Shenduo Zhang

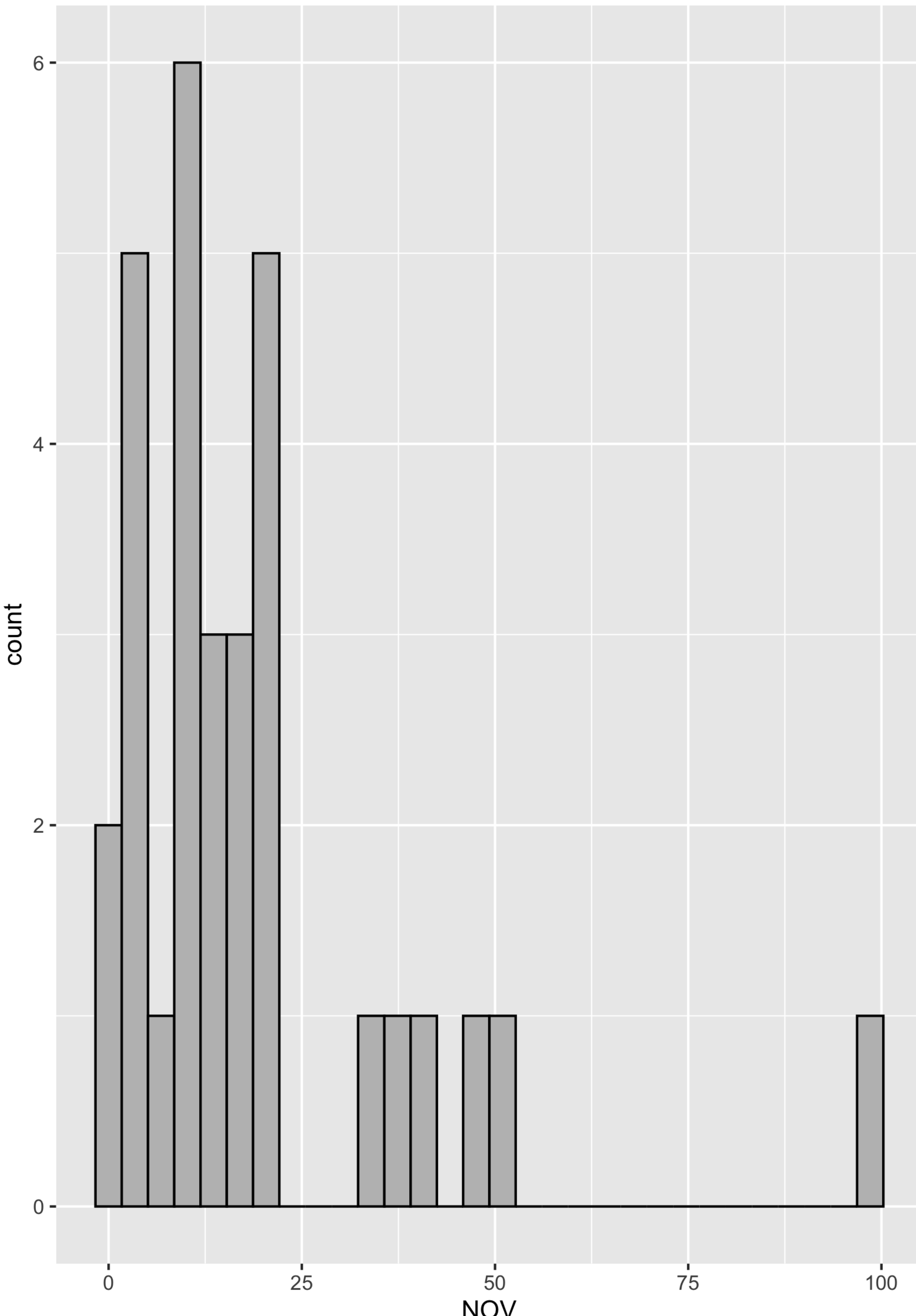
I omit the library and read.table calls because the tex template do not work in Chinese while the file path are in Chinese. Two '.txt' files are imported with the read.table with the original name "exersice1_4" and "exersice1_7". One would be expected to have packages ggplot2, Hmisc, moments and their dependency installed if they like to see the code running.

Problem 1

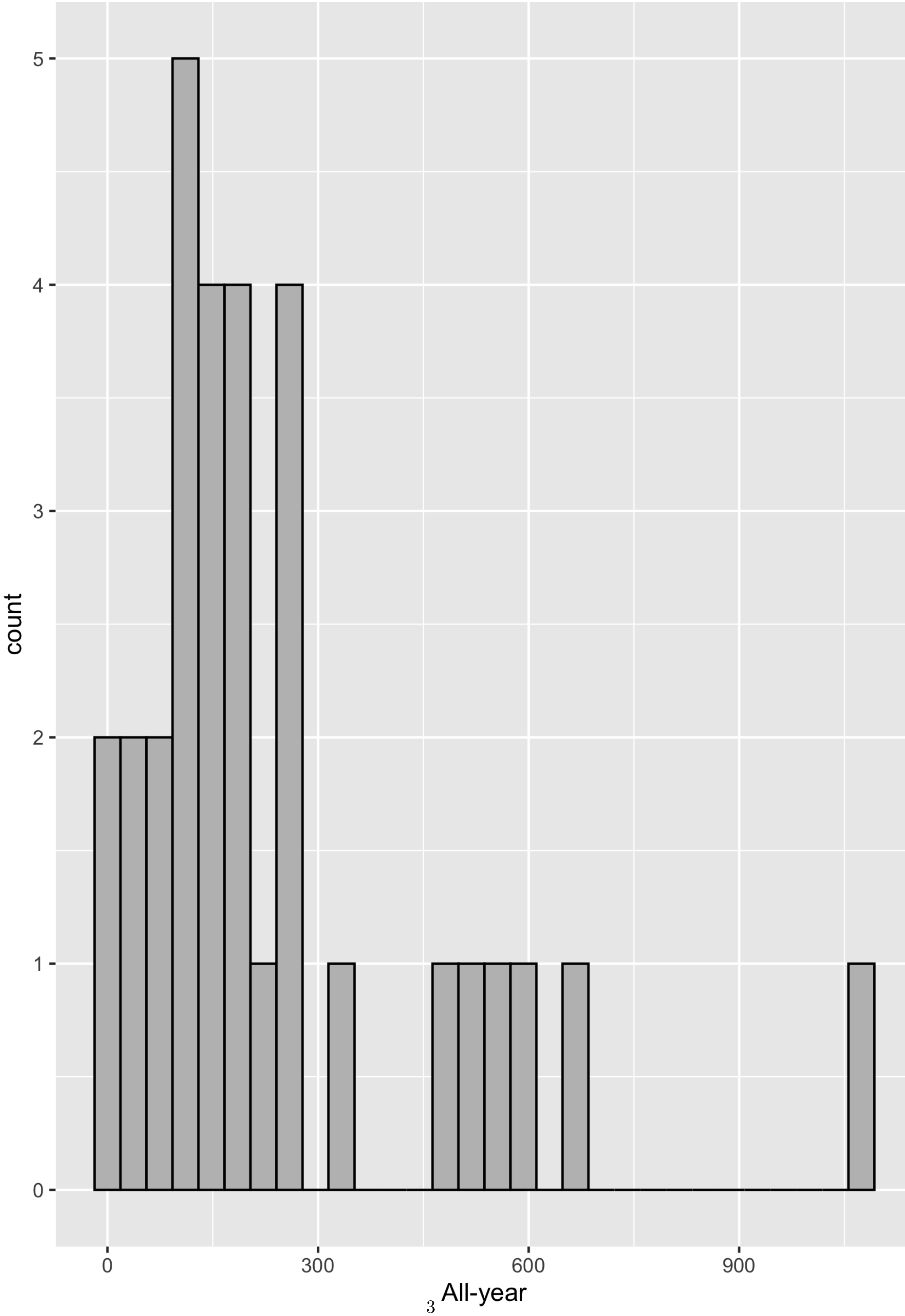
***Solution* 1.a**

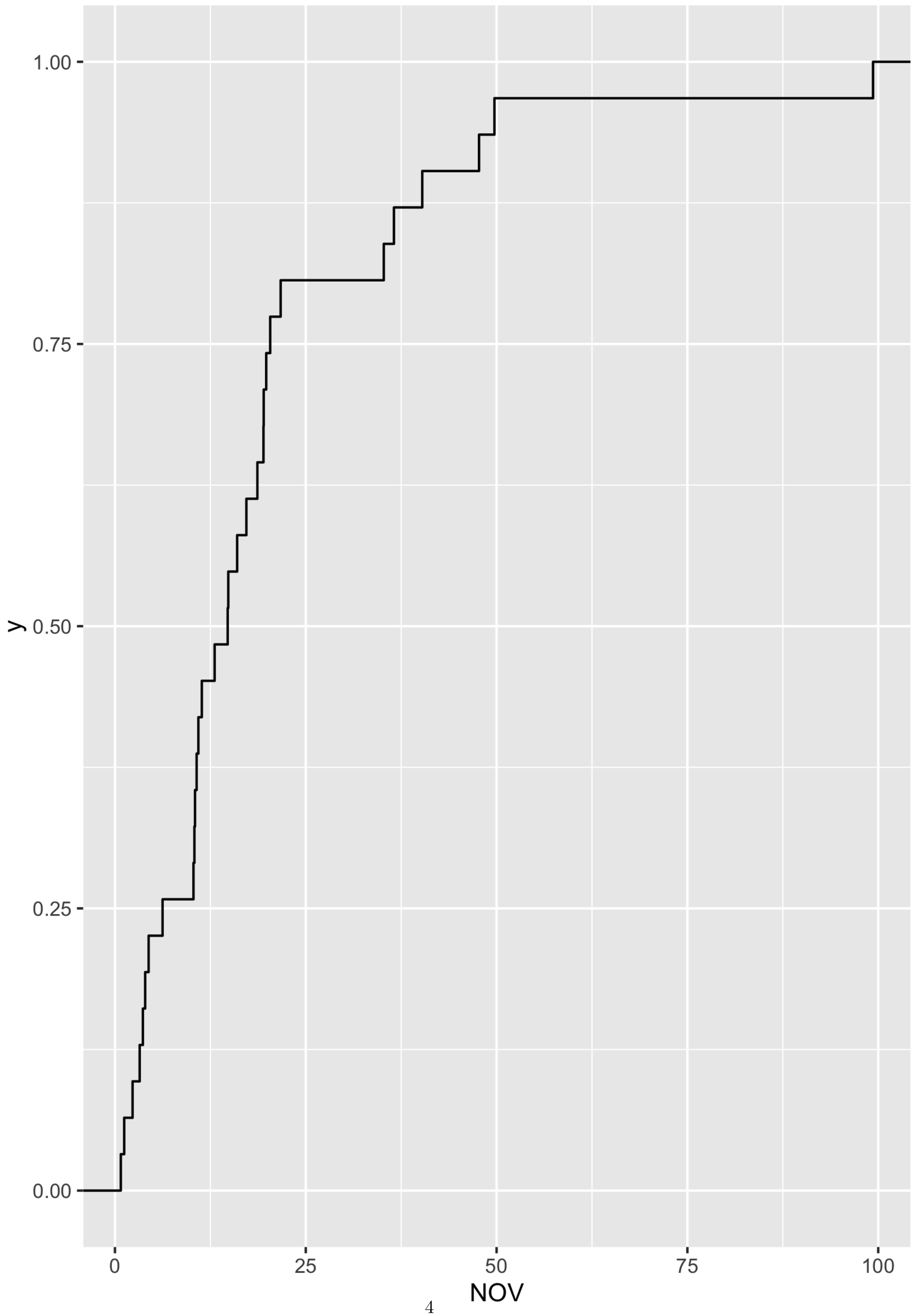
I would gave my result ahead to make everyone's life easier. The vector stands for the result of applying the certain function to X_1, X_2 respectively.

$$\begin{aligned}
\text{Mean}(X_1, X_2) &= (19.2, 246.2) \\
\text{Var}(X_1, X_2) &= (392.0308, 54276.9982) \\
\text{Standard deviation}(X_1, X_2) &= (19.79977, 233.97210) \\
\text{Coefficients of variation}(X_1, X_2) &= (1.033043, 12.155202) \\
\text{Skewness}(X_1, X_2) &= (2.391967, 1.821974) \\
\text{Kurtosis}(X_1, X_2) &= (9.804995, 6.521676) \\
\text{Median}(X_1, X_2) &= (14.77, 179.41) \\
\text{Quantile}(0.25, 0.75)X_1 &= (8.265, 20.080) \\
\text{Quantile}(0.25, 0.75)X_2 &= (105.350, 270.745) \\
\text{Interquantilerange}(X_1, X_2) &= (11.815, 165.395) \\
Correlation_{peason} &= 0.9762474 \\
Correlation_{spearson} &= 0.9278226
\end{aligned}$$

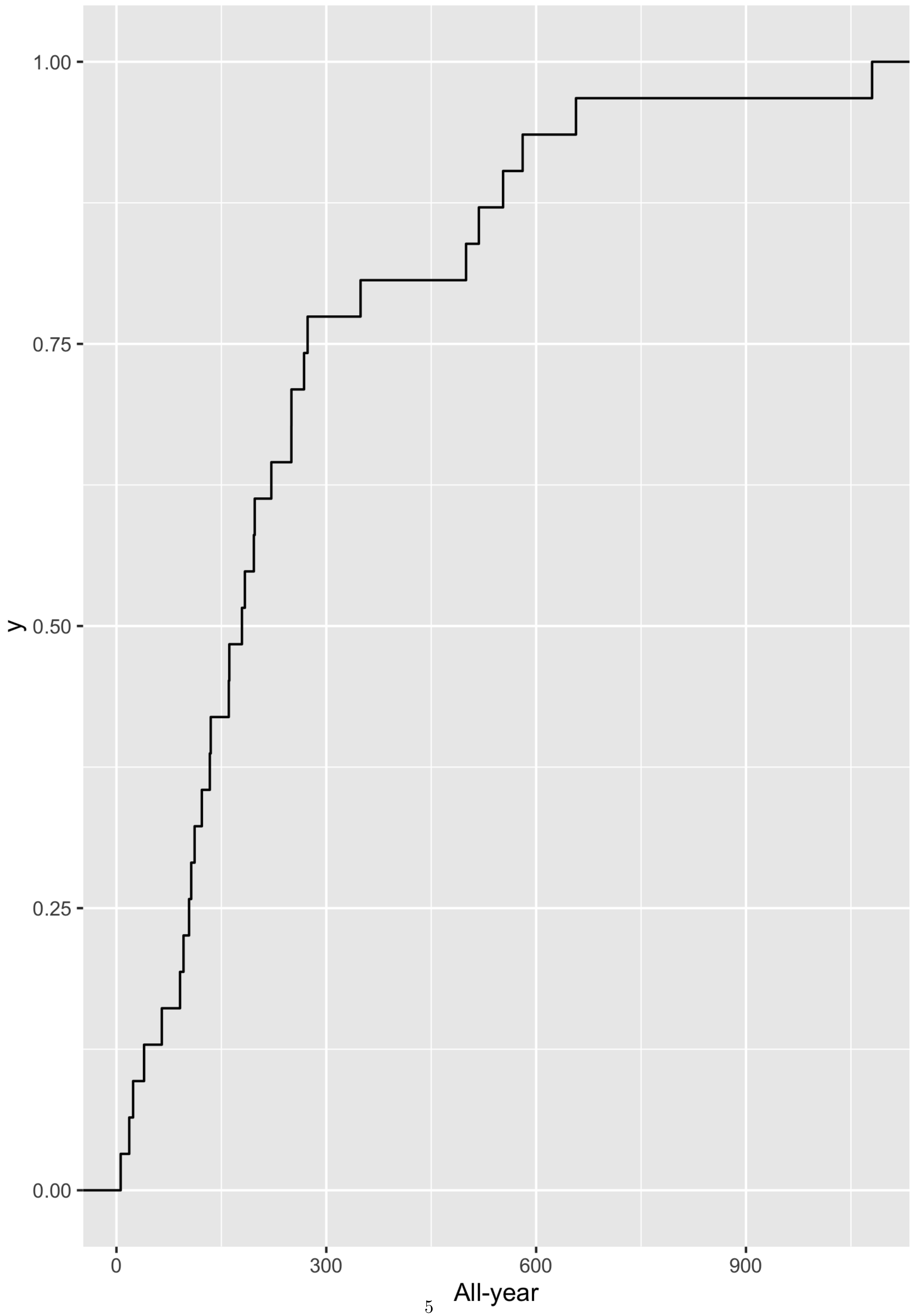


6
4
2
0
count
0
25
50
75
100
NOV


count
5
4
3
2
1
0
0
300
600
900
All-year



1.00
0.75
0.50
0.25
0.00
y
0
25
50
75
100
NOV



1.00
0.75
0.50
0.25
0.00
y
0
300
600
900
All-year

```
> X_1 <- exersice1_4[,2]
> X_2 <- exersice1_4[,3]
> Mean <- c(mean(X_1, na.rm=FALSE),mean(X_1,na.rm=FALSE))
> Var <- c(var(X_1),var(X_2))
> Standard_Deviation <- c(sd(X_1, na.rm=FALSE),sd(X_2, na.rm=FALSE))
> Coefficents_of_variation <- Standard_Deviation/Mean
> Skewness <- c(skewness(X_1),skewness(X_2))
> Kurtosis <- c(kurtosis(X_1),kurtosis(X_2))
> Median <- c(median(X_1),median(X_2))
> QuantileX_1 = c(quantile(X_1, probs = seq(0, 1, 0.25)),quantile(X_1,0.75))
> QuantileX_2 = c(quantile(X_2, probs = seq(0, 1, 0.25)),quantile(X_2,0.75))
> Interquantilerange <- c(QuantileX_1[2]-QuantileX_1[1],QuantileX_2[2]-QuantileX_2[1] )
> Mean
[1] 19.16645 19.16645
> Var
[1]   392.0308 54275.9982
> Standard_Deviation
[1]  19.79977 232.97210
> Coefficents_of_variation
[1]  1.033043 12.155202
> Skewness
[1] 2.391967 1.821974
> Kurtosis
[1] 9.804995 6.521676
> Median
[1]  14.77 179.41
> QuantileX_1
    0%    25%    50%    75%   100%    75%
 0.770  8.265 14.770 20.080 99.320 20.080
> QuantileX_2
      0%      25%      50%      75%     100%      75%
   6.080  105.350  179.410  270.745 1080.260  270.745
> Interquantilerange
   25%    25%
 7.495 99.270
>
>
> ggplot(exersice1_4,aes(V2)) + geom_histogram(color="black",fill="gray")+xlab("NOV")
`stat_bin()` using `bins = 30`. Pick better value with
`binwidth`.
> ggsave("Plot1.png")
Saving 5.68 x 8.08 in image
`stat_bin()` using `bins = 30`. Pick better value with
`binwidth`.
> ggplot(exersice1_4,aes(V3)) + geom_histogram(color="black",fill="gray")+xlab("All-year")
`stat_bin()` using `bins = 30`. Pick better value with
`binwidth`.
> ggsave("Plot2.png")
Saving 5.68 x 8.08 in image
`stat_bin()` using `bins = 30`. Pick better value with
`binwidth`.
> ggplot(exersice1_4,aes(V2)) + stat_ecdf(geom = 'step')+xlab("NOV")
> ggsave("Plot3.png")
Saving 5.68 x 8.08 in image
> ggplot(exersice1_4,aes(V3)) + stat_ecdf(geom = 'step')+xlab("All-year")
> ggsave("Plot4.png")
Saving 5.68 x 8.08 in image
>
> Correlation_pearson <- cor(X_1,X_2,method='pearson')
> Correlation_spearman <- cor(X_1,X_2,method='spearman')
> Correlation_pearson
[1] 0.9762474
```

```
> Correlation_spearman
[1] 0.9278226
```

□

Problem 2

I believe this part of code is neat and self explained. By writing the result ahead won't help in any way of conveniance.

```
> apply(exercise1_7,MARGIN = 2,FUN=mean )
   V1    V2    V3
14.41 16.02  4.23
> apply(exercise1_7,MARGIN = 2,FUN=median )
V1 V2 V3
15 15  4
> rcorr(as.matrix(exercise1_7),type = 'spearman')
     V1   V2   V3
V1 1.00 0.55 0.51
V2 0.55 1.00 0.53
V3 0.51 0.53 1.00

n= 50


P
   V1     V2     V3
V1        0e+00  2e-04
V2 0e+00         0e+00
V3 2e-04  0e+00
> rcorr(as.matrix(exersice1_7),type = 'pearson')
     V1   V2   V3
V1 1.00 0.62 0.52
V2 0.62 1.00 0.46
V3 0.52 0.46 1.00

n= 50


P
   V1     V2     V3
V1        0e+00  1e-04
V2 0e+00         7e-04
V3 1e-04  7e-04
```

No matter which techniques we choose, the P-value is so small that with significance level as low as 0.001, one can infer that there's a linear correlation among those three featuresX_1, X_2, X_3.