

ggplot2

Jingwen SU

12/9/2020

Introduction

Dans ce fichier, nous utilisons principalement ggplot2 pour dessiner un histogramme et une courbe de densité sur les données. Parmi eux, nous importerons les packages d'installation gcookbook et MASS afin d'utiliser l'ensemble de données.

Matériel de référence: R Graphics Cookbook (<http://www.cookbook-r.com/Graphs/>)

Importer ggplot2

```
## install.package("ggplot2")  
library(ggplot2)
```

Faire une Histogramme

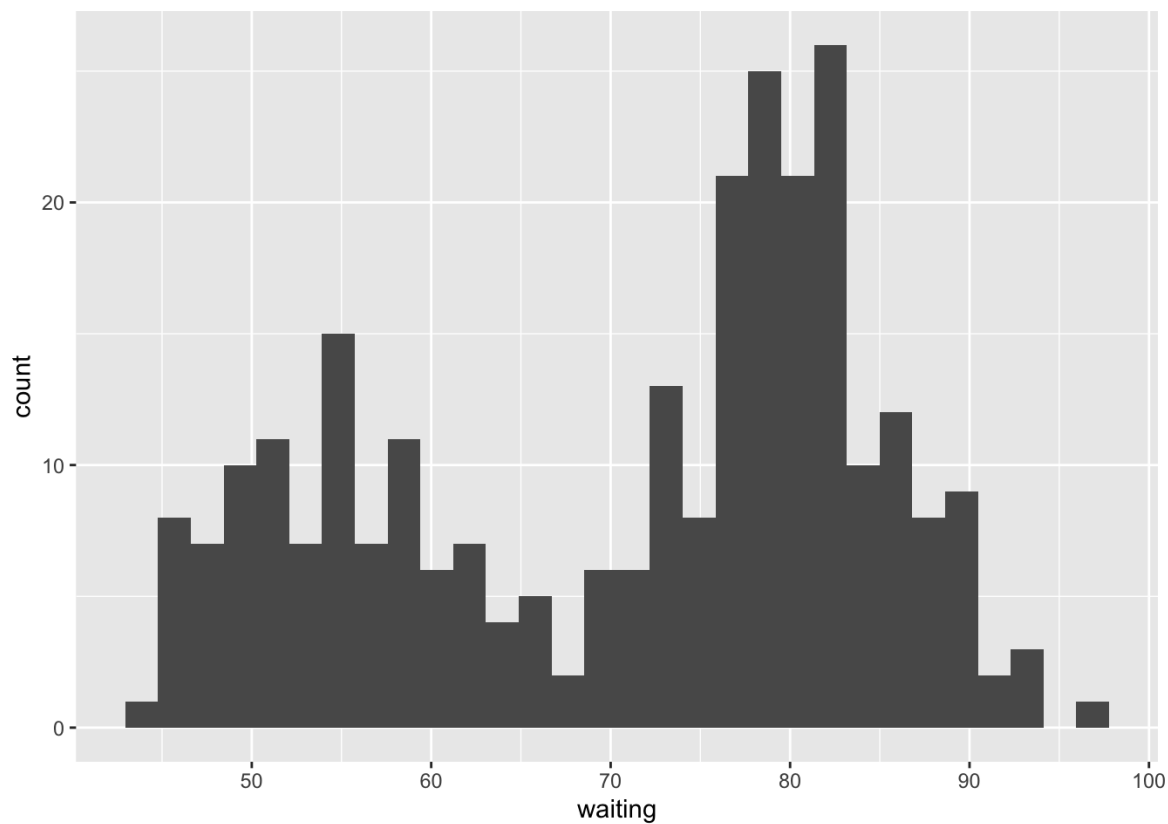
une base Histogramme

```
# Importez le package d'installation où se trouve l'ensemble de données et lisez les données  
library(gcookbook)  
head(faithful)
```

```
##   eruptions waiting  
## 1      3.600      79  
## 2      1.800      54  
## 3      3.333      74  
## 4      2.283      62  
## 5      4.533      85  
## 6      2.883      55
```

```
# Pour dessiner un histogramme, la largeur par défaut de la colonne est la valeur maximale moins la  
# valeur minimale divisée par 30, et la largeur peut être définie par binwidth = x  
ggplot(faithful, aes(x=waiting)) + geom_histogram()
```

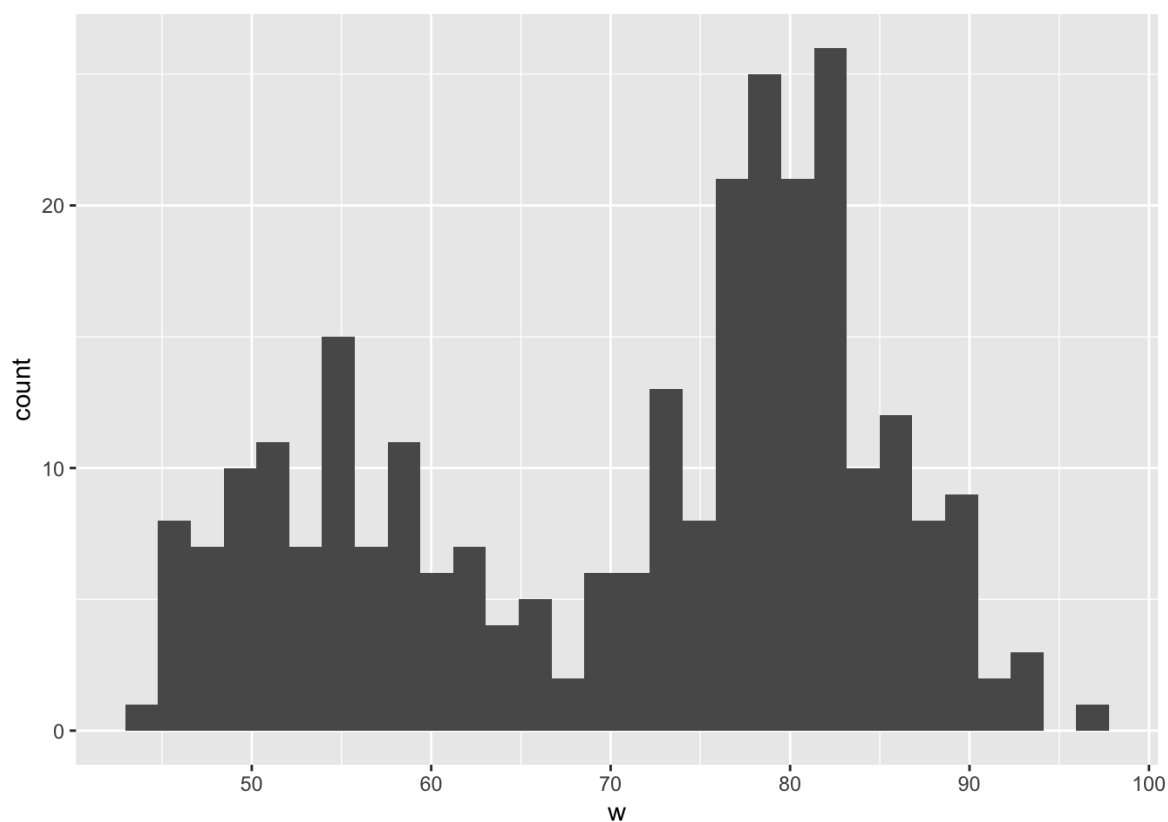
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Si vous n'écrivez pas les premières données, il est également possible d'utiliser directement les données comme valeur x.

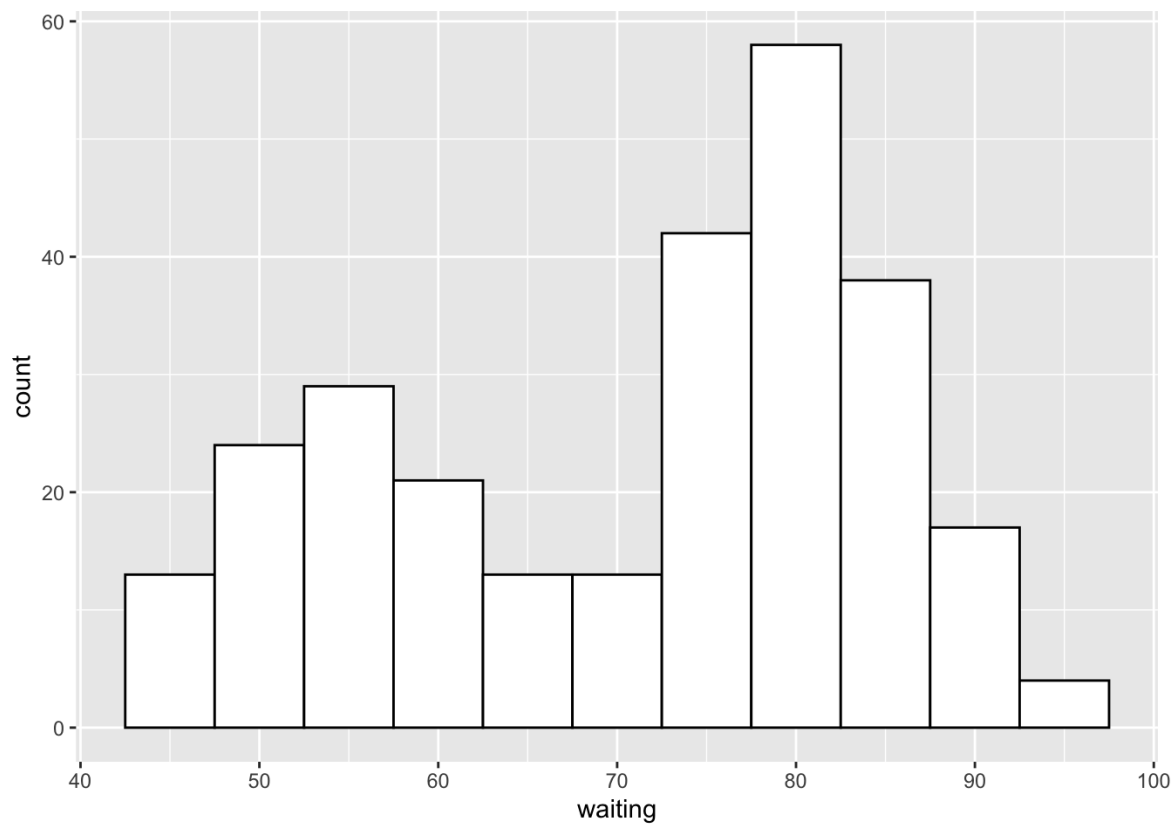
```
w = faithful$waiting
ggplot(NULL, aes(x=w)) + geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Définir la propriété columnarity

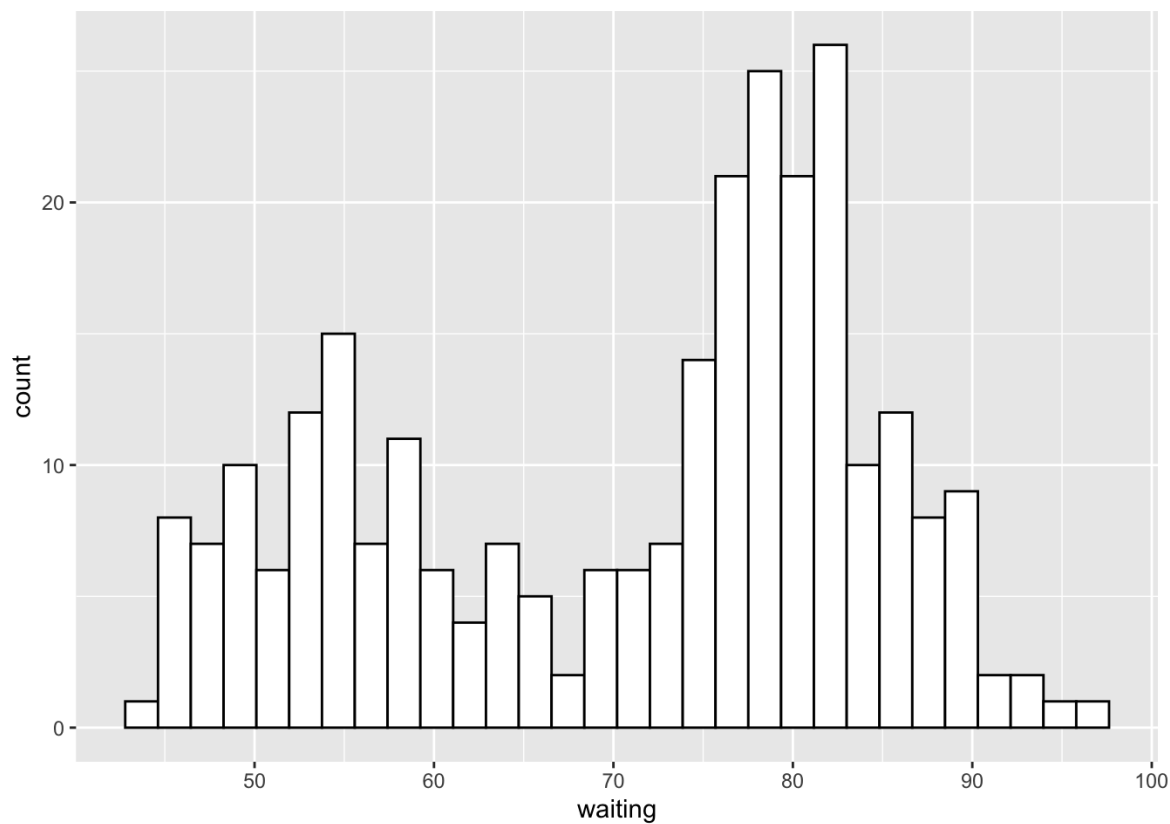
```
# Définir binwidth, fill, colour
ggplot(faithful, aes(x=waiting)) +
  geom_histogram(binwidth=5, fill="white", colour="black")
```



```
# Définir le point de départ de l'abscisse "origin"
ggplot(faithful, aes(x=waiting)) +
  geom_histogram(fill="white", colour="black", origin=30)
```

```
## Warning: `origin` is deprecated. Please use `boundary` instead.
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



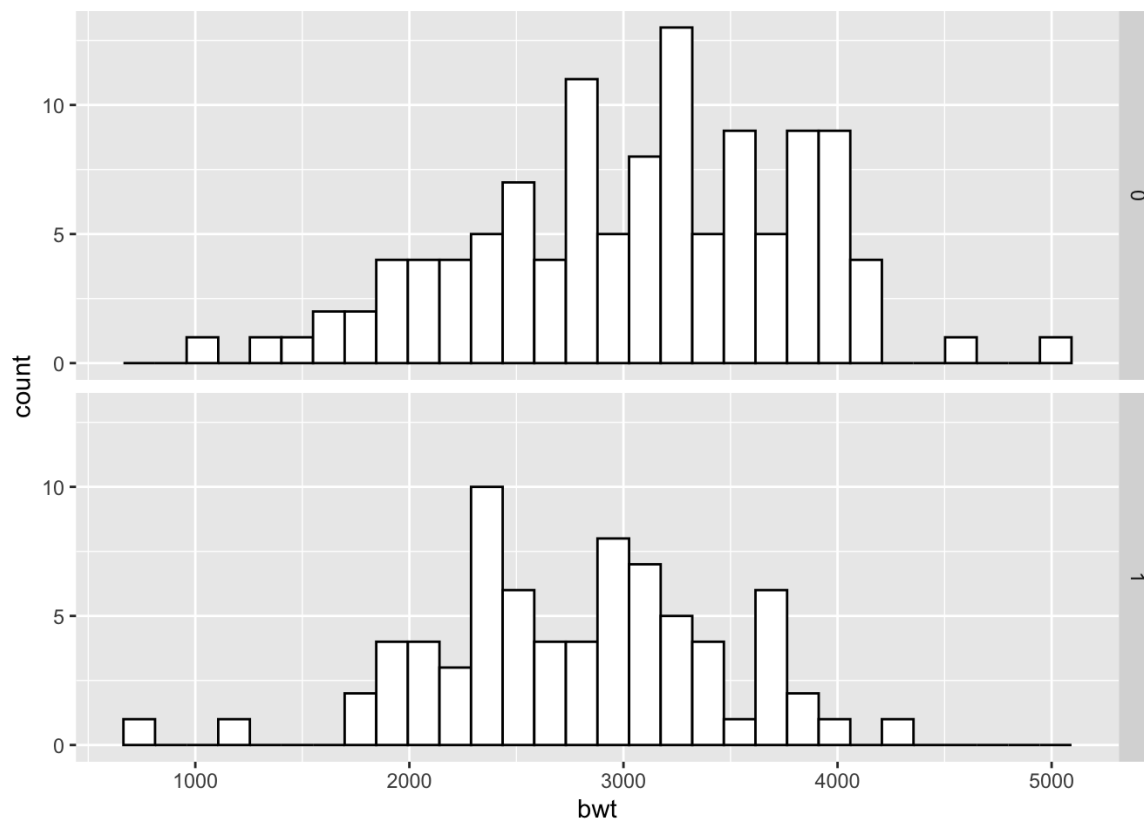
#Le point de départ de l'axe des x est 30, et l'image se décale vers la droite et devient dense

Création de plusieurs histogrammes à partir de données groupées

```
library(MASS)
```

```
# Dessinez un histogramme et séparez tout avec de la fumée
ggplot(birthwt, aes(x=bwt)) + geom_histogram(fill="white", colour="black") +
  facet_grid(smoke ~ .)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Pour une compréhension intuitive, réattribuez smoke et importez un package d'installation plyr qui réassigne les données.

```
birthwt1 = birthwt
birthwt1$smoke = factor(birthwt1$smoke)
levels(birthwt1$smoke)
```

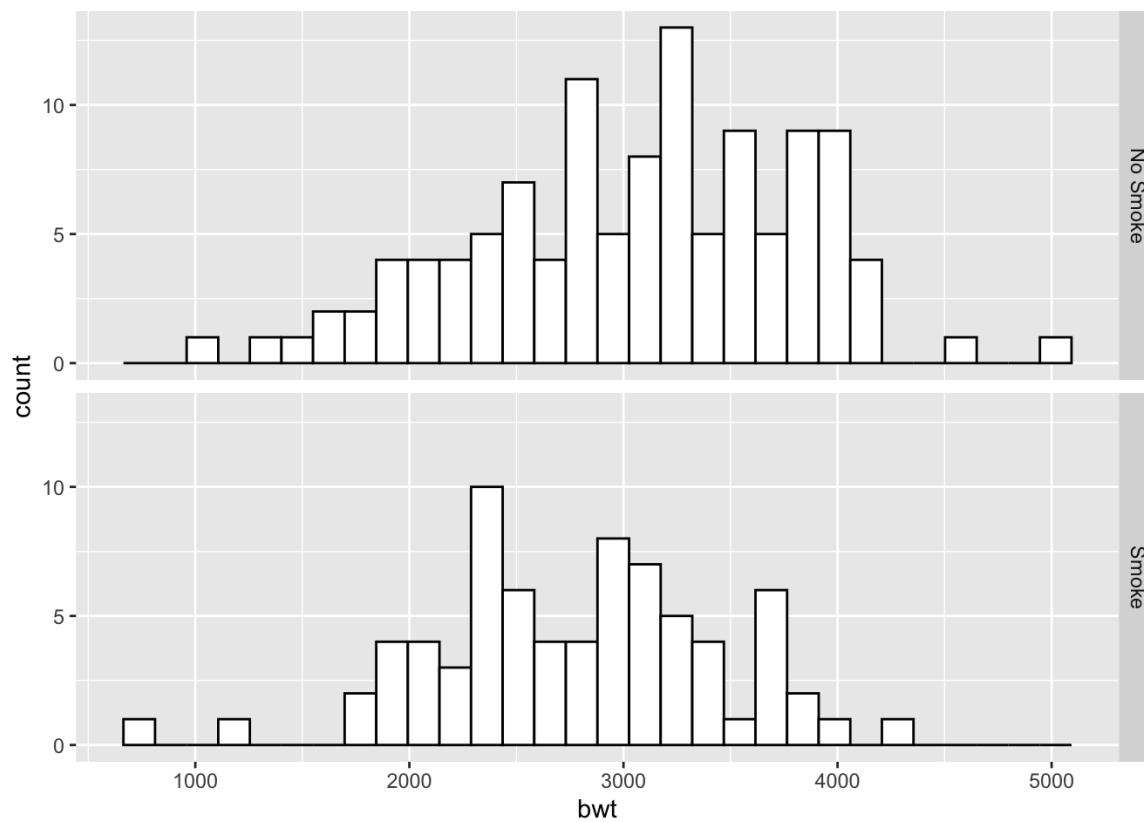
```
## [1] "0" "1"
```

```
require(plyr)
```

```
## Loading required package: plyr
```

```
library(plyr)
birthwt1$smoke <- revalue(birthwt1$smoke, c("0"="No Smoke", "1"="Smoke"))
ggplot(birthwt1, aes(x=bwt)) + geom_histogram(fill="white", colour="black") +
  facet_grid(smoke ~ .)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

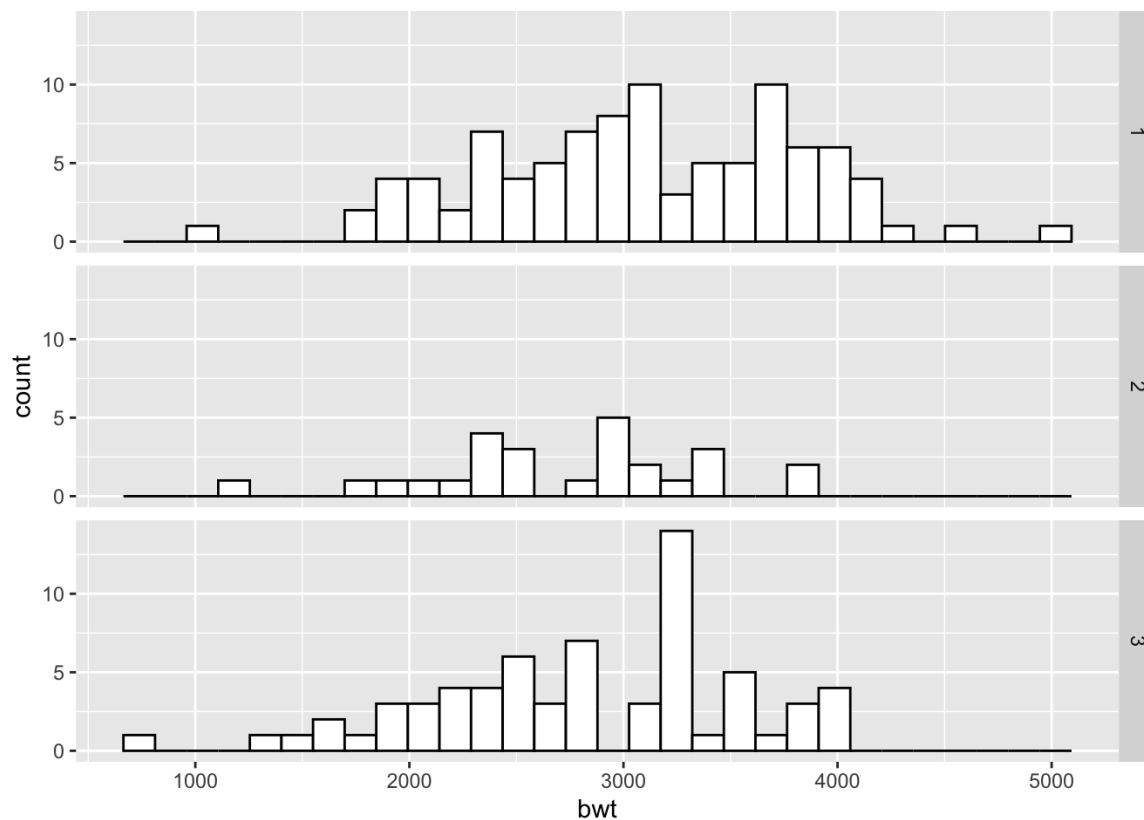


Réinitialiser la valeur de l'axe y

Regroupées par la race de la mère

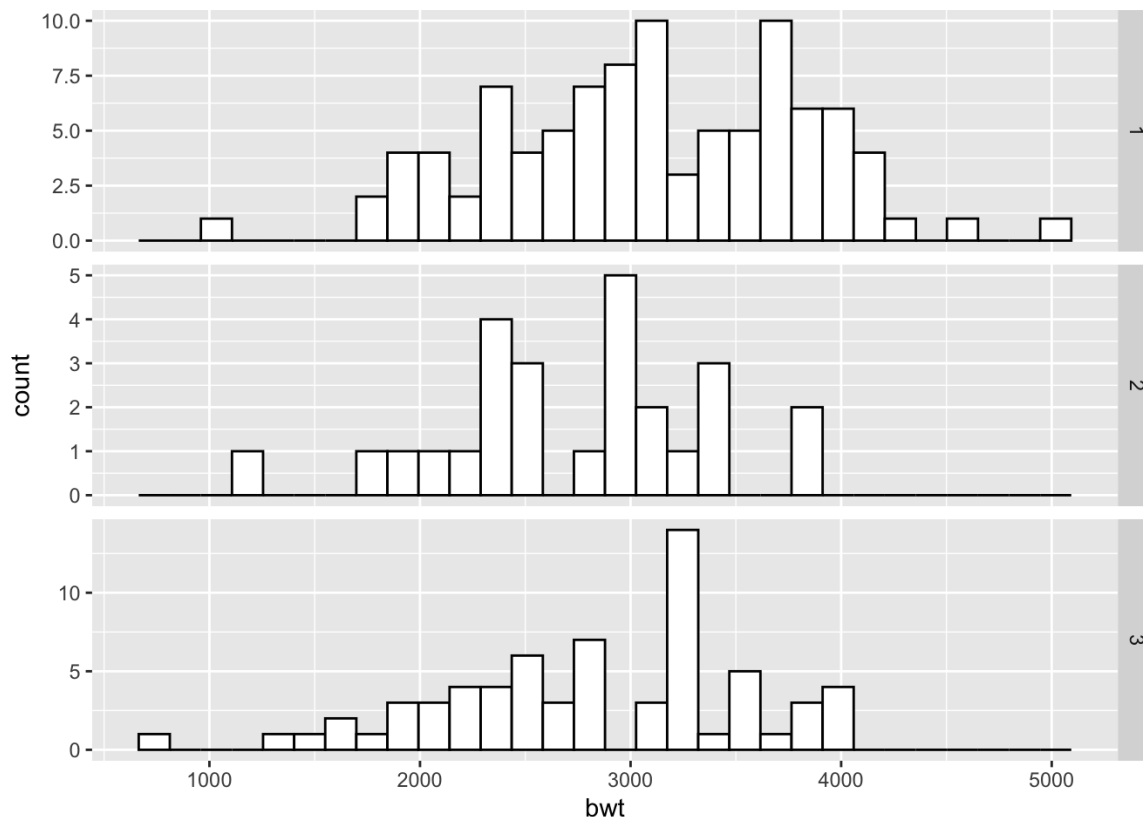
```
ggplot(birthwt, aes(x=bwt)) + geom_histogram(fill="white", colour="black") +  
  facet_grid(race ~ .)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
#Réinitialiser la valeur de l'axe y: scales="free"
ggplot(birthwt, aes(x=bwt)) + geom_histogram(fill="white", colour="black") +
  facet_grid(race ~ ., scales="free")
```

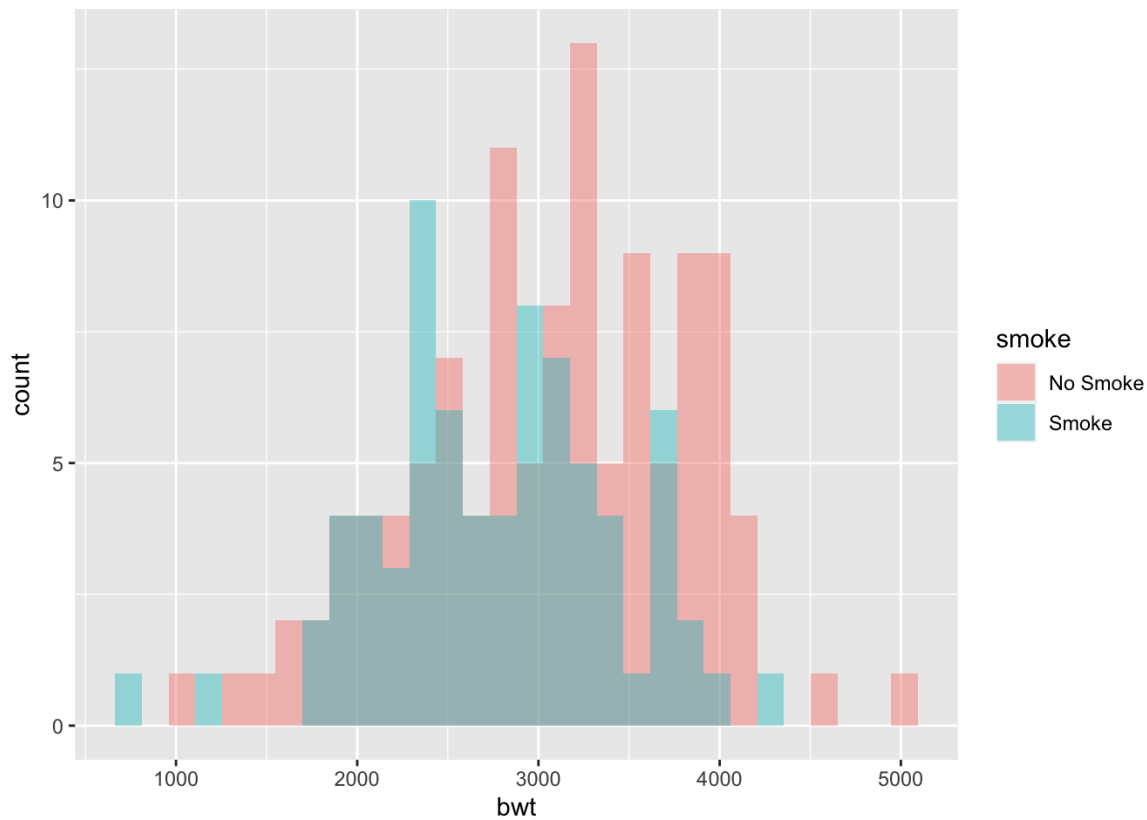
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Regroupez les données dans le même tableau fill=smoke; position n'est pas ajusté, faites la comparaison d'origine; alpha est le paramètre de transparence.

```
ggplot(birthwt1, aes(x=bwt, fill=smoke)) +
  geom_histogram(position="identity", alpha=0.4)
```

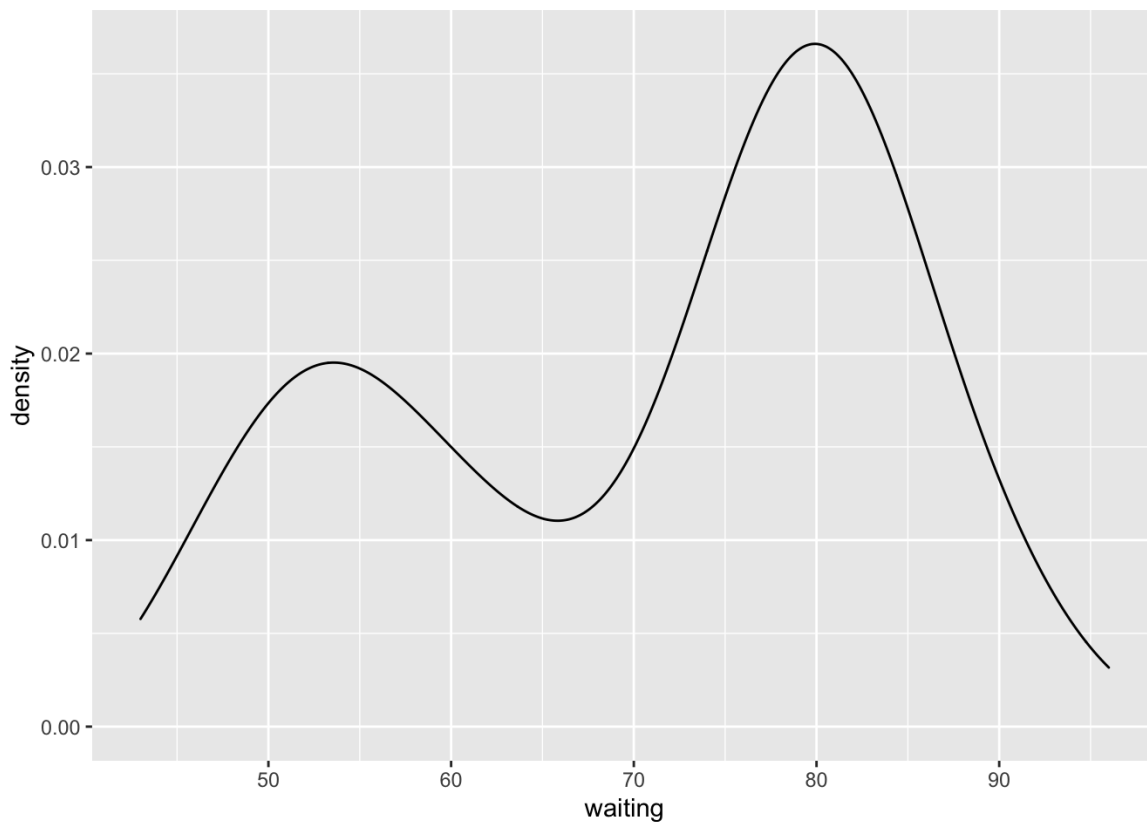
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Créer une courbe de densité

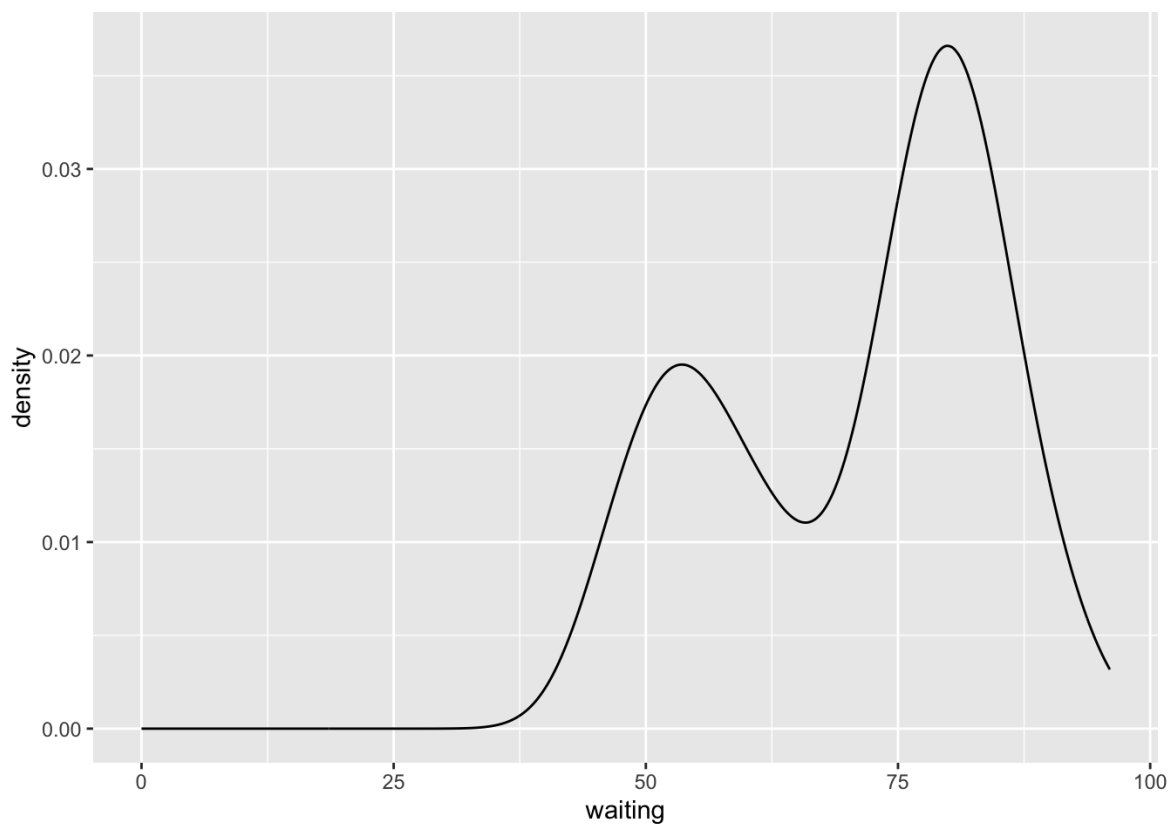
base

```
ggplot(faithful, aes(x=waiting)) + geom_density()
```



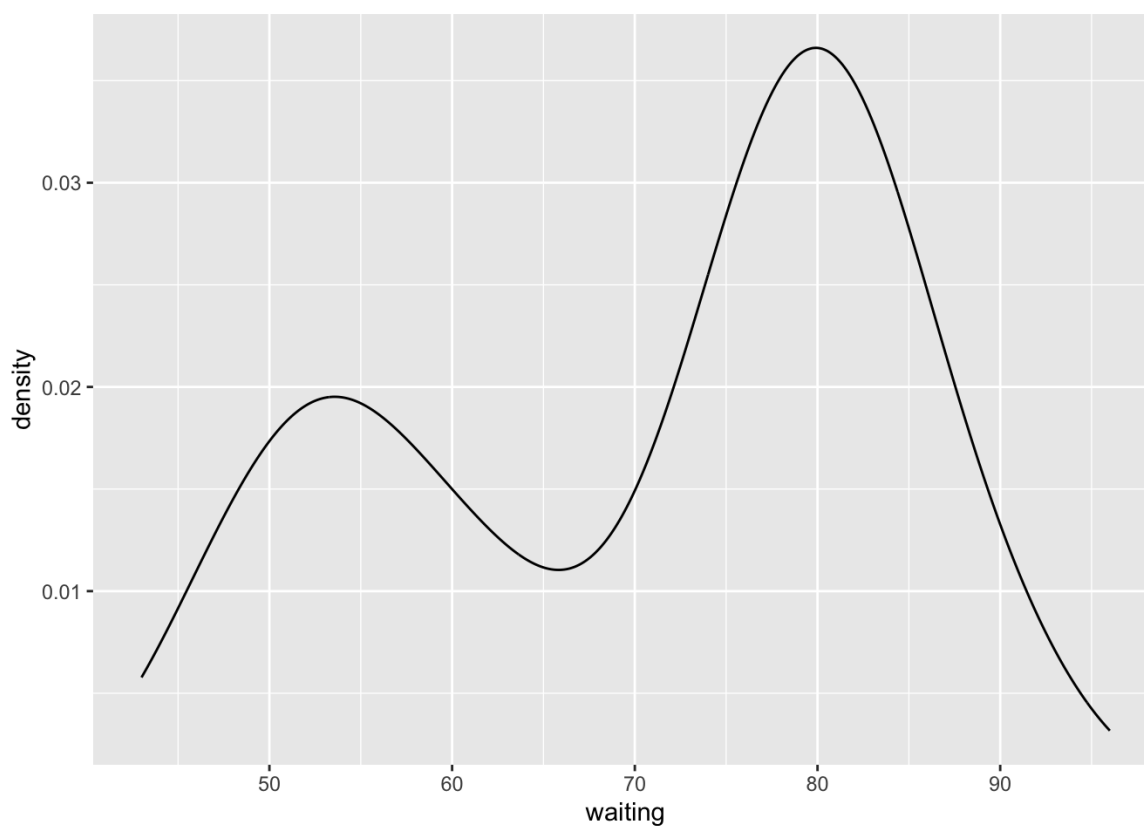
expand_limits: Dériver la valeur de x à 0


```
ggplot(faithful, aes(x=waiting)) + geom_line(stat="density") +  
  expand_limits(x=0)
```



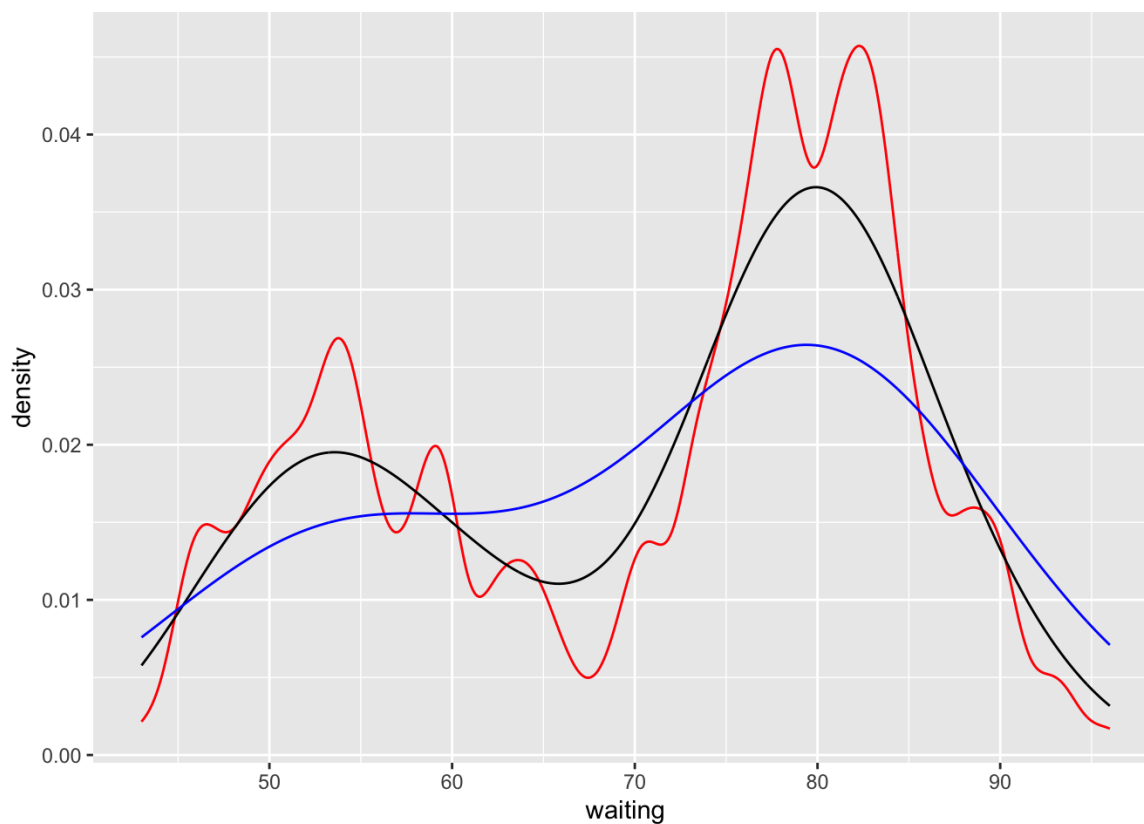
geom_line est la ligne d'aire définie

```
ggplot(faithful, aes(x=waiting)) + geom_line(stat="density")
```



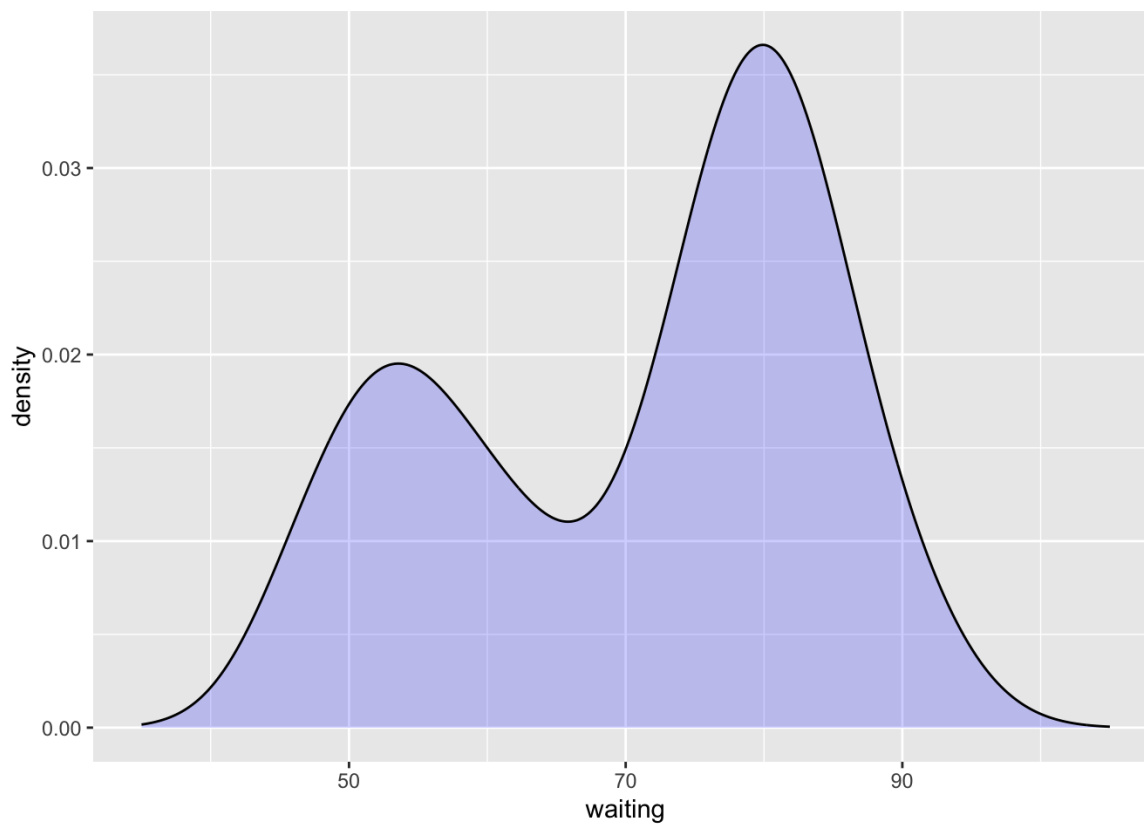
ajuster définit la régularité de la courbe de densité

```
ggplot(faithful, aes(x=waiting)) +  
  geom_line(stat="density", adjust=0.25, colour="red") +  
  geom_line(stat="density") +  
  geom_line(stat="density", adjust=2, colour="blue")
```



Définissez la plage de l'axe des x, à partir de 35 et se terminant à 105, couleur = NA signifie que la couleur n'a pas de contour

```
ggplot(faithful, aes(x=waiting)) +  
  geom_density(fill="blue", colour=NA, alpha=.2) +  
  geom_line(stat="density") +  
  xlim(35, 105)
```

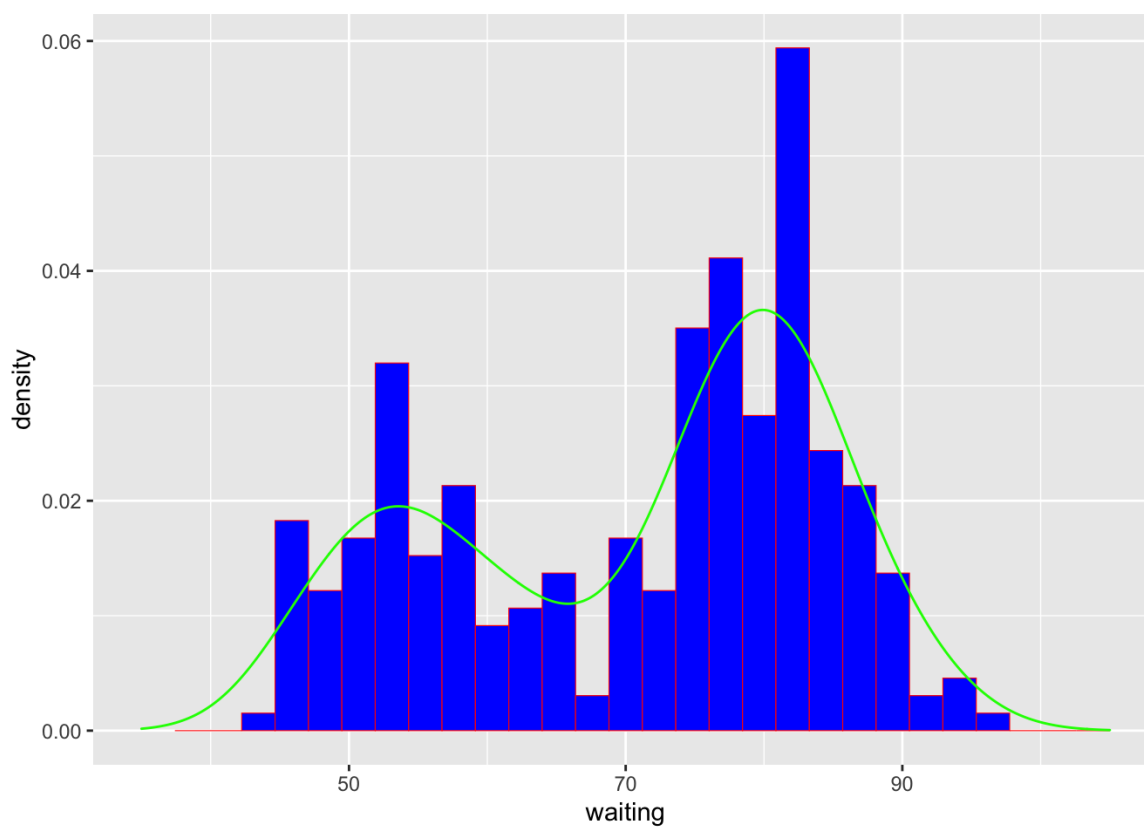


Combinez l'histogramme et la courbe de densité

```
ggplot(faithful, aes(x=waiting, y=..density..)) +  
  geom_histogram(fill="blue", colour="red", size=.2) +  
  geom_density(color="green") +  
  xlim(35, 105)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 1 rows containing missing values (geom_bar).
```



Création de courbes de densité multiples à partir de données groupées.

```
ggplot(birthwt1, aes(x=bwt, fill=smoke)) + geom_density(alpha=.3)
```

