is realized that there could be other points as valid answers and examiner have given credit for any alternative approach or interpretation which they consider to be reasonable.

Solution 1:

i)

```
> data<-read.csv("weights.csv")                                   (1)
> summary(data)                                                   (1)
Gender           Weights          Day
 Length:216        Min.   : 5.00   Min.   : 1.00
 Class :character  1st Qu.: 7.00   1st Qu.: 9.00
 Mode  :character  Median :10.00   Median :17.00
                   Mean   :10.22   Mean   :17.16
                   3rd Qu.:13.00   3rd Qu.:25.25
                   Max.   :16.00   Max.   :34.00
```

[2]

ii)

```
> weight_M <- subset(data$Weights, data$Gender == "M" , select = c(data$Weights), drop = FALSE)     (1)
```

Alternate:

```
M_subset<-data[data$Gender=="M",]
weight_M <- M_subset$Weights
```

Marks given for other valid alternate solutions.

```
> nm <- length(weight_M) -1  /* n-1
> sm <- sd(weight_M)                                              (0.5)
> nm                                                              (0.5)
[1] 107
> sm
[1] 2.306458
> sm*sqrt(nm/qchisq(c(0.95,0.05),nm))
[1] 2.075453 2.601171                                             (1)
```

[3]

iii)

```
>  mum<-12                                                        (0.5)
> xbarm <-mean(weight_M)                                          (0.5)
> nm <- length(weight_M)                                          (0.5)
> Statisticm <- ((xbarm-mum)/(sm/sqrt(nm)))
> Statisticm
[1] 3.295867                                                      (1)
>  1- pnorm(Statisticm)
[1] 0.0004905918                                                  (1)
```

p- value <5%, there is no significant evidence to accept the null hypothesis

[Max 3]

Alternate:

```
One Sample t-test

data:  weight_M                                                   (2)
t = 3.2959, df = 107, p-value = 0.001332
alternative hypothesis: true mean is not equal to 12
95 percent confidence interval:
 12.29151 13.17145
sample estimates:                                                 (0.5)
mean of x
 12.73148
```

p- value = .001332 <5%, there is no significant evidence to accept the null hypothesis (1)

[Max 3]

iv)

```
> weight_F <- subset(data$Weights, data$Gender == "F" , select = c(data$Weights), drop = FALSE)     (1)
> nf <- length(weight_F)                                                    (0.5)
> xbarf                                                                     (0.5)
[1] 7.703704
> sf <- sd(weight_F)                                                        (0.5)
> nf
[1] 108
> sf
[1] 2.079087
> Statisticf <- ((xbarf-muf)/(sf/sqrt(nf)))
> Statisticf
[1] 3.517459                                                                (0.5)
> pnorm(Statisticf)
[1] 0.0002178499
```

p- value <5%, there is no significance evidence to accept the null hypothesis (0.5)

Alternate: (1)

```
F_subset<-data[data$Gender=="F",]
weight_F <- F_subset$Weights
 t.test(weight_F,mu=7,alternative = "two.sided")                            (1)
                                                                            (2)
            One Sample t-test

data: weight_F
t = 3.5175, df = 107, p-value = 0.0006407
alternative hypothesis: true mean is not equal to 7
95 percent confidence interval:
 7.307108 8.100300
sample estimates:
mean of x
 7.703704
```

The required p value is 0.0006407. (1)

p- value <5%, there is no significance evidence to accept the null hypothesis

[Max 4]

v)

```
> set.seed(2022)
> male<-c(rnorm(10,mum,sm))                                                 (1)
> female<-c(rnorm(10,muf,sf))                                               (0.5)
> mean(male)
[1] 10.69983
> mean(female)
[1] 7.015142
> t.test(male,female,paired=TRUE,alternative="less",mu=5)                   (2)

       Paired t-test

data:  male and female
t = -1.7111, df = 9, p-value = 0.06061
alternative hypothesis: true mean difference is less than 5
95 percent confidence interval:
    -Inf 5.093811
sample estimates:
mean difference
      3.684693
```

Since p-value >5%,there is no strong evidence to reject null hypothesis (0.5)

The average weights used by females is 3.7kg lesser than weights used by male (0.5)

[Max 4]

[16 Marks]

Solution 2:

i)

```
> abc <- c(-6,-10,3,18,-10,1,3,-13,-14,13)
> xyz <- c(8,0,-4,10,20,8,0,10,5,-19)
> pqr <- c(-20,19,3,13,20,7,-3,13,20,1)
> lmn <- c(14,4,5,15,19,9,6,10,7,16)
>
> abc_mean = mean(abc)
> xyz_mean = mean(xyz)
> pqr_mean = mean(pqr)
> lmn_mean = mean(lmn)
>
> print(abc_mean)
[1] -1.5
> print(pqr_mean)
[1] 7.3
> print(xyz_mean)
[1] 3.8
> print(lmn_mean)
[1] 10.5
>
> abc_sd = sd(abc)
> xyz_sd = sd(xyz)
> pqr_sd = sd(pqr)
> lmn_sd = sd(lmn)
>
> print(abc_sd)
[1] 11.00757
> print(pqr_sd)
[1] 12.62317
> print(xyz_sd)
[1] 10.46476
> print(lmn_sd)
[1] 5.190804
```

[Max 5]

ii)

```
> r_abc_xyz = cor(abc,xyz)
> r_abc_pqr = cor(abc,pqr)
> r_abc_lmn = cor(abc,lmn)
> r_xyz_pqr = cor(xyz,pqr)
> r_xyz_lmn = cor(xyz,lmn)
> r_pqr_lmn = cor(pqr,lmn)
>
> print(r_abc_xyz)
[1] -0.4456345
> print(r_abc_pqr)
[1] -0.2842739
> print(r_abc_lmn)
[1] 0.2634939
> print(r_xyz_pqr)
[1] 0.2923746
> print(r_xyz_lmn)
[1] 0.2577296
> print(r_pqr_lmn)
[1] -0.08393819
```

[3]

iii)

```
> cor.test(abc,xyz,method="pearson",alternative="two.sided",conf.level = 0.95)

        Pearson's product-moment correlation

data:  abc and xyz
t = -1.408, df = 8, p-value = 0.1968
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.8396649  0.2557513
sample estimates:
      cor
-0.4456345
```

(1)

```
> cor.test(pqr,lmn,method="pearson",alternative="two.sided",conf.level = 0.95)

        Pearson's product-moment correlation

data:  pqr and lmn
t = -0.23825, df = 8, p-value = 0.8177
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.6777458  0.5761368
sample estimates:
       cor
-0.08393819
```

(1)

(1)

Confidence interval for correlation coefficient between returns of ABC Oil and XYZ Airways is (-0.84,0.26). Since it does not contain -1, we can say that there is no possibility of a perfect negative correlation at 5% level of significance.

(1)

Confidence interval for correlation coefficient between returns of PQR Realty and LMN Bank is (-0.68,0.58). Since it does not contain -1, we can say that there is no possibility of a perfect negative correlation at 5% level of significance.

[4]

iv)

```
> pABC = 0.50
> pXYZ = 0.50
> pPQR = 0.75
> pLMN = 0.25
>
> strat_A = pABC * abc + pXYZ * xyz
> print(mean(strat_A))
[1] 1.15
>
> strat_B = pPQR * pqr + pLMN * lmn
> print(mean(strat_B))
[1] 8.1
```

(1.5)

(0.5)

Since, strategy B gives higher mean returns, it would be wise to go with strategy B, if mean returns is the metric to be targeted.

[2]

v) H_0: Variance of Returns from Strategy A = Variance of Returns from Strategy B (1)

H_1: Variance of Returns from Strategy A ≠ Variance of Returns from Strategy B

```
> var.test(x=strat_A,y=strat_B,conf.level = 0.90)

        F test to compare two variances

data:  strat_A and strat_B
F = 0.35856, num df = 9, denom df = 9, p-value = 0.1426
alternative hypothesis: true ratio of variances is not equal to 1
90 percent confidence interval:
 0.112795 1.139835
sample estimates:
ratio of variances
       0.3585634
```

(2)

Since the p-value 0.1426 > 10%, we have sufficient evidence to accept the null hypothesis at the 10% level of significance. Hence, the investor's assumption of the two strategies being equally risky seems reasonable.

(1)

[4]

vi)

```
> sd_A1 = (pABC^2 * abc_sd^2 + 2 * pABC * pXYZ * abc_sd * xyz_sd * -0.83 + pXYZ^2 * xyz_s
d^2)^(1/2)
> sd_A2 = (pABC^2 * abc_sd^2 + 2 * pABC * pXYZ * abc_sd * xyz_sd * 0.26 + pXYZ^2 * xyz_sd
^2)^(1/2)
>
> print(sd_A1)
[1] 3.140851
> print(sd_A2)
[1] 8.523165

> sd_B1 = (pPQR^2 * pqr_sd^2 + 2 * pPQR * pLMN * pqr_sd * lmn_sd * -0.68 + pLMN^2 * lm
n_sd^2)^(1/2)
> sd_B2 = (pPQR^2 * pqr_sd^2 + 2 * pPQR * pLMN * pqr_sd * lmn_sd * 0.58 + pLMN^2 * lmn
_sd^2)^(1/2)
>
> print(sd_B1)
[1] 8.637509
> print(sd_B2)
[1] 10.27457
```

The limits for standard deviation of Strategy A are (3.14, 8,52) and the limits for standard deviation of Strategy B are (8.64, 10.27).

Based on Metric 2, since Strategy A has lower range of standard deviation, Strategy A would be selected.

[Max 4]

vii) We develop limits for the risk-adjusted returns under Strategies A and B using the following code:

```
> RAR_A1 = mean(strat_A)/sd_A1
> RAR_A2 = mean(strat_A)/sd_A2
> RAR_B1 = mean(strat_B)/sd_B1
> RAR_B2 = mean(strat_B)/sd_B2
>
> print(RAR_A1)
[1] 0.3661428
```

```
> print(RAR_A2)
[1] 0.1349264
> print(RAR_B1)
[1] 0.9377704
> print(RAR_B2)
[1] 0.788354                                                    (1)
```

Based on the above the limits for Risk-Adjusted Return for Strategy A are (0.13,0.37) and limits for Strategy B are (0.79,0.94). Since Strategy B gives higher risk-adjusted returns, we should go for Strategy B using Metric 3.

Strategy B gives higher returns and has higher risk, but it eventually ends up giving higher risk adjusted returns (i.e. more returns per unit of risk) as compared to Strategy A. (1)

[2]

[24 Marks]

Solution 3:

i)

```
> #i.
> PA<-read.csv("PA_Data.csv")                                                          (1)
> model1<-glm(Claim~Gender*Health+Age,family=poisson(lin="log"),data=PA)               (2)


> summary(model1)                                                                      (1)
Call:
glm(formula = Claim ~ Gender * Health + Age, family = poisson(lin = "log"),
   data = PA)
Deviance Residuals:
     Min       1Q    Median       3Q       Max
-1.40121  -0.78101   0.03472   0.42900   1.37735

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)              0.17452    0.95762   0.182   0.8554
GenderM                  0.06235    0.46853   0.133   0.8941
HealthNonDiabetic       -1.31680    0.64556  -2.040   0.0414 *
Age                      0.02248    0.02052   1.095   0.2734
GenderM:HealthNonDiabetic -0.10401    0.82008  -0.127   0.8991
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 30.147  on 19  degrees of freedom
Residual deviance: 13.179  on 15  degrees of freedom
AIC: 62.146

Number of Fisher Scoring iterations: 5                                            [Max 4]
```

ii)

```
>
> #linear predictor for Model 1 is
> # a+b1X1+b2X2+b3X3+b4X1X2  where                                                     (1)
Alt: 0.17452 + .06235X1 -1.31680X2 + .02248X3 -.10401X1X2
> # X1 = 0 for Female Gender and 1 for Male Gender                                     (1)
> # X2 = 0 for Diabetic and 1 for Non Diabetic                                         (1)
> # X3 is Age                                                                          (0.5)
> # X1X2 indicates interaction term between Gender and Health Condition                (0.5)
>                                                                                  [Max 3]
```

iii)

```
> #iii.                                                                      (0.5)
> # Gender is not significant                                                  (1)
> # Health Condition is significant                                          (0.5)
> # Age is not significant                                                   (0.5)
> # Interaction term between Gender and Health condition is not significant

                                                                             (0.5)
> #Scaled deviance = 13.179
> #AIC = - 2LogL(Model) + 2*Parameters
> #LogL(Model) = Parameters - AIC/2                                            (2)
>
> L<- 4- model1$aic/2
> L
[1] -27.07302
>
> #Log Likelihood of Model1 is -27.07302                                   [Max 4]
>
```

iv)

```
> #iv.
> model2<-glm(Claim~Health+Age,family=poisson(lin="log"),data=PA)              (1)
>
> model2$aic < model1$aic
[1] TRUE
> # Model 2 AIC is lower than Model 1 showing Model2 outperforms Model1      (1.5)
                                                                           [Max 2]
```

v)

```
> #v.                                                                          (1)
> summary(model2)

Call:
glm(formula = Claim ~ Health + Age, family = poisson(lin = "log"),
   data = PA)

Deviance Residuals:
    Min       1Q    Median       3Q      Max
-1.38527  -0.76449   0.06081   0.36914   1.40103

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)       0.21958    0.87208   0.252 0.801208
HealthNonDiabetic -1.38710    0.39059  -3.551 0.000383 ***
Age               0.02252    0.02037   1.106 0.268841
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

   Null deviance: 30.147  on 19  degrees of freedom
Residual deviance: 13.201  on 17  degrees of freedom
AIC: 58.168

Number of Fisher Scoring iterations: 5
                                                                             (1.5)
> #Age is not significant and thus can be dropped to improve the model
```

Give full marks in case reached to same conclusion using alternate methods.

(1.5)

```
> model3<-glm(Claim~Health,family=poisson(lin="log"),data=PA)
> summary(model3)

Call:
glm(formula = Claim ~ Health, family = poisson(lin = "log"),
    data = PA)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.22474  -0.81754  -0.07119   0.27453   1.44149

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)         1.1394     0.2000   5.697 1.22e-08 ***
HealthNonDiabetic  -1.4271     0.3887  -3.671 0.000241 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 30.147  on 19  degrees of freedom
Residual deviance: 14.436  on 18  degrees of freedom
AIC: 57.403

Number of Fisher Scoring iterations: 5

> model3$aic < model2$aic
[1] TRUE
>
> # Under Model 3. AIC reduced from 58.17 to 57.40                              (1)
>                                                                          [Max 5]
```

vi)

```
> #vi.
> Student1<- data.frame(Gender ="M",Health ="Diabetic",Age=30)
> Actuary2<- data.frame(Gender ="F",Health ="NonDiabetic",Age=50)               (1)
>                                                                           (0.5)
> # Price = 5000 * expected number of claims                                    (1)
> # Price of student1 is 12240.06 for Model 2 and 15625 for Model 3
> 5000*predict.glm(model2,newdata = Student1,type= "response")                  (1)
       1
12240.06                                                                    (0.5)
> 5000*predict.glm(model3,newdata = Student1,type= "response")
    1
15625
> # Price of Actuary2 is 4797.457 for Model 2 and 3750 for Model 3            (0.5)
> 5000*predict.glm(model2,newdata = Actuary2,type= "response")
       1
4797.457                                                                    (0.5)
> 5000*predict.glm(model3,newdata = Actuary2,type= "response")
   1
3750
>
> #Under both models, Student1 price is coming higher.                          (2)
> #Since student is diabetic. Under both models, health condition
> # is significant and for non-diabetic, parameter value is negative
> #implying lower claims for non-diabetic                                  [Max 7]
>
```

vii) a)

```
> #vii. a.
>
> Student1_mean=predict.glm(model3,newdata = Student1,type= "response")          (0.5)
> Actuary2_mean=predict.glm(model3,newdata = Actuary2,type= "response")          (0.5)
>
> #price in case of modified product is
> # 4000* expected number of claims + 2000 * probability of 1 or more claim.     (0.5)
>
> #price of Student1:
>
> #compute probability of 0                                                      (1.5)
> sp0<-dpois(0,Student1_mean)
>
> 4000*Student1_mean + 2000*(1-sp0)                                              (2)
       1
14412.13
>
> #price for Actuary2
> ap0<-dpois(0,Actuary2_mean)                                                    (1)
> 4000*Actuary2_mean + 2000*(1-ap0)                                              (1)
       1
4055.267                                                                     [Max 6]
```

b)

```
> #vii. b.
> Student1_mean
    1
3.125
> #For student1, expected claims are 3.125. Since, more claims are expected
> # for Student1, reduction of payment lead to more saving than                  (1)
> # extra payment for 1st claim and thus, less price
>
> Actuary2_mean
   1
0.75
> #Whereas for Actuary2, expected claims are 0.75 close to 1.                    (1)
> #thus, more payment expected for Actuary2 resulting in higher price.       [Max 2]
```

[33 Marks]

Solution 4:

i) a) We define two data sets w_draw and r_draw corresponding to the white balls and red balls to be matched for cases A to I.

R Code and Output:

```
> w_draw <- c(5,5,4,4,3,3,2,1,0) # white balls to be matched corresponding to cases A to I     (1)

> r_draw <- c(1,0,1,0,1,0,1,1,1) # red balls to be matched corresponding to cases A to I       (1)
```

[2]

b) Formula for the probability mass function of Hyper-Geometric Distribution in terms of the arguments specified in the question is given below:

$P(X=x) = {}^{m}C_{x} * {}^{n}C_{(k-x)} / {}^{(m+n)}C_{k}$

Where –

x = 0, 1, 2, 3, ………………
0 < p < 1
(p = m/(m+n)) [2]

c) Further we define variables prob_w_draw and prob_r_draw to determine the probabilities under hypergeometric distribution where w_draw and r_draw will be used as input, m will be the successes (5 and 1), n will be the failures (64 and 25) and k will be the number of balls drawn i.e. sample size (5 and 1)

R Code and Output:

```
> prob_w_draw <- dhyper(w_draw,m=5,n=64,k=5)
> print(prob_w_draw)
[1] 8.897974e-08 8.897974e-08 2.847352e-05 2.847352e-05 1.793832e-03
[6] 1.793832e-03 3.707252e-02 2.826780e-01 6.784271e-01
```

(2)

```
> prob_r_draw <- dhyper(r_draw,m=1,n=25,k=1)
> print(prob_r_draw)
[1] 0.03846154 0.96153846 0.03846154 0.96153846 0.03846154 0.96153846
[7] 0.03846154 0.03846154 0.03846154
```

(2)

[Max 4]

d) Finally, we define prob_draw to be the multiplication of prob_w_draw and prob_r_draw assuming the draws are independent of each other.

R Code and Output:

```
> prob_draw = prob_w_draw * prob_r_draw # multiplication of probabilities
> print(prob_draw)
[1] 3.422298e-09 8.555745e-08 1.095135e-06 2.737838e-05 6.899352e-05 1.724838e-03
[7] 1.425866e-03 1.087223e-02 2.609335e-02
```

[2]

e) We define another data set prize which defines the prize amounts for Cases A to I.

R Code and Output:

```
>
> prize <- c(20000000,1000000,50000,100,100,7,7,4,4)
> print(prize)
[1] 2e+07 1e+06 5e+04 1e+02 1e+02 7e+00 7e+00 4e+00 4e+00
```

[1]

f) We then define amt by multiplying prob_draw with prize to arrive at the expected amount one can win from the lottery / jackpot.

R Code and Output:

```
> amt = prize * prob_draw
> print(amt)
[1] 0.068445956 0.085557445 0.054756765 0.002737838 0.006899352 0.012073867
[7] 0.009981063 0.043488918 0.104373403
> print(sum(amt))
[1] 0.3883146
```

[2]

g) Since, the expected prize amount is INR 0.39 and the prize of the lottery ticket is INR 0.50, there is a profit of INR 0.11 implicit in the ticket price. [1]

ii)a) Formula for the probability mass function of Binomial Distribution in terms of the arguments specified in the question is given below:

$P(X=x) = {}^{k}C_{x} * (p)^{x} * (1-p)^{(k-x)}$

Where –
x = 0, 1, 2, 3, ………………
0 < p < 1
(p = m/(m+n)) [2]

b) The joint probabilities and the expected prize money pay-out is re-determined using binomial distribution for determining matching probabilities:

R Code and Output:

```
> prob_w_draw_bin <- dbinom(w_draw,5,5/69,log=FALSE)
> print(prob_w_draw_bin)
[1] 1.998042e-06 1.998042e-06 1.278747e-04 1.278747e-04 3.273592e-03
[6] 3.273592e-03 4.190197e-02 2.681726e-01 6.865219e-01
```

(1.5)

```
> prob_r_draw_bin <- dbinom(r_draw,1,1/26,log=FALSE)
> print(prob_r_draw_bin)
[1] 0.03846154 0.96153846 0.03846154 0.96153846 0.03846154 0.96153846
[7] 0.03846154 0.03846154 0.03846154
```

(1.5)

```
> prob_draw_bin = prob_w_draw_bin * prob_r_draw_bin
> print(prob_draw_bin)
[1] 7.684776e-08 1.921194e-06 4.918257e-06 1.229564e-04 1.259074e-04
[6] 3.147684e-03 1.611614e-03 1.031433e-02 2.640469e-02
```

(1)

```
> amt_bin = prize * prob_draw_bin
> print(amt_bin)
[1] 1.53695522 1.92119403 0.24591284 0.01229564 0.01259074 0.02203379
[7] 0.01128130 0.04125733 0.10561876
> print(sum(amt_bin))
[1] 3.90914
```

(1)

[Max 4]

iii) If we compare binomial probabilities with hyper-geometric probabilities: (1)

1. For the power ball, the probabilities for all cases are same under both binomial and hyper-geometric distribution. (1)
2. For white balls, the probabilities under binomial distribution tend to be higher than those under hyper-geometric distribution. (1)
3. Due to this the expected pay-out determined using binomial distribution tends to be on the higher side as compared to one determined using hyper-geometric distribution. **[Max 2]**

iv) We use the formulae from the tables to determine the mean and variance for X using both binomial and hyper-geometric distributions.

R Code and Output:

```
> mean_x_hyper = 5 * 5/69
> mean_x_binom = 5 * 5/69
> print(mean_x_hyper)
[1] 0.3623188
> print(mean_x_binom)
[1] 0.3623188
>
```

(1 marks for mean)

```
> var_x_hyper = 5*5*(69-5)*(69-5)/((69-1)*69^2)
> print(var_x_hyper)
[1] 0.3162954
> var_x_binom = 5 * 5/69 * (1-5/69)
> print(var_x_binom)
```

(1 marks for variance)

```
[1] 0.3360639
```

Mean is the same in both cases.

However, variance under hyper-geometric distribution is lower as compared to the variance under the binomial distribution. This is because hyper-geometric distribution is a without replacement alternative of the binomial distribution. Since, after a trial, that observation is not replaced, the variability in the results is reduced.

(2 mark for comment on the difference.)

[Max 3]

v) In reality, for lotteries, draws are done without replacement and hence hyper-geometric distribution would be more suitable in modelling matching probabilities. (1)

However, as the size of the population (m+n) goes on increasing, binomial distribution provides a good approximation for hyper-geometric probabilities. (1)

[2]

[27 Marks]

****