

TP1 SY02 : Prise en main de R

3 Structures de données usuelles

On utilise les vecteurs pour des variables quantitatives.

2. Initialiser le vecteur colonne notes.

```
notes <- c(18, 1.5, 9.5, 15.5, 15, 15.5, 0.5, 14.5, 10)
```

3. Ajoute la valeur 4.

```
notes <- c(notes, 4)
```

4. La variable notes10 devient donc le vecteur notes avec toutes les valeurs divisées par 2.

```
> notes10 <- notes / 2  
[1] 9.00 0.75 4.75 7.75 7.50 7.75 0.25 7.25 5.00 2.00
```

On renvoie toutes les notes supérieurs à 10.

```
> notes10 > 6  
TRUE FALSE FALSE TRUE TRUE TRUE FALSE TRUE FALSE FALSE
```

Il y a donc 5 étudiants qui ont eu plus de 6/10.

5. Moyenne de 3 valeurs (Il faut les mettre dans un $c(..., ..., ...)$ car moyenne prend un seul argument qui est un vecteur)

```
(v[1] + v[5] + v[3]) / 3  
mean(c(v[1], v[5], v[3]))
```

6. Filtrage on cherche combien de notes sont supérieurs à 10.

```
choix <- notes > 10  
length(notes[choix])
```

7.

```
min(notes[(notes - floor(notes)) == 0])
```

notes - floor(notes) renvoie la partie décimal du nombre. Si elle est == à 0, le nombre est non fractionnaire. Il reste juste à appliquer un filtrage comme avec choix.

8. Met dans notes2 le vecteur notes avec toutes les valeurs diminuées par 2

```
notes2 <- notes - 2
```

9.

```
# Renvoie le nombre de valeurs inférieurs à 0  
length(notes2 [notes2 < 0])
```

```
# Met à 0 toutes les notes inférieurs à 0
```

```
notes2[notes2 < 0] <- 0
```

On utilise des facteurs pour les variables qualitatives.

```
> collection <- c("R", "V", "B", "V")
```

```
> collection
```

```
[1] "R" "V" "B" "V"
```

```
> f <- factor( c("R", "V", "B", "V"))
```

```
> f
```

```
[1] R V B V
```

```
Levels: B R V # Correspond aux modalités (valeurs prises dans l'échantillon)
```

```
(f <- ordered(collection))
```

```
[1] R V B V
```

```
Levels: B < R < V
```

```
f > "B"
```

```
[1] TRUE TRUE FALSE TRUE
```

```
f <- factor(collection, ordered = TRUE)
```

```
f < "R"
```

```
[1] FALSE FALSE TRUE FALSE
```

```
(f <- factor(collection, ordered = TRUE, levels = c("R", "V", "B")))
```

```
[1] R V B V
```

```
Levels: R < V < B
```

```
f < "R"
```

```
[1] FALSE FALSE FALSE FALSE
```

10.

```
ADN <- factor(c("A","C","A","A","G","A","T","G","C","C","A","T","T","G","T","C"))
```

```
ADN
```

```
# [1] A C A A G A T G C C A T T G T C
```

```
# Levels: A C G T
```

```
nlevels(ADN)
```

```
# [1] 4
```

```
levels(ADN)
```

```
# [1] "A" "C" "G" "T"
```

11.

```
# Nombre de A dans le brin d'ADN
```

```
length(ADN[ADN == "A"])
```

```
# [1] 5
```

```
length(ADN[ADN == "T"])
```

```
# [1] 4
```

```
length(ADN[ADN == "C"])
```

```
# [1] 4
```

```
length(ADN[ADN == "G"])
```

```
# [1] 3
```

12.

```
length(X) # Nb de colonnes de X
ncol(X) # Nb de colonnes de X
nrow(X) # Nb de lignes de X
names(X) # Nom des colonnes de X
# [1] "correcteur.median" "median" "correcteur.final" "final" "moyenne" "resultat"
```

TODO : Quel est la différence entre length et ncol

13. Il y a 3 variables qualitatives et 3 quantitatives sur chaque ligne.

14.

```
X[1,1] # Extrait le 1er élément
X[,3] # Extrait la 3e colonne
X[1:10,] # Extrait les 10 premières lignes
X[c(1,3),c(1,4)] # Extraire les lignes 1 et 3 et les colonnes 1 et 4

X[,c(2,6)] # Extrait la 2e et la dernière colonne du tableau X
```

15.

```
mean(X[X$correcteur.median == "EG",2]) # Moyenne des étudiants du médian ayant été corrigé
# [1] 12.63208
```

16.

```
nrow(X[X$median < X$final,]) / nrow(X) # Proportion d'étudiants qui ont eu une meilleure note
```

17.

```
mean(X$final) # Moyenne
sd(X$final) # écart-type
var(X$final) # Variance
median(X$final) # Médiane
max(X$final)
min(X$final)
```

18.

```
> head(X)
```

	correcteur.median	median	correcteur.final	final	moyenne	resultat
1	BR	11.0	ALC	17.5	14.9	C
2	EN	14.0	BR	16.0	15.2	C
3	ALC	10.5	ALC	13.0	12.0	D
4	BR	17.0	BR	13.0	14.6	C
5	EG	14.5	EN	14.0	14.2	C
6	EG	12.0	EN	19.5	16.5	B

```
summary(X$final)
```

correcteur.medianmedian		correcteur.final		final	moyenne	resultat
Length:297	Min. :	Length:297	Min. :	Min. :	Min. :	Length:297
	0.50		0.00		0.20	
Class	1st	Class	1st		1st	Class
:character	Qu.:10.50	:character	Qu.:12.50	Qu.:12.30		:character
Mode	Median	Mode	Median	Median		Mode
:character	:13.50	:character	:16.00	:14.90		:character
	Mean		Mean	Mean		
	:12.96		:14.76	:14.04		
	3rd		3rd	3rd		
	Qu.:16.50		Qu.:18.00	Qu.:17.00		
	Max.		Max.	Max.		
	:20.00		:20.00	:20.00		

19. Quartiles

```
IQR(X$median) # Etendue inter-quartile (Q3-Q1)
quantile(X$median)
```

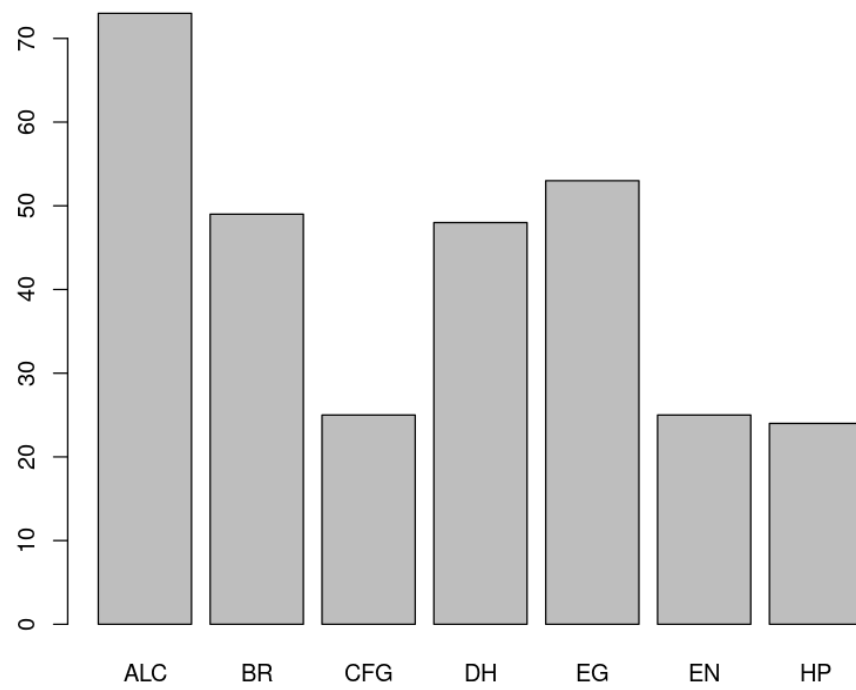
0%	25%	50%	75%	100%
0.5	10.5	13.5	16.5	20.0

20. Moyenne tronquée d'ordre 10

```
mean(sort(X$median)[11:287]) # Moyenne tronquée d'ordre 10 (moyenne en enlevant les 10 prem
```

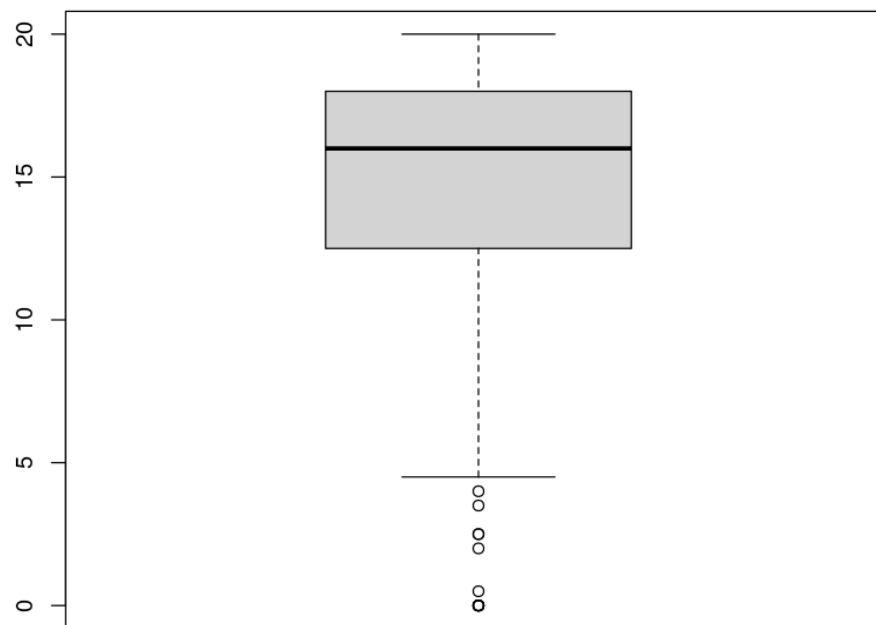
21. Diagramme en bande

```
barplot(table(X$correcteur.median)) # Diagramme en bande
# Nb de copies corrigé par correcteur.
# On remarque que ALC a corrigé le plus de copies
```



22. Boite à moustache des notes de final

```
boxplot(X$final)
```



23. TODO : ?

24. Diagramme en tige et feuille de la moyenne

`stem(X$moyenne)`

```

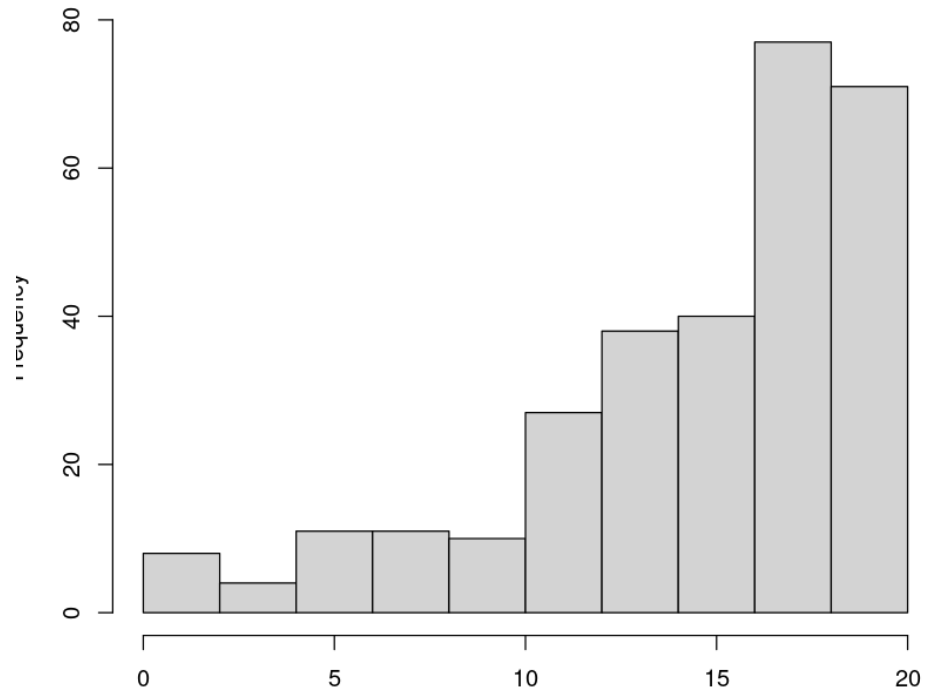
0 | 224
1 | 247
2 | 3
3 | 22
4 | 79
5 | 0124469
6 | 01134579
7 | 02457
8 | 22355888
9 | 2224
10 | 1255
11 | 000111223345899
12 | 00000011122333355556666777788999
13 | 00022234455666667777
14 | 000001112233333444455666666778889999
15 | 00000122233334444556666666677889999
16 | 00011122333333455556667777788999
17 | 0000111223333444444456667888999

```

```
18 | 00000011122223444566788899999
19 | 000112233333444678
20 | 00
```

25. Histogramme des notes du final

```
hist(X$final)
```



26. Coupe l'histogramme à 15

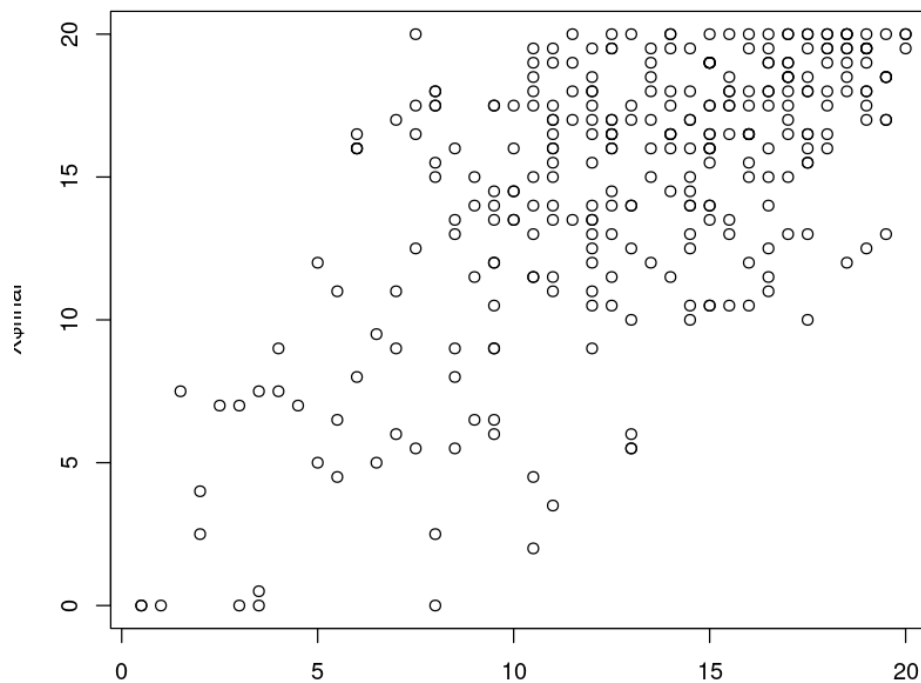
```
hist(X$final, breaks = c(0,15,20))
```

27. TODO

28. TODO

29.

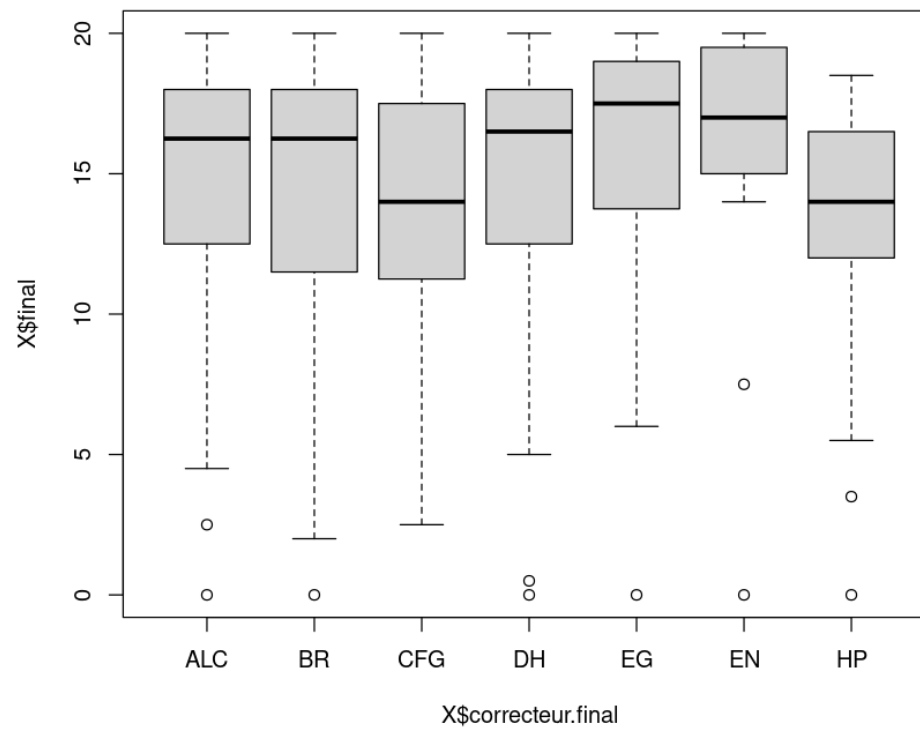
```
plot(X$median , X$final)
```



On remarque que les valeurs sont dans la diagonale. Les notes du médian et du final sont donc un peu près égales pour tout les étudiants : Un étudiant qui réussit le final réussit aussi le médian et inversement.

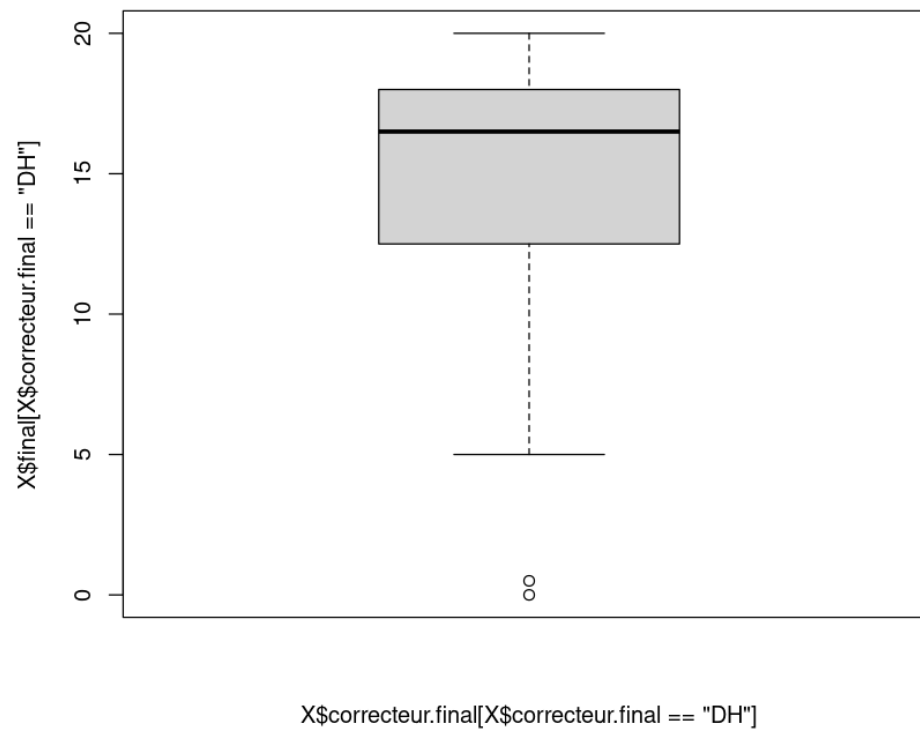
30. Boite à moustache en fonction des correcteurs

```
boxplot(X$final ~ X$correcteur.final) # Boite à moustache en fonction des correcteurs
```

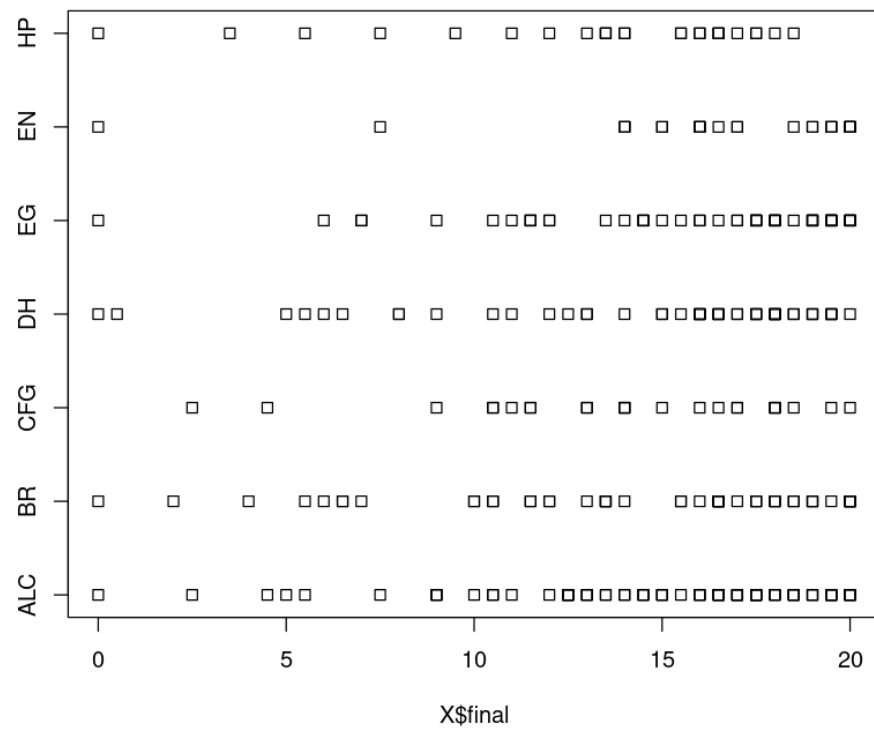
31. Boite à moustache du correcteur DH

```
boxplot(X$final[X$correcteur.final == "DH"] ~ X$correcteur.final[X$correcteur.final == "DH"])
```



32. Graph avec les carrés

```
stripchart(X$final ~ X$correcteur.final, data = X)
```



33. *method = "jitter"* permet de décaler les carrés pour mieux voir le nombre de points présents

```
stripchart(X$final ~ X$correcteur.final, data = X, method = "jitter")
```

