Projet fraudes covid

LUCEA Lenny

2022-05-24

Sommaire

```r
#Fonction graphes
library(tidyr)
library(dplyr)
vec <- function(df){
  x <- c(df$new_cases)
  x <- x[x!=0]
  j=1
  for(i in 1:length(x)){
  if(is.na(x[j])){
    x <- x[-j]
  }
  else{
    j <- j +1
  }
  }
  return(x)
}
benf <- function(dd){
x <- vec(dd)
benlaw <- function(d) log10(1 + 1 / d)
digits <- 1:9
firstDigit <- function(x) substr(gsub('[0.]', '', x), 1, 1)
pctFirstDigit <- function(x) data.frame(table(firstDigit(x)) / length(x))
df <- pctFirstDigit(x)
if(length(x)==0|any(1==df)==F| any(2==df)==F | any(3==df)==F | any(4==df)==F | any(5==df)==F| any(6==df
    return()
}
else{
baseBarplot <- barplot(df$Freq[1:9], names.arg = digits, xlab = "First Digit",ylab="frequency", ylim =
lines(x = baseBarplot[,1], y = benlaw(digits), col = "red", lwd = 4,
      type = "b", pch = 23, cex = 1.5, bg = "red")
legend(x="topright", legend=c("Benford distribution","data distribution"),
       col=c("red","blue"), lty=1,lwd=4, cex=0.8)
}
}
```

```r
#fonction nom de pays pouvant passer le test
givenames <- function(df){
  name <- distinct(df,df$location)
  w <- 0
  x <- as.vector(name$`df$location`)
  t=0
  for(i in 1:length(x)){
    d <- df[df$location==x[i],]
    v <- vec(d)
    dx <- firstDigit(v)
    if(length(x)==0|any(1==dx)==F| any(2==dx)==F | any(3==dx)==F | any(4==dx)==F | any(5==dx)==F| any(6
      t=t
    }
    else{
      t=t+1
      w[t]=x[i]
    }
  }
```

```
  return(w)
}
```

```
# fonction T_2
library(BenfordSmoothTest)
```

```
## Le chargement a nécessité le package : gtools
```

```
## Le chargement a nécessité le package : polynom
```

```
T2 <- function(df){
  x <- vec(df)
  benf <- BenfordSmooth.test(x)
  return(benf)
}
```

Explication du T_2

Comme observé précédemment il existe de nombreux tests d'adéquation pour la détection de fraudes via la loi de Newcomb-Benford.Par soucis de performance on décidera pour la suite d'appliquer le test du T_2 considéré comme l'un des plus puissants parmis les tests d'adéquations lisses pour la loi de Newcomb-Benford.

Avant de rentrer dans le vif du sujet nous allons donc tout d'abord faire un point sur les tests pour la loi de newcomb-benford.

Fanny ..

Passons alors à une brève introduction du T_2.

Théorème Soit $X_1, \ldots, X_n$ des copies indépendantes d'une variable aléatoire X de densité $f(.)$ par rapport à une mesure v. Soit $\{h_0(.) := 1, h_k(.), k = 1, 2, ...\}$ une suite de fonctions orthonormales par rapport à $f(.)$; plus précisément, $\int h_k(x)h_{k'}(x)f(x)dv(x) = \delta_{kk'}$, la fonction delta de Kronecker. Soit $U_k = n^{-1/2}\sum_{i=1}^{n} h_k(X_i)$ et pour un entier $K \geq 1$, soir $T_k = \sum_{k=1}^{K} U_k^2$. Alors sous H_0, $T_K \xrightarrow{L} \chi^2_K$, la loi khi-deux à K degrés de liberté, et un test de niveau asymptotique α rejette H_0 si la valeur observée de T_K dépasse $x^2_{K,1-\alpha}$, le quantile d'ordre $1-\alpha$ de cette loi χ^2_K.

introduction sur notre base de donnée (expliquer les selections)

source : https://github.com/CSSEGISandData/COVID-19

Explication des procédés d'analyse

Nous décidons alors dans un premier temps d'appliquer le test du T_2 sur la fréquence de distribution du premier chiffre significatif des nouveaux cas quotidiens de COVID-19 rapportés par 189 pays.

(On considère aussi le monde , l'union européenne et l'europe).

Analyse des données observées (*p_value significative ou non*)

Après analyse, on observe qu'on rejette H_0 : l'échantillon suit une loi de Newcomb-Benford contre H_1 : l'échantillon ne suis pas une loi de Newcomb-Benford pour 74 pays ce qui explique que globalement au niveau mondial on rejette H_0. On remarque alors qu'on ne rejette pas H_0 pour les 115 autres pays avec un risque d'erreur $\alpha = 5\%$.

Cette étude nous permet de lever un drapeau d'alerte face aux pays qui ne passent pas le test comme nous pouvons le voir ci-dessous.

```
##
## Attachement du package : 'kableExtra'

## L'objet suivant est masqué depuis 'package:dplyr':
##
##     group_rows
```

Pays	T_2	p_value	Pays	T_2	p_value
Africa	14.371	0.0008	Japan	33.564	0
Armenia	10.703	0.0047	Kenya	9.038	0.0109
Asia	99.288	0.0000	Kiribati	6.385	0.0464
Australia	16.357	0.0003	Kosovo	6.05	0.0486
Austria	14.602	0.0007	Kuwait	24.557	0
Azerbaijan	8.743	0.0126	Latvia	27.235	0
Bahrain	10.094	0.0064	Libya	28.853	0
Belarus	19.117	0.0001	Lower middle income	17.737	1e-04
Bonaire Sint Eustatius and Saba	8.213	0.0165	Luxembourg	7.209	0.0272
Bulgaria	6.433	0.0401	Malaysia	21.142	0
Cambodia	12.878	0.0016	Maldives	12.581	0.0019
Chile	9.327	0.0094	Mongolia	8.426	0.0148
China	7.659	0.0217	Montenegro	6.489	0.039
Colombia	10.340	0.0057	Morocco	17.035	2e-04
Comoros	6.494	0.0372	Myanmar	19.548	1e-04
Croatia	7.417	0.0245	North Macedonia	9.672	0.0079
Cuba	39.368	0.0000	Oceania	31.464	0
Cyprus	16.007	0.0003	Pakistan	9.476	0.0088
Czechia	8.062	0.0178	Panama	10.474	0.0053
Denmark	8.889	0.0117	Poland	18.842	1e-04
Djibouti	16.094	0.0002	Portugal	6.203	0.045
Ecuador	7.150	0.0232	Qatar	22.296	0
Eritrea	11.361	0.0034	Russia	24.994	0
Estonia	18.337	0.0001	Serbia	52.819	0
Ethiopia	11.719	0.0029	Slovenia	7.191	0.0274
Europe	18.830	0.0001	South Africa	13.119	0.0014
European Union	11.449	0.0033	South Korea	13.121	0.0014
Faeroe Islands	8.910	0.0096	Sri Lanka	12.911	0.0016
Finland	11.423	0.0033	Syria	14.114	9e-04
Ghana	6.149	0.0464	Taiwan	18.496	1e-04
Gibraltar	15.726	0.0004	Trinidad and Tobago	41.511	0
Greece	16.425	0.0003	Turkey	12.823	0.0016
Guinea-Bissau	7.490	0.0224	United Arab Emirates	11.662	0.0029
Guyana	9.240	0.0099	United Kingdom	28.201	0
High income	13.666	0.0011	Uzbekistan	10.269	0.0059
Iceland	12.349	0.0036	Vanuatu	6.146	0.0434
Iraq	9.375	0.0092	Vietnam	56.115	0
Ireland	21.690	0.0000	World	11.188	0.0037
Israel	6.331	0.0422			

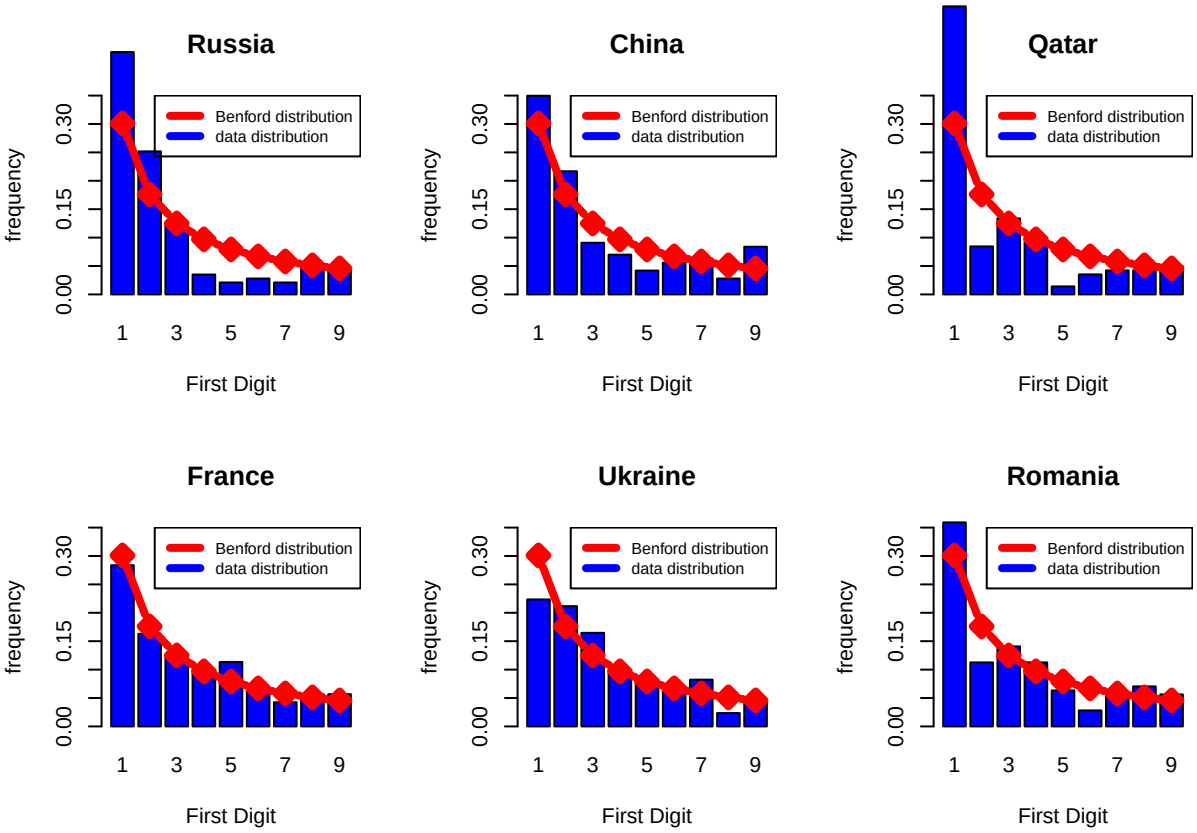
[1] Pays : nom des différents pays;

[2] T_2 : valeur de la statistique du T_2;

[3] p_value : p_value associée.

Figure 1

Parmi eux on retrouve notamment Cuba, le Qatar, la Russie, le Japon ou encore la Chine.


Russia
China
Qatar
France
Ukraine
Romania
Benford distribution
data distribution
frequency
0.00
0.15
0.30
1
3
5
7
9
First Digit