Ajustement d'une V.A

Abdessabour MOUTIK

2020-07-29

Table des matières

Table des figures

La variable aléatoire **X** a étudié est le nombre de sinistres. D'abord, nous importons le fichier `nombresinistre30.csv` a notre session R on utilisons la fonction `read.csv` :

```
db.nbrsin <- read.csv2(path)
```

On affecte la deuxième colone de `db.nbrsin` a notre vecteur `x`

```
x <- db.nbrsin[[2]]
```

On a l'espérance et la variance de X sont :

$$\bar{X} = 0.464347826086957 \quad \text{ET} \quad \hat{\sigma}_X^2 = 0.355840157346294$$

On remarque que

$$\bar{X} > \hat{\sigma}_X^2$$

donc la distribution la mieux ajustée peut être celle d'une loi **binomiale**.

J'appelle alors ma fonction `getgoodfit` qui retourne une liste contenant l'*estimation*, les *effectifs théorique*, le *vecteur de probabilté*, le *degré de liberté*, la *p-value*, et l'$\tilde{\chi}^2$ pour les distributions données dans la colonne `distributions` du résultat. Simultanément, cette fonction crée le diagramme à barres des *effectifs observés* et les rootogrammes suspendus pour chaque ajustement.

```
result <- getgoodfit(x, short=FALSE, plots.as.vars = TRUE)
```

```
## [1] "Error occured whilst estimating the size n = NA of the negative binomial distribution."
```

Les distributions utilisées sont :

```
result$distributions
```

```
## [1] "pois"   "binom"  "nbinom"
```

On trace l'histogramme des nombres de sinistres **X**.

```
result$Xplot
```

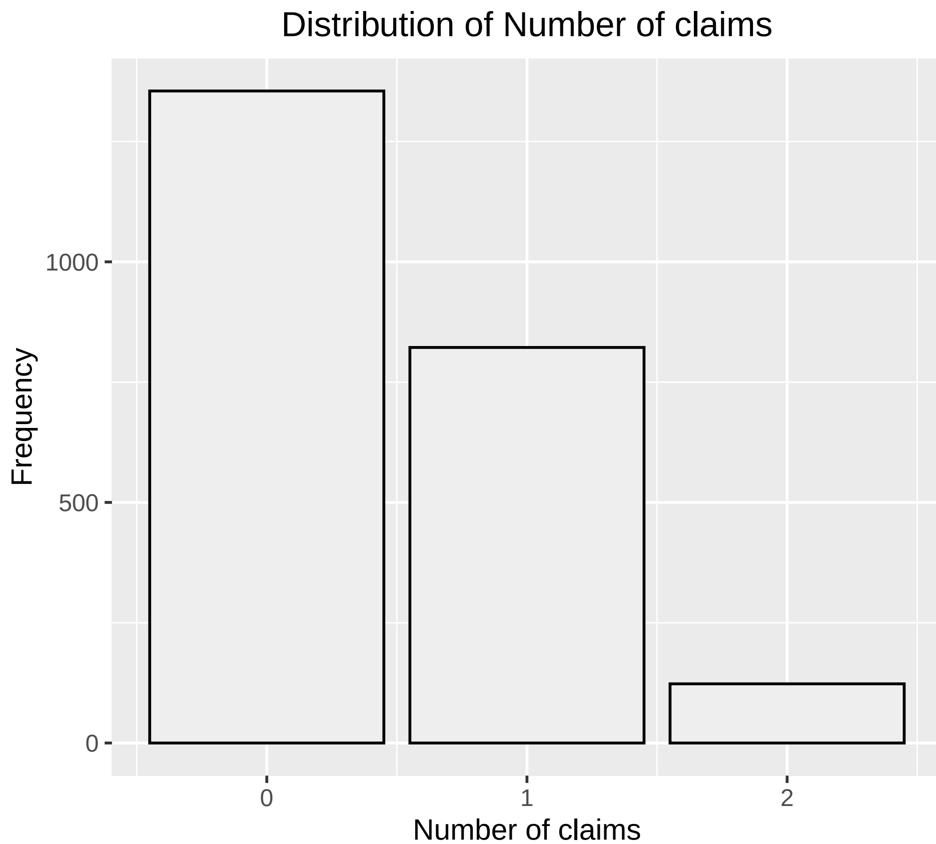

FIGURE 1 – Distribution des Nombre de sinistres

Ajustement par une distribution de loi de poisson

Après avoir utilisé `vcd::goodfit(x, "pois")` pour obtenir la distribution de poisson ajustée dans le corps de notre fonction `getgoodfit`. On retrouvera l'*estimation* suivantes:

```
result$pois$estimate
```

```
## $lambda
## [1] 0.4643478
```

On affiche le rootograme suspendu :

```
result$pois$plot
```

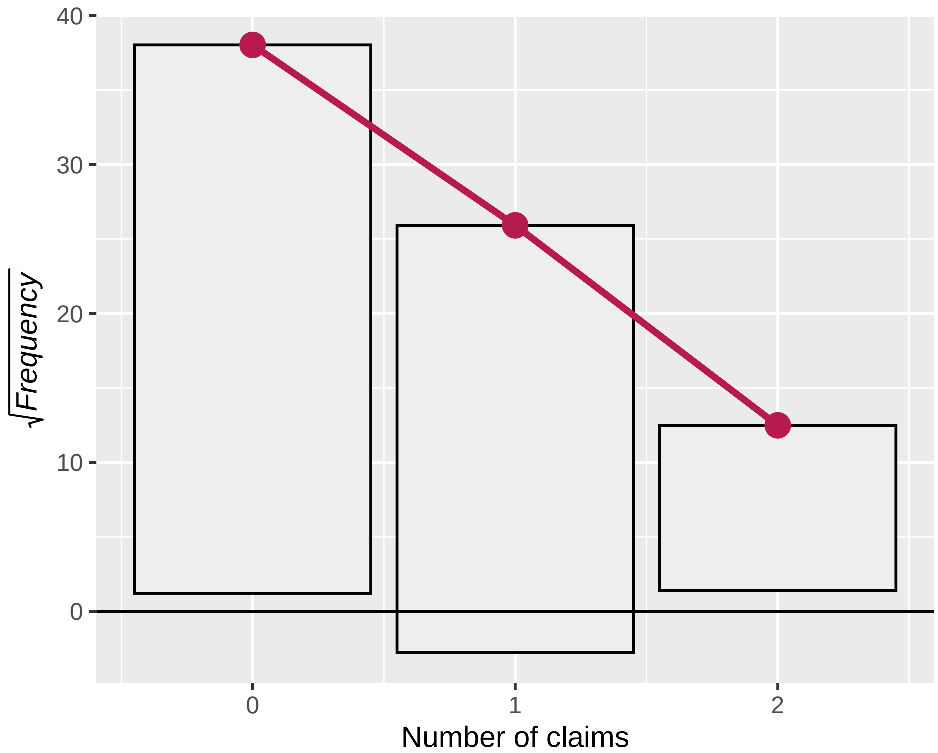

Figure 2 – Ajustement par une distrubition $\sim \mathcal{P}(0.46)$

On affiche les valeurs de **X**, les *effectifs théoriques*, les *effectifs observés* et le *vecteur de probabilté* :

```
result$pois
```

```
##
## Observed and fitted values for poisson distribution
## with parameter 0.464347826086957 estimated by `ML`
##
##  count observed    fitted       prob
##      0     1355 1445.6533 0.62854489
##      1      822  671.2859 0.29186345
##      2      123  155.8551 0.07959165
##
## X-squared = 59.22774, df = 1, p.value = 1.398881e-14
## We reject the null hypothesis
```

En utilisant le test de $\tilde{\chi}^2{}_1$ on retrouvera la

$$pvalue = 1.3988810110277e - 14$$

et

$$\tilde{\chi}^2 = 59.2277422610588$$

Le 95^e centile d'une loi χ_1^2

$$\chi_1^2(95\%) = 3.84145882069412$$

on a donc

$$\tilde{\chi}^2 > \chi_1^2(95\%)$$

et

$$pvalue < 5\%$$

On rejete l'hypthèse Nulle $\mathcal{H}_0$. Donc la distribution de loi **de poisson** ne permet pas de modèlisé celle de **X**.

Ajustement par une distribution de loi binomiale

Après avoir utilisé `vcd::goodfit(x, "binom", size=2)` pour obtenir la distribution binomiale ajustée dans le corps de notre fonction `getgoodfit`. On retrouvera l'*estimation* suivantes:

```
result$binom$estimate
```

```
## $prob
## [1] 0.2321739
##
## $size
## [1] 2
```

On affiche le rootograme suspendu :

```
result$binom$plot
```

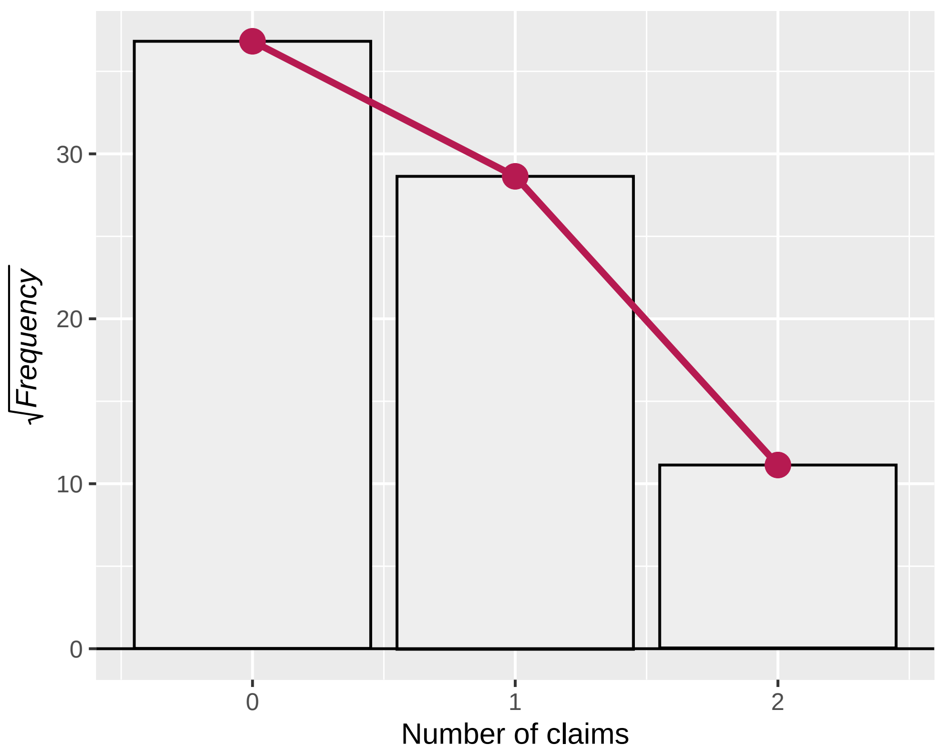

FIGURE 3 – Ajustement par une distrubition $\sim \mathcal{B}\,(0.23, 2)$

On remarque que la distribution binomiale est la mieux ajustée à la distribution de **X**.

On affiche les valeurs de **X**, les *effectifs théoriques*, les *effectifs observés* et le *vecteur de probabilté* :

```
result$binom
```

```
##
## Observed and fitted values for binomial distribution
```

```
## with parameters (0.232173913043478, 2) estimated by `ML`
##
##  count observed    fitted       prob
##      0     1355 1355.9809 0.58955690
##      1      822  820.0383 0.35653837
##      2      123  123.9809 0.05390473
##
## X-squared = 0.01316261, df = 1, p.value = 0.9086604
## We accept the null hypothesis
```

En utilisant le test de $\tilde{\chi}^2{}_1$ on retrouvera la

$$pvalue = 0.908660377467628$$

et

$$\tilde{\chi}^2 = 0.0131626130456561$$

Le 95ᵉ centile d'une loi χ^2_1

$$\chi^2_1(95\%) = 3.84145882069412$$

on a donc

$$\tilde{\chi}^2 < \chi^2_1(95\%)$$

et

$$pvalue > 5\%$$

On accepte l'hypothèse Nulle $\mathcal{H}_0$. Donc la distribution de loi **binomiale** permet de bien modéliser la distribution la valeur aléatoire **X**.

Ajustement par une distribution de loi binomiale négative

Après avoir utilisé `vcd::goodfit(x, "nbinom", size=2)` pour obtenir la distribution binomiale négative ajustée dans le corps de notre fonction `getgoodfit`. On retrouvera l'*estimation* suivantes:

```
result$nbinom$estimate
```

```
## $size
## [1] 2
##
## $prob
## [1] 0.8115737
```

On affiche le rootograme suspendu :

```
result$nbinom$plot
```

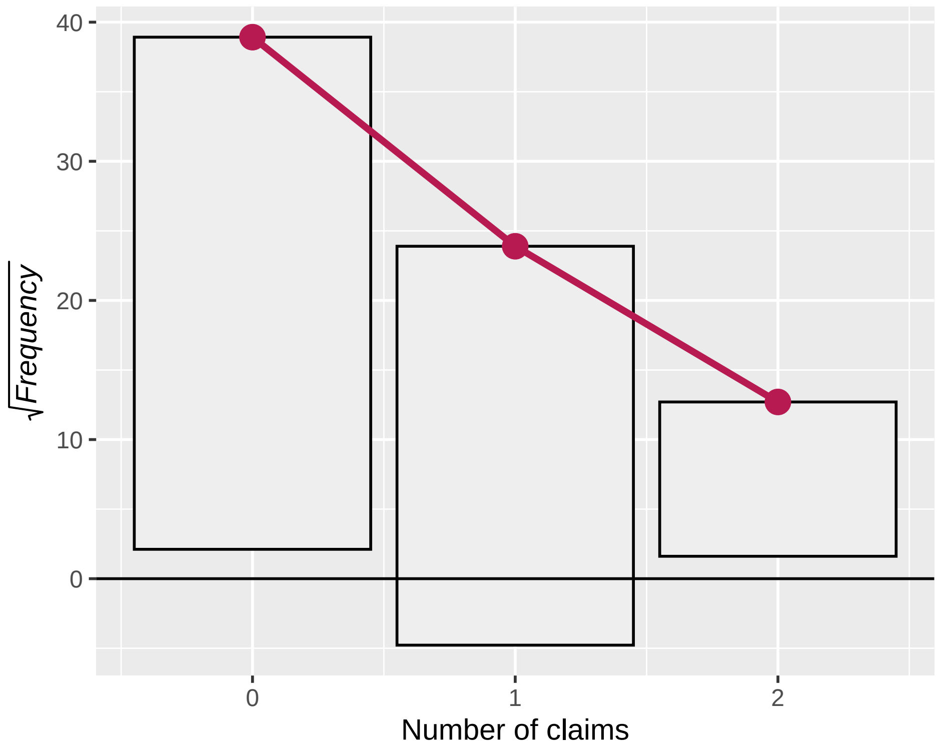

FIGURE 4 – Ajustement par une distrubition $\sim \mathcal{NB}\,(2, 0.81)$

On affiche les valeurs de $\mathbf{X}$, les *effectifs théoriques*, les *effectifs observés* et le *vecteur de probabilté* :

```
result$nbinom
```

```
##
## Observed and fitted values for negative binomial distribution
## with parameters (2, 0.811573747353564) estimated by `ML with size fixed`
##
##  count observed    fitted       prob
##      0     1355 1514.8995 0.65865195
##      1      822  570.8937 0.24821464
##      2      123  161.3570 0.09313342
##
## X-squared = 166.161, df = 1, p.value = 0
## We reject the null hypothesis
```

En utilisant le test de $\tilde{\chi}^2{}_1$ on retrouvera la

$$pvalue = 0$$

et

$$\tilde{\chi}^2 = 166.1610079339$$

Le 95$^{\text{e}}$ centile d'une loi χ_1^2

$$\chi_1^2(95\%) = 3.84145882069412$$

on a donc

$$\tilde{\chi}^2 > \chi_1^2(95\%)$$

et

$$pvalue < 5\%$$

On rejete l'hypthèse Nulle $\mathcal{H}_0$. Donc la distribution de loi **binomiale négative** ne permet pas de modèlisé celle de $\mathbf{X}$.

Conclusion : La distribution la mieux ajustée est celle de la loi binomiale

On écrit :

$$\mathbf{X} \sim \mathcal{B}(0.23, 2)$$