Ishaan Taneja

Grading:
30 marks- Complete submission of Problem 1,2
70 marks- Problem 1

1. Suppose p is the unknown probability of an event A, and we estimate p by the sample proportion $\hat{p}$ based on an i.i.d. sample of size n.
   (a) Write $Var[\hat{p}]$ and $SD[\hat{p}]$ as functions of n and p.
   (b) Using the relations derived above, determine the sample size n, as a function of p, that is required to acheive $SD(\hat{p}) = 0.01$. How does this required value of n vary with p?
   (c) Design and implement the following simulation study to verify this behaviour. For $p = 0.01$, 0.1, 0.25, 0.5, 0.75, 0.9, and 0.99,
      (i) Simulate 1000 values of $\hat{p}$ with $n = 500$.
      (ii) Simulate 1000 values of $\hat{p}$ with n chosen according to the formula derived above.

      In each case, you can think of the 1000 values as i.i.d. samples from the distribution of $\hat{p}$, and use the sample standard deviation as an estimate of $SD[\hat{p}]$. Plot the estimated values of $SD(\hat{p})$ against p for both choices of n.

Solution: 1

(a) Let X_1, X_2,...,X_n be i.i.d sample of size n.
The sample proportion p is given by $\hat{p} = \dfrac{\#\{X_i \in A\}}{n}$
Let,

$$Z_i = \begin{cases} 1 & ; \text{if } X_i \in A \\ 0 & ; \text{otherwise} \end{cases}$$

Therefore, $P(Z_i = 1) = P(X_i \in A) = p$ and $P(Z_i = 0) = 1 - p$
Hence, $Z_i \sim Bernoulli(p)$; i=1,2,...,n
Let us define a random variable, $Y = \sum_{i=1}^{n} Z_i \sim Binomial(n, p)$
So, $\hat{p} = \dfrac{Y}{n}$

$$\begin{aligned} Var(\hat{p}) &= Var\left(\frac{Y}{n}\right) \\ &= \frac{1}{n^2} Var(Y) \\ &= \frac{1}{n^2} np(1-p) \\ \therefore Var(\hat{p}) &= \frac{p(1-p)}{n} \end{aligned}$$

$$\text{And,}\quad S.D(\hat{p}) = \sqrt{Var(\hat{p})}$$
$$\therefore S.D(\hat{p}) = \sqrt{\frac{p(1-p)}{n}}$$

(b)

$$S.D(\hat{p}) = 0.01$$
$$\sqrt{\frac{p(1-p)}{n}} = 0.01$$
$$\frac{p(1-p)}{n} = 0.0001$$
$$\therefore n = f(p) = 10000 \times p(1-p)$$
$$f'(p) = 1000(1-2p)$$

Since,

$$f'(p) = \begin{cases} < 0 & ;\text{ if } p > 1/2 \\ > 0 & ;\text{ if } p < 1/2 \end{cases}$$

Therefore, n increases with increase in $p \in [0, 0.5]$ and n decreases with increases in $p \in [0.5, 1]$

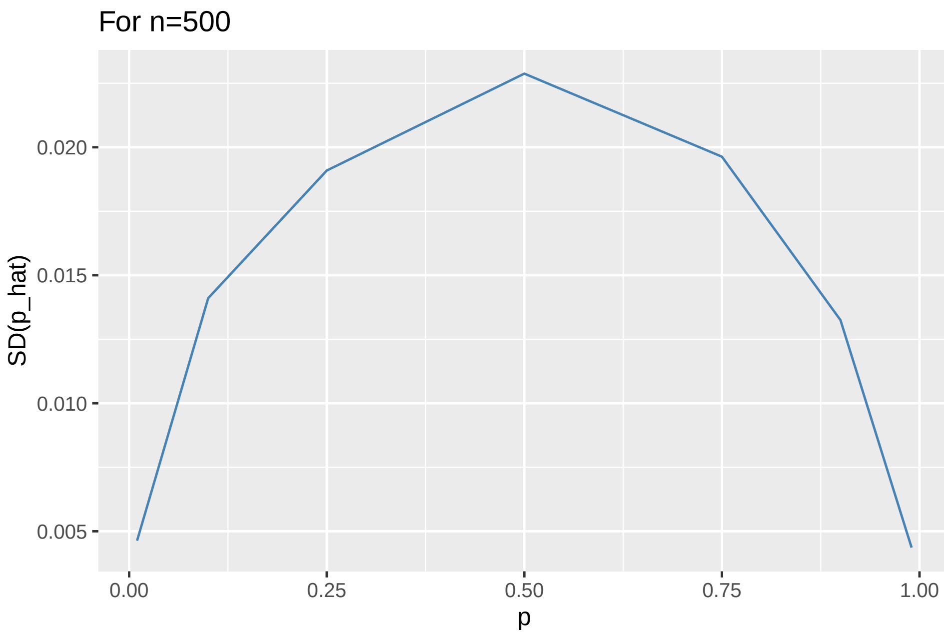
(c) (i)

```
> s_sd_1=function(p){
+   s_1=c()
+   for (i in 1:length(p)){
+     p_hat=rbinom(1000,500,p[i])/500
+     s_1[i]=sd(p_hat)
+   }
+   return(s_1)
+ }
```

(ii)

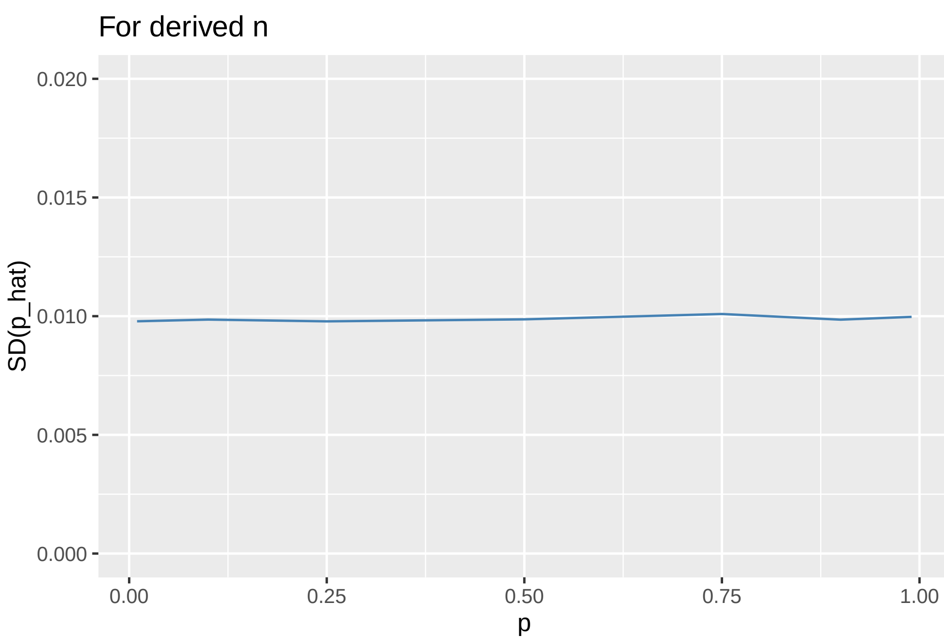
```
> s_sd_2=function(p){
+   s_2=c()
+   for (i in 1:length(p)){
+     n=round(10000*p[i]*(1-p[i]))
+     p_hat=rbinom(1000,n,p[i])/n
+     s_2[i]=sd(p_hat)
+   }
+   return(s_2)
+ }
```

```
> p=c(0.01,0.1,0.25,0.5,0.75,0.9,0.99)
> df=data.frame(p,s_sd_1(p),s_sd_2(p))
> colnames(df)<-c('p','s_1','s_2')
> library(ggplot2)
> ggplot(df,aes(x=p))+geom_line(aes(y=s_1),colour='steelblue')+
+   labs(x='p',y='SD(p_hat)')+ggtitle("For n=500")
```



We observe that the $SD(\hat{p})$ increases till p=0.5 and then decreases.

```
> ggplot(df,aes(x=p))+geom_line(aes(y=s_2),colour='steelblue')+
+   labs(x='p',y='SD(p_hat)')+ggtitle("For derived n")+
+   coord_cartesian(ylim=c(0,0.02))
```



The value of $SD(\hat{p})$ remain close to 0.01 as 'n' is derived using that formula only.

2. Consider Poisson λ distribution.

   (a) Show that both the sample mean and the sample variance of a sample obtained from the Poisson(λ) distribution will be unbiased estimators of λ.

   (b) For $\lambda = 10, 20, 50$ simulate $100, 500, 1000$ random observations from the Poisson(λ) distribution for various values of λ using the inbuilt function `rpois`.

(c) Explore the behaviour of the two estimates for each λ as well as three sample sizes.

Solution: 2

(a) Let X_1, X_2,...,X_n be an i.i.d sample of size n from Poisson(λ) distribution.

Sample mean ; $\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$

Sample variance ; $S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2$

Now,

$$
\begin{aligned}
E(\overline{X}) &= E\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) \\
&= \frac{1}{n}\sum_{i=1}^{n} E(X_i) \\
&= \frac{1}{n}\sum_{i=1}^{n} \lambda \\
&= \frac{1}{n} n\lambda \\
\therefore E(\overline{X}) &= \lambda
\end{aligned}
$$

Hence, sample mean is an unbiased estimator of λ. And,

$$
\begin{aligned}
E(S^2) &= E\left(\frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2\right) \\
&= \frac{1}{n-1} E\left(\sum_{i=1}^{n}(X_i^2 + \overline{X}^2 - 2\overline{X}X_i)\right) \\
\therefore E(S^2) &= \frac{1}{n-1}\left(\sum_{i=1}^{n} E(X_i^2) - nE(\overline{X}^2)\right)
\end{aligned}
$$

We know that,
$E(X_i^2) = V(X_i) + [E(X_i)]^2 = \lambda + \lambda^2$
and, $E(\overline{X}^2) = V(\overline{X}) + [E(\overline{X})]^2 = \frac{\lambda}{n} + \lambda^2$
Therefore,

$$
\begin{aligned}
E(S^2) &= \frac{1}{n-1}\left(\sum_{i=1}^{n}(\lambda + \lambda^2) - n(\frac{\lambda}{n} + \lambda^2)\right) \\
&= \frac{1}{n-1}(n\lambda + n\lambda^2 - \lambda - n\lambda^2) \\
&= \frac{1}{n-1}(n-1)\lambda \\
\therefore E(S^2) &= \lambda
\end{aligned}
$$

Hence, sample variance is an unbiased estimator of λ.

(b)
```
> n=c(100,500,1000)
> mean=c()
> d_mean=c()
> variance=c()
> d_variance=c()
> pois=function(lambda){
+   for(i in n){
+     p=rpois(i,lambda)
+     mean=append(mean,mean(p))
+     variance=append(variance,var(p))
+     d_mean=append(d_mean,lambda-mean(p))
+     d_variance=append(d_variance,lambda-var(p))
+   }
+   return(data.frame(n,mean,variance,d_mean,d_variance))
+ }
```

(c)
```
> #Lambda = 10
> pois(10)

     n   mean  variance d_mean  d_variance
1  100  9.740  9.224646  0.260  0.77535354
2  500 10.144  9.446156 -0.144  0.55384369
3 1000 10.188 10.084741 -0.188 -0.08474074

> #Lambda = 20
> pois(20)

     n   mean variance d_mean d_variance
1  100 19.980 23.73697  0.020 -3.7369697
2  500 19.912 19.14655  0.088  0.8534509
3 1000 20.031 19.18923 -0.031  0.8107718

> #Lambda = 50
> pois(50)

     n   mean variance d_mean d_variance
1  100 49.380 39.69253  0.620 10.3074747
2  500 50.136 48.47445 -0.136  1.5255471
3 1000 50.352 49.85395 -0.352  0.1460501
```

From the above tables, we can notice that with increase in the value of n, the difference between the sample mean and true mean (λ) and the sample variance and true variance (λ) is decreasing.

3. Biologists use a technique called "capture-recapture" to estimate the size of the population of a species that cannot be directly counted.

   Suppose the unknown population size is N, and fifty members of the species are selected and given an identifying mark. Sometime later a sample of size twenty is taken from the population, and it is found to contain X of the twenty previously marked. Equating the proportion of marked members in the second sample and the population, we have $\frac{X}{20} = \frac{50}{N}$, giving an estimate of $\hat{N} = \frac{1000}{X}$.

(a) Show that the distribution of X has a hypergeometric distribution that involves N as a parameter.

(b) Using the function `rhyper`. For each $N = 50$, 100, 200, 300, 400, and 500, simulate 1000 values of $\hat{N}$ and use them to estimate $E[\hat{N}]$ and $Var[\hat{N}]$. Plot these estimates as a function of N.

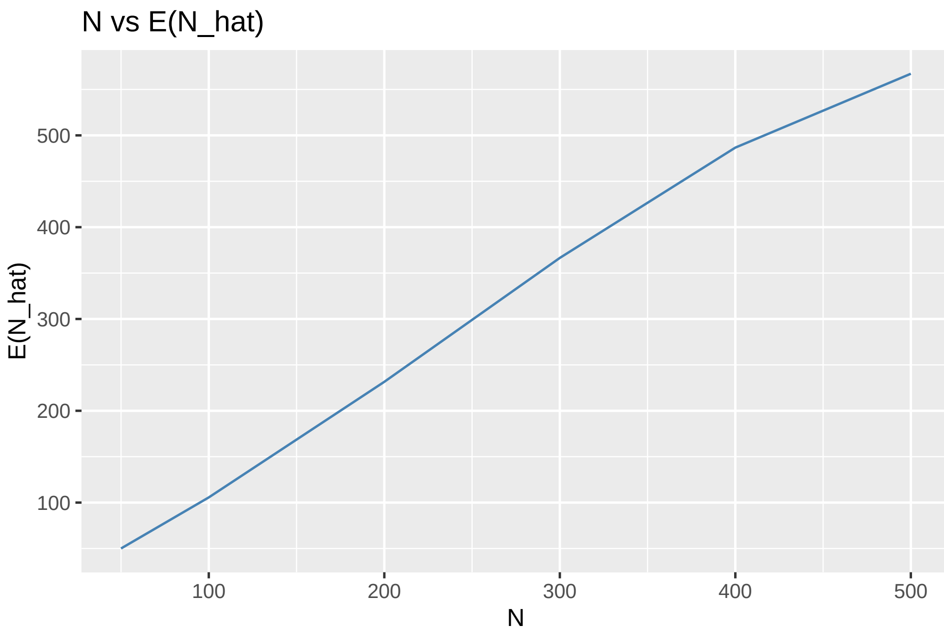
Solution: 3

(a) If the second sample is done at random and without replacement then,
Total population; N = N
Number of objects with favorable feature; K = 50
Number of draws; n = 20
Number of observed successes = k
X represents the number of marked member of that species, in a sample of 20 taken randomly and without replacement.

$$\therefore P(X = k) = \frac{{}^{K}C_k \times {}^{N-K}C_{n-k}}{{}^{N}C_n} \quad ; max(0, n + K - N) \leq k \leq min(K, n)$$

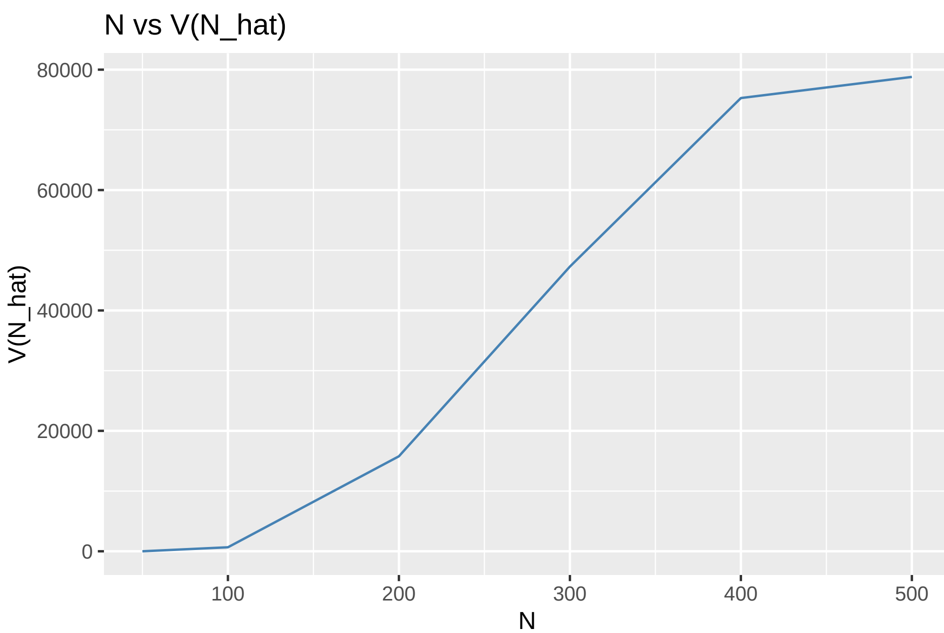
i.e. $X \sim Hypergeometric(N, 50, 20)$

(b)
```
> n=c(50,100,200,300,400,500)
> N_hat_mean=c()
> N_hat_var=c()
> N_hat=c()
> for(j in n){
+   N=c()
+   X=c()
+
+     X=rhyper(1000,50,j-50,20)
+     X[X==0] <- 50*20/j #Replacing 0 values esle N_hat=Inf
+     N=1000/X
+
+     N_hat_mean=append(N_hat_mean,mean(N))
+     N_hat_var=append(N_hat_var,var(N))
+ }
> library(ggplot2)
> df=data.frame(n,N_hat_mean,N_hat_var)
> colnames(df)<-c('N','E(N_hat)','V(N_hat)')
> df

    N E(N_hat)   V(N_hat)
1  50  50.0000     0.0000
2 100 105.6388   661.6081
3 200 231.5589 15776.7240
4 300 366.5187 47283.6089
5 400 486.6825 75281.4487
6 500 567.1429 78792.6248

> ggplot(df,aes(x=n))+geom_line(aes(y=N_hat_mean),colour='steelblue')+
+   labs(x='N',y='E(N_hat)')+ggtitle("N vs E(N_hat)")
```



From the plot, we can see that the $E(\hat{N})$ is close to the N.

```
> ggplot(df,aes(x=n))+geom_line(aes(y=N_hat_var),colour='steelblue')+
+   labs(x='N',y='V(N_hat)')+ggtitle("N vs V(N_hat)")
```



From the plot, we can see that with increase in N, the $Var(\hat{N})$ increases sharply.