

Defect Detection while Setting up an Assembly Line - Analytical Approach to Reduce the N-Dimensional Solution Space

By
Dr. Malolan Sundararaman

R-Code to Generate Bootstrapped Sampling Distribution and Determine the Cumulative Distribution Function

```
Data<- read.csv("**PATH TO FILE**\\**FILE_NAME**.csv", header = TRUE)
Data<- subset.data.frame(Data, select = c(-Code, -Distance))
CCC_TT<-as.vector(Data$Error_Code)
CCC_TT<-as.vector(unique(CCC_TT))
N_CCC_TT<-length(CCC_TT)
Data_Run<-data.frame()
df1<-data.frame()
df2<-data.frame()
Z<-data.frame()
P<-data.frame()
X<- data.frame()
Y<- data.frame()
YY<- data.frame()
B<-data.frame()
BB<-data.frame()
YY<- data.frame()
BB<-data.frame()
Q<-data.frame()
C<-data.frame()
KM<-data.frame()
Boot=100
percent=0.2
pb<- txtProgressBar(min = 1, max = N_CCC_TT, style = 3)
for(b in 1:N_CCC_TT){
  CCC_Chosen<-CCC_TT[b]
  Data_Run<-subset(Data, Error_Code==CCC_Chosen)
  N<- nrow(Data_Run)
  #print(N)
  for(V in (N*percent):N){
    #print(V)
    K<-ncol(Data_Run)
    for(i in 1:Boot){
      df1<- Data_Run[sample(1:nrow(Data_Run),V, replace=TRUE),]
      for(a in 3:K){
        df2<- df1[,c(1:a)]
        #print(a)
        df2<- df2[,c(-1,-2)]
        #print(df2)
        if(a==3){
          S<-sum(df2)
          if(S>0){S=1} else {S=0}
        }
        else {
          S<- sum(t(colSums(df2)))
          if(S>0){S=1} else {S=0}
        }
      }
      Z<-rbind(Z,S)
    }
  }
}
```

Defect Detection while Setting up an Assembly Line - Analytical Approach to Reduce the N-Dimensional Solution Space

By

Dr. Malolan Sundararaman

```
X<-rbind(X,a)
YY<-rbind(YY,V)
BB<-rbind(BB,b)
S<-c()
}
}
Q<-cbind(Z,X,YY,BB)
nms1<- c("sum","col","Vehicle","CCC")
Q<-setNames(Q,nms1)
col<-Q$col
Z<-data.frame()
X<-data.frame()
PMF<- round(aggregate(Q$sum, by=list(col), FUN=mean),digits = 1)
KM<-aggregate(Q$col, by=list(col), FUN=mean)
Y1<- aggregate(Q$Vehicle, by=list(col), FUN=mean)
B1<- aggregate(Q$CCC, by=list(col), FUN=mean)
P<-rbind(P,PMF)
C<- rbind(C,KM)
Y<-rbind(Y,Y1)
B<-rbind(B,B1)
Y1<- data.frame()
PMF<- data.frame()
KM<- data.frame()
Q<-data.frame()
col<-data.frame()
YY<-data.frame()
BB<-data.frame()
}
Sys.sleep(1)
setTxtProgressBar(pb,b)
}
Output<-cbind(B[,c(2)],Y[,c(2)],C[,c(2)],P[,c(2)])
write.csv(Output,"**FILE_LOCATION_PATH**\\**FILE_NAME**" )
```