# Pig Data

Adrien Gluckman

2024-11-03

```
# Charger les librairies nécessaires
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(RColorBrewer)
# import data
train <- read.csv("~/Documents/M2QF/MOSA/Projet_Pig_data/pig_data_proj/train1.csv")
test <- read.csv("~/Documents/M2QF/MOSA/Projet_Pig_data/pig_data_proj/test1.csv")
```

## Introduction

The aim of this study is to predict the weight of pig in differents farms. Our datasets contains 9 explanatory variables and our target the weight as described below :

```
head(train)
```

```
##   Farm        Day ID Species Gender Age Weight Chest Length NumberID
## 1    1 2020-08-08  3       2      2   4    9.0    NA     NA        2
## 2    1 2020-08-15  3       2      2   5   11.5    NA     NA        2
## 3    1 2020-08-22  3       2      2   6   15.5    NA     NA        2
## 4    1 2020-08-29  3       2      2   7   20.0    NA     NA        2
## 5    1 2020-09-05  3       2      2   8   21.0    64     52        2
## 6    1 2020-09-12  3       2      2   9   24.0    67     50        2
```

## Data exploratory

```
summary(train)
```

```
##       Farm          Day                  ID          Species
##  Min.   :1.00   Length:2729        Min.   : 1.00   Min.   :1.000
##  1st Qu.:2.00   Class :character   1st Qu.: 7.00   1st Qu.:1.000
##  Median :4.00   Mode  :character   Median :13.00   Median :1.000
```

```
##  Mean   :3.93                     Mean   :12.78   Mean   :1.531
##  3rd Qu.:6.00                     3rd Qu.:19.00   3rd Qu.:2.000
##  Max.   :7.00                     Max.   :28.00   Max.   :3.000
##
##      Gender           Age             Weight           Chest
##  Min.   :1.000   Min.   : 3.00   Min.   :  5.30   Min.   : 57.00
##  1st Qu.:1.000   1st Qu.: 9.00   1st Qu.: 23.00   1st Qu.: 74.00
##  Median :1.000   Median :13.00   Median : 46.00   Median : 88.00
##  Mean   :1.387   Mean   :13.48   Mean   : 50.84   Mean   : 89.12
##  3rd Qu.:2.000   3rd Qu.:18.00   3rd Qu.: 77.50   3rd Qu.:101.00
##  Max.   :2.000   Max.   :25.00   Max.   :124.10   Max.   :138.00
##                                                   NA's   :1404
##      Length          NumberID
##  Min.   : 46.00   Min.   :  2.0
##  1st Qu.: 75.00   1st Qu.: 42.0
##  Median : 86.00   Median : 83.0
##  Mean   : 92.21   Mean   : 83.3
##  3rd Qu.:111.00   3rd Qu.:123.0
##  Max.   :148.00   Max.   :170.0
##  NA's   :1404
```

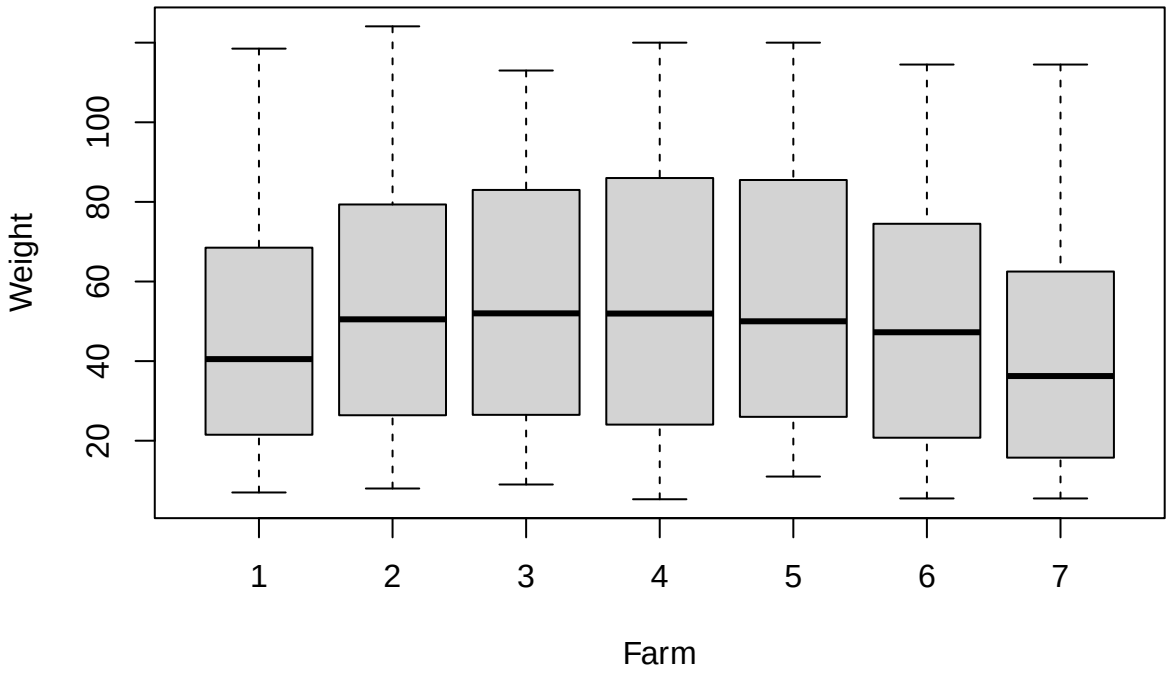
## Missing values

```
colSums(is.na(train))
```

```
##     Farm      Day       ID  Species   Gender      Age   Weight    Chest
##        0        0        0        0        0        0        0     1404
##   Length NumberID
##     1404        0
```

## Weight by Farms

```
boxplot(Weight ~ Farm, data = train, main = "Distribution des poids par ferme")
```



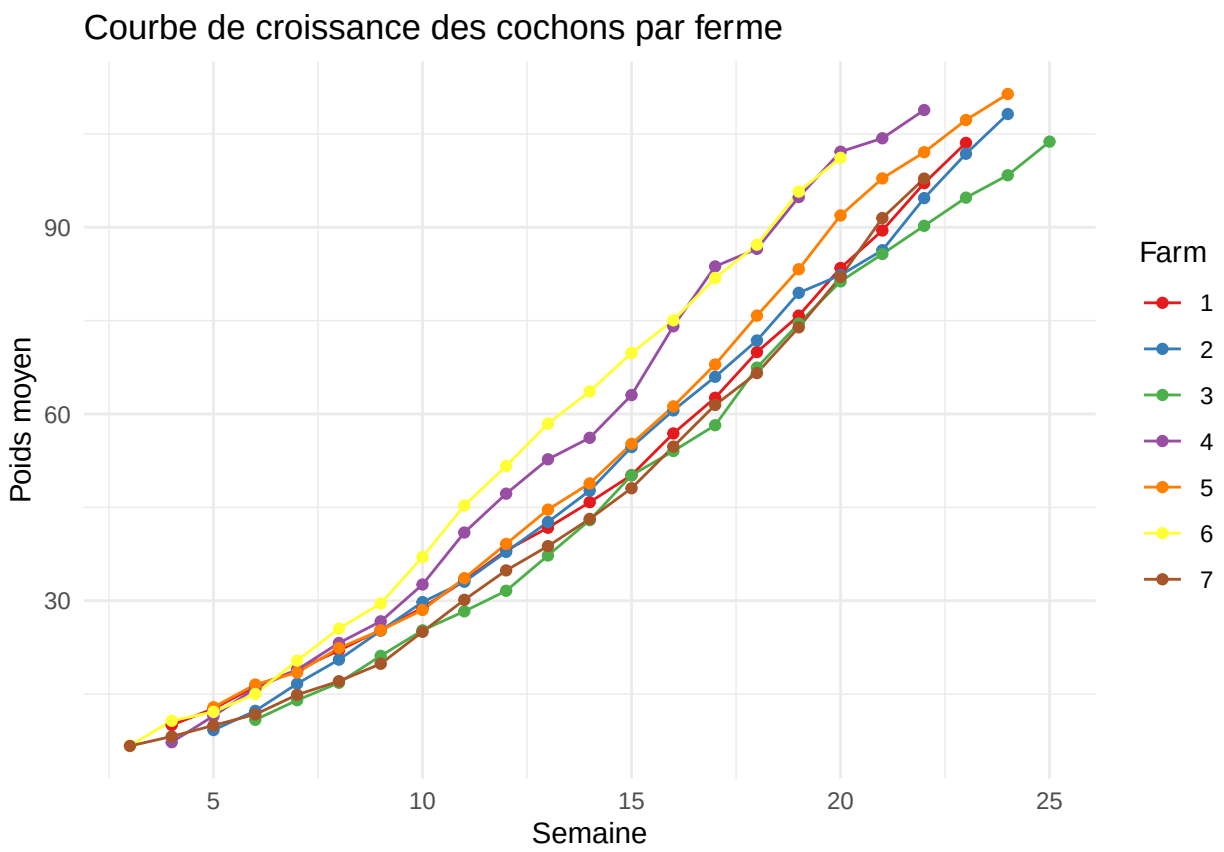

```r
# Charger les bibliothèques nécessaires
library(dplyr)
library(ggplot2)
library(RColorBrewer)

# S'assurer que Farm est un facteur
train$Farm <- as.factor(train$Farm)

# Calculer la moyenne du poids par semaine et par ferme, avec .groups = "drop"
mean_weight <- train %>%
    group_by(Farm, Age) %>%
    summarise(moyenne_poids = mean(Weight, na.rm = TRUE), .groups = "drop")

# Choisir une palette de couleurs bien contrastées
ferme_colors <- brewer.pal(7, "Set1")  # "Set1" est une palette de couleurs discrètes bien distinctes

# Tracer les courbes de croissance avec des couleurs personnalisées
ggplot(mean_weight, aes(x = Age, y = moyenne_poids, color = Farm)) +
    geom_line() +
    geom_point() +
    scale_color_manual(values = ferme_colors) +  # Utilise les couleurs définies
    labs(title = "Courbe de croissance des cochons par ferme", x = "Semaine", y = "Poids moyen") +
    theme_minimal()
```




```
desc_stats <- train %>%
    group_by(Farm) %>%
    summarise(
        moyenne = mean(Weight, na.rm = TRUE),
        mediane = median(Weight, na.rm = TRUE),
        ecart_type = sd(Weight, na.rm = TRUE),
        min = min(Weight, na.rm = TRUE),
        max = max(Weight, na.rm = TRUE),
        .groups = "drop"  # Désactive le regroupement après summarise()
    )

print(desc_stats)
```

```
## # A tibble: 7 x 6
##   Farm  moyenne mediane ecart_type   min   max
##   <fct>   <dbl>   <dbl>      <dbl> <dbl> <dbl>
## 1 1        46.9    40.5       28.8   7    118.
## 2 2        54.0    50.5       31.2   8    124.
## 3 3        54.3    52         30.6   9    113
## 4 4        54.5    52.0       33.2   5.3  120
## 5 5        56.3    50         32.0  11    120
## 6 6        49.3    47.2       30.7   5.5  114.
## 7 7        41.4    36.2       28.7   5.5  114.
```

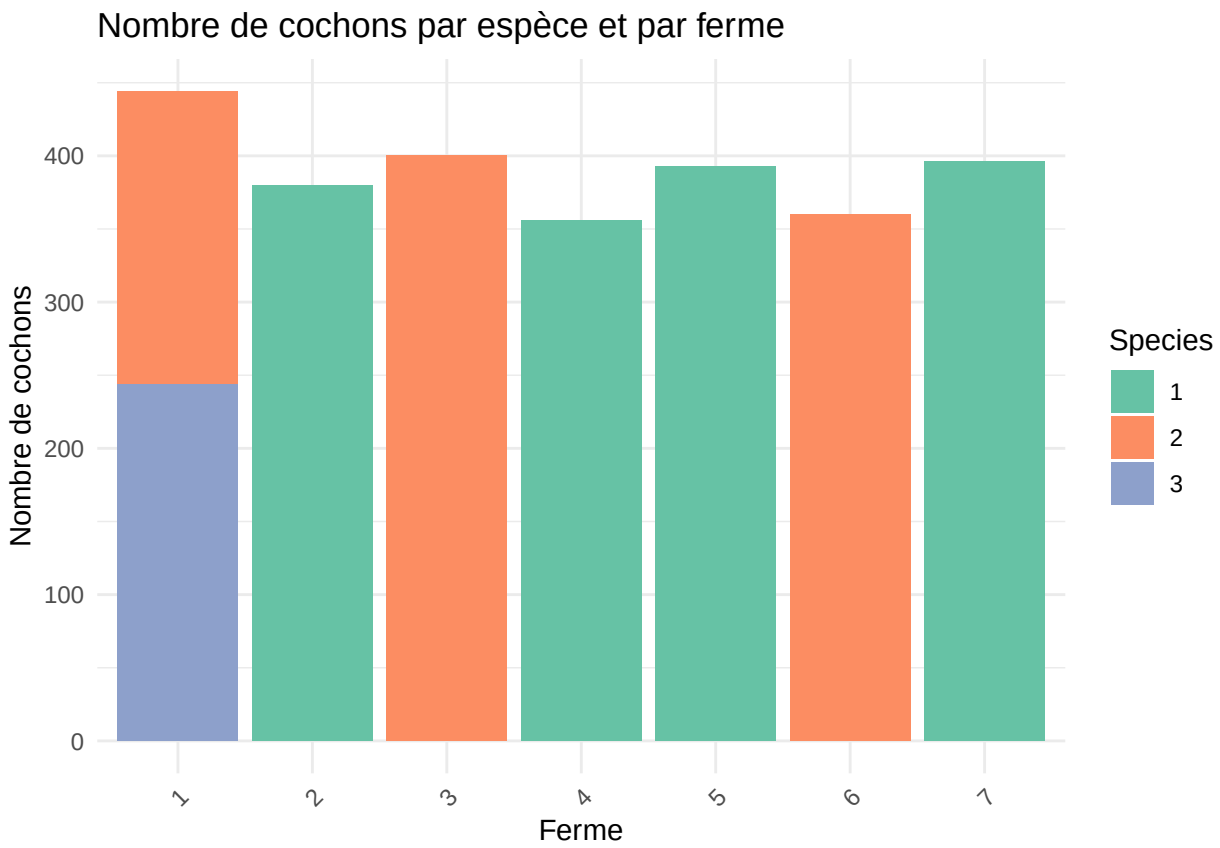
### Species by farm

```
# Vérifiez que la variable Species est un facteur
train$Species <- as.factor(train$Species)

# Compter le nombre de cochons de chaque espèce par ferme
species_count <- train %>%
    group_by(Farm, Species) %>%
    summarise(count = n(), .groups = "drop")

# Choisir une palette de couleurs bien contrastées
# Vous pouvez choisir une palette de couleurs avec plus de couleurs si nécessaire
ferme_colors <- brewer.pal(n = length(unique(species_count$Species)), name = "Set2")

# Créer le graphique
ggplot(species_count, aes(x = Farm, y = count, fill = Species)) +
    geom_bar(stat = "identity", position = "stack") +
    scale_fill_manual(values = ferme_colors) + # Utilisation de la palette de couleurs personnalisée
    labs(title = "Nombre de cochons par espèce et par ferme",
         x = "Ferme",
         y = "Nombre de cochons") +
    theme_minimal() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1)) # Rotation des étiquettes de l'axe x pour
```




```
# Calculer le poids moyen par espèce et par ferme
species_weight_stats <- train %>%
    group_by(Species) %>%
```

```
    summarise(
        poids_moyen = mean(Weight, na.rm = TRUE),
        ecart_type = sd(Weight, na.rm = TRUE),
        .groups = "drop"
    )

print(species_weight_stats)
```

```
## # A tibble: 3 x 3
##   Species poids_moyen ecart_type
##   <fct>         <dbl>      <dbl>
## 1 1              51.4       31.8
## 2 2              51.5       30.5
## 3 3              44.4       27.6
```

**Evolution of the weight regarding species**

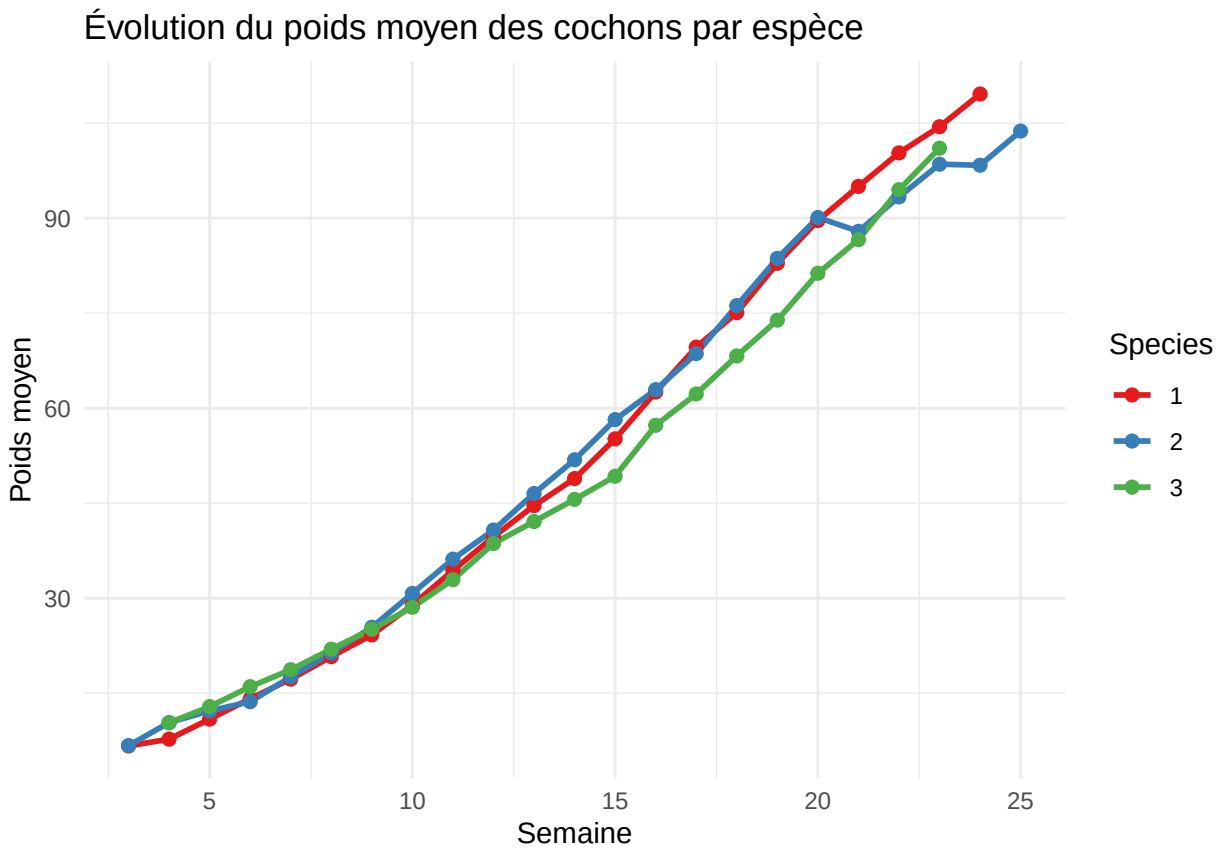
```
# Vérifiez que la variable Species est un facteur
train$Species <- as.factor(train$Species)

# Calculer le poids moyen par espèce et par semaine (Age)
species_weight_evolution <- train %>%
    group_by(Species, Age) %>%
    summarise(
        poids_moyen = mean(Weight, na.rm = TRUE),
        .groups = "drop"
    )

# Choisir une palette de couleurs bien contrastées
ferme_colors <- brewer.pal(n = length(unique(species_weight_evolution$Species)), name = "Set1")

# Créer le graphique
ggplot(species_weight_evolution, aes(x = Age, y = poids_moyen, color = Species)) +
    geom_line(size = 1) +           # Courbe pour chaque espèce
    geom_point(size = 2) +          # Points pour chaque valeur
    scale_color_manual(values = ferme_colors) +
    labs(title = "Évolution du poids moyen des cochons par espèce",
         x = "Semaine",
         y = "Poids moyen") +
    theme_minimal()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```


Évolution du poids moyen des cochons par espèce
Poids moyen
90
60
30
5
10
15
20
25
Semaine
Species
1
2
3