

Mise à niveau R 3/3

Eric Marcon

25 January 2024

Tidyverse



Univers bien rangé.

Extension de R : ensemble de packages

```
library("tidyverse")
```

Manifeste

```
vignette("manifesto")
```



Introduction à R et au tidyverse, [Le tidyverse](#)

Autant que possible dans un dataframe.

tibble : dataframe amélioré.

```
(mon_tibble <- tibble(nom = c("X", "Y"), taille = c(100, 120)))
```

```
## # A tibble: 2 x 2
##   nom      taille
##   <chr>   <dbl>
## 1 X             100
## 2 Y             120
```

Le résultat d'une fonction est le premier argument de la fonction suivante.

```
x <- runif(100, max = 10) %>% mean()  
x
```

```
## [1] 4.575803
```

```
# ou même
```

```
100 %>% runif(max = 10) %>% mean() %>% print() -> x
```

```
## [1] 5.079611
```

Data wrangling : lecture des données dans un tibble, sélection des lignes et colonnes, création de colonnes...

```
# Lecture des arbres de la parcelle 6 de Paracou
read_csv2("data/Paracou6.csv") %>%
  # Ne garder que les fabaceae
  filter(Family == "Fabaceae") %>%
  # Sélectionner les colonnes espèce et circonférence
  select(spName, CircCorr) %>%
  # Calculer la surface terrière de chaque arbre en m2
  mutate(G = CircCorr^2 / 4 / pi / 10000) %>%
  # Grouper par espèce
  group_by(spName) %>%
  # Calculer le nombre de tiges et la surface terrière par ha
  summarize(Abondance = n(), Surface = sum(G) / 6.25) %>%
  # Trier par G/ha décroissant
  arrange(desc(Surface)) ->
mon_tibble
```

Travail fastidieux :

- prévoir du temps
- capitaliser.

```
mon_tibble
```

```
## # A tibble: 46 x 3
##   spName                Abundance Surface
##   <chr>                  <int>   <dbl>
## 1 Eperua_falcata         266     5.68
## 2 Eperua_grandiflora      67     1.50
## 3 Vouacapoua_americana   91     1.44
## 4 Dicorynia_guianensis   44     0.762
## 5 Recordoxylon_speciosum 31     0.438
## # i 41 more rows
```

Graphiques avec ggplot

Package destiné à la création de graphiques.

Respecte la **grammaire graphique par couches** :

```
ggplot(data = <DATA>) +  
  <GEOM_FUNCTION>(  
    mapping = aes(<MAPPINGS>),  
    stat = <STAT>,  
    position = <POSITION>  
  ) +  
  <COORDINATE_FUNCTION> +  
  <FACET_FUNCTION>
```

Les données sont obligatoirement un dataframe (un tibble est un dataframe).

L'esthétique désigne ce qui est représenté :

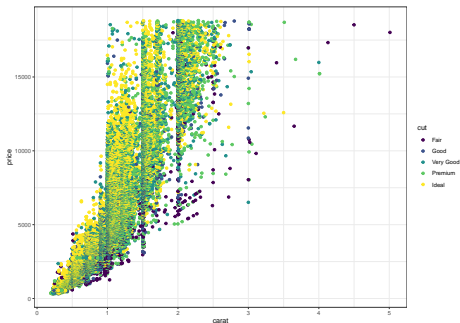
- x et y (ou fill pour un histogramme...)
- transparence, couleur, type de courbe, taille... : voir l'aide de chaque `geom_`.

Fonction `aes()` à plusieurs niveaux :

- argument `mapping` de `ggplot()`, hérité par les couches (`geom_`)
- ou argument `mapping` de chaque couche.

La géométrie est définie par une fonction `geom_xxx` et une esthétique (ce qui est représenté).

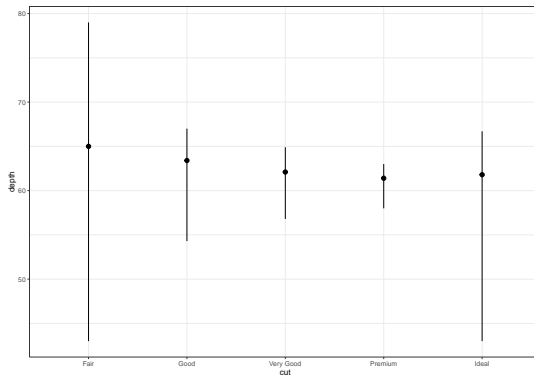
```
ggplot(data = diamonds) +  
  geom_point(mapping = aes(x = carat, y = price, color = cut))
```



Chaque `geom_` va de pair avec une statistique de transformation des données :

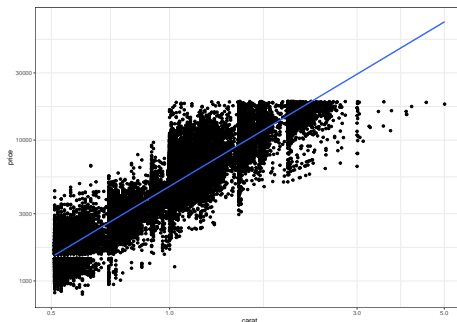
- “identity” pour `geom_point`
- “boxplot” pour `geom_boxplot`
- 20 statistiques disponibles...

```
ggplot(data = diamonds) +  
  stat_summary(  
    mapping = aes(x = cut, y = depth),  
    fun.min = min,  
    fun.max = max,  
    fun = median  
  )
```



Transformation de variable.

```
diamonds %>% filter(carat > .5) %>%  
  ggplot(aes(x=carat, y=price)) + geom_point() +  
  scale_x_log10() + scale_y_log10() + geom_smooth(method="lm")
```



La position définit l'emplacement des objets sur le graphique.

- “identity” en général,
- “stack” empile les catégories dans un histogramme,
- “jitter” déplace aléatoirement les points dans un `geom_point` pour éviter les superpositions.

Position (2)

Mise à niveau
R 3/3

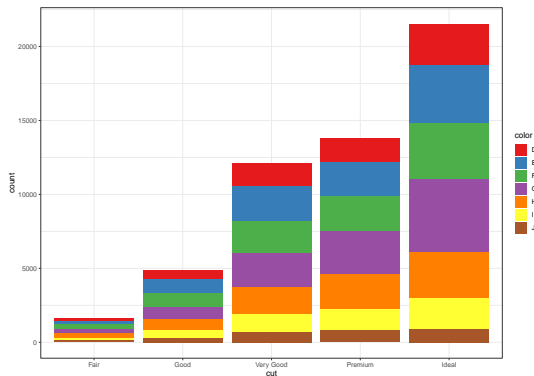
Eric Marcon

Tidyverse

Graphiques
avec ggplot

Aller plus loin

```
ggplot(data = diamonds) +  
  geom_bar(mapping = aes(x = cut, fill = color), position = "stack") +  
  scale_fill_brewer(palette = "Set1")
```



Système de coordonnées :

- `coord_flip()` intervertit x et y,
- `coord_polar()` : coordonnées polaires,
- `coord_trans()` transforme l'affichage des coordonnées (mais pas les données comme `scale_()`),
- etc.

Exemple : tracer la carte des wapas de la parcelle 6.

Coordonnées (2)

Mise à niveau
R 3/3

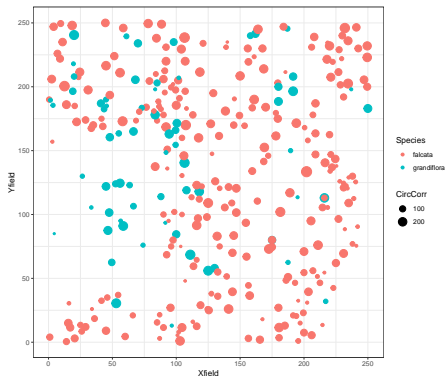
Eric Marcon

Tidyverse

Graphiques
avec ggplot

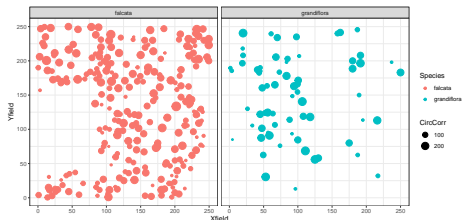
Aller plus loin

```
read_csv2("data/Paracou6.csv") %>% filter(Genus == "Eperua") %>%
  ggplot() +
    geom_point(aes(x = Xfield, y = Yfield, size = CircCorr, color = Species))
coord_fixed() -> P6Map
P6Map
```



Présente plusieurs aspects du même graphique:

P6Map + facet_wrap(~ Species)



Possibilité d'affiner un graphique

Mise à niveau
R 3/3

Eric Marcon

Tidyverse

Graphiques
avec ggplot

Aller plus loin

Aller plus loin



Plutôt qu'un code commenté, un texte avec du code.

Tricot : production de documents HTML ou PDF.

Rédaction d'articles, de mémoires, de diaporama.

Reproductibilité : le projet contient les données, le code, le texte et le modèle de mise en forme.

Galerie : <https://ericmarcon.github.io/memoiR/>

git (contrôle de source) et GitHub (plateforme web) pour :

- tracer les versions d'un projet,
- collaborer,
- tester le code automatiquement,
- tricoter automatiquement.

Exemple: <https://github.com/EricMarcon/travailleR>

Il existe des packages pour tout.

Exemples :

- Site web : <https://ericmarcon.github.io/>,
- Application Shiny :
https://vac-lshtm.shinyapps.io/ncov_tracker/,
- TP en ligne :
<https://eric-marcon.shinyapps.io/TP-Biodiversite/>.

Mise à niveau
R 3/3

Eric Marcon

Tidyverse

Graphiques
avec ggplot

Aller plus loin