

Calcule ta route

Daniel Lavallée

Mathieu Sévégné

Tristan Lafontaine

Émile Bois

Vincent Kilaknowvski

2024-10-31

Contents

Analyse des données réelles vs. les données mesurées par le capteur de distance	2
Lecture des données	2
Analyse des données	3

Analyse des données réelles vs. les données mesurées par le capteur de distance

Lecture des données

```
## Fonction qui vérifie si un packet est installé et qui l'installe avant
## de le charger au besoin.
loadPackage <- function(package) {
  if (!require(package, character.only = TRUE)) {
    install.packages(package, quiet = TRUE)
    library(package, character.only = TRUE, quietly = TRUE)
  }
  else library(package, character.only = TRUE, quietly = TRUE)
}
```

```
loadPackage("tidyverse")
loadPackage("ggplot2")
loadPackage("readxl")
loadPackage("Metrics")
```

Lecture des données lues par le capteur de distance par rapport à la distance réelle.

```
## Chemin vers les données
path_donnees_obstacle <- "../data/donnees_detecteur_obstacle.xlsx"

## Lecture des données
donnees_detecteur_obstacle <- read_excel(path_donnees_obstacle)
donnees_detecteur_obstacle <- donnees_detecteur_obstacle %>%
  set_names(c("real_distance_cm", "measured_distance_cm"))

## Aperçu des données
head(donnees_detecteur_obstacle, 3)
```

```
## # A tibble: 3 x 2
##   real_distance_cm measured_distance_cm
##           <dbl>           <dbl>
## 1             2             -1
## 2             5              2
## 3            10              6
```

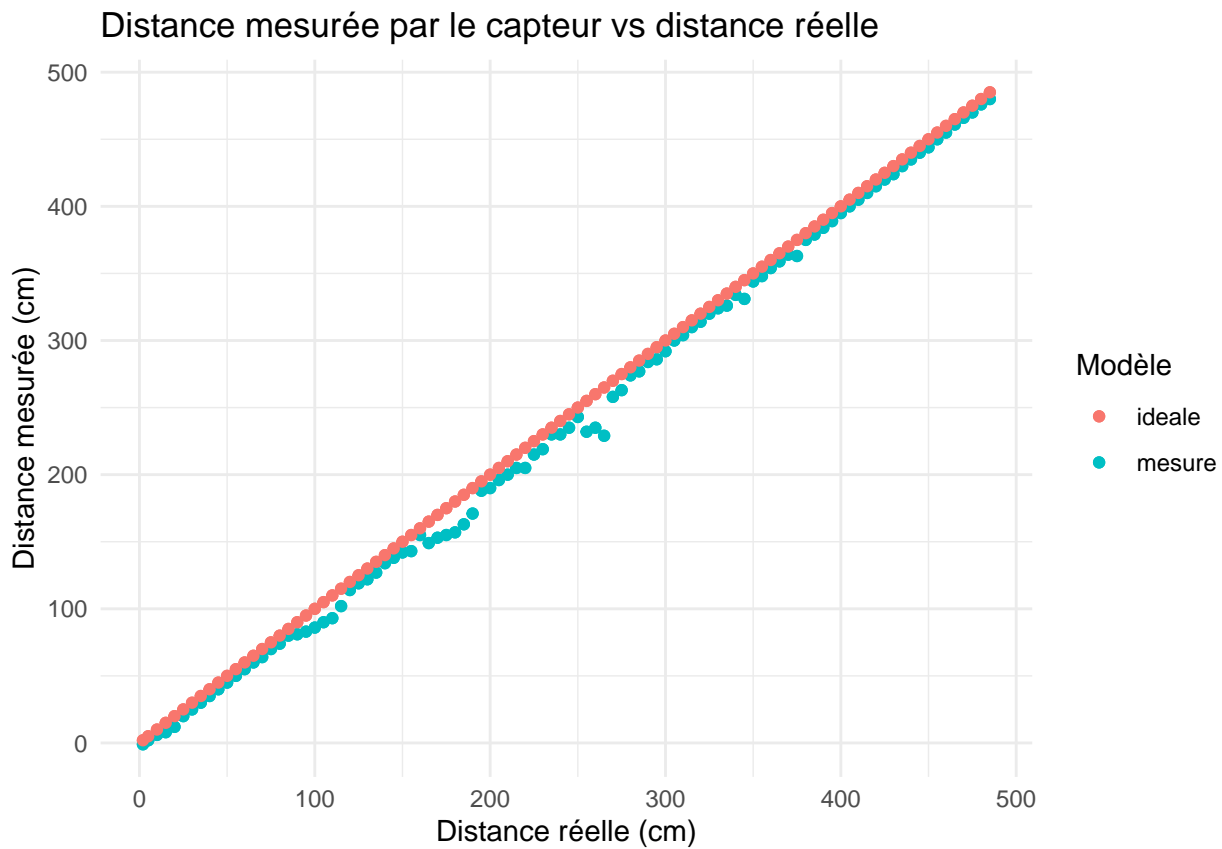
Analyse des données

```
## Jeu de données contenant les données mesurées.
donnees_mesurees <- tibble(
  real_distance_cm = donnees_detecteur_obstacle$real_distance_cm,
  measured_distance_cm = donnees_detecteur_obstacle$measured_distance_cm,
  Modèle = "mesure"
)

## Jeu de données contenant les données idéales.
donnees_ideales <- tibble(
  real_distance_cm = donnees_detecteur_obstacle$real_distance_cm,
  measured_distance_cm = donnees_detecteur_obstacle$real_distance_cm,
  Modèle = "ideale"
)

## Fusion des deux jeux de données
all_data <- donnees_mesurees %>% bind_rows(donnees_ideales)

## Affichage des données
ggplot(all_data,
  aes(x = real_distance_cm,
      y = measured_distance_cm,
      color = Modèle)) +
  geom_point() +
  geom_point(aes(x = real_distance_cm, y = measured_distance_cm)) +
  labs(title = "Distance mesurée par le capteur vs distance réelle",
    x = "Distance réelle (cm)",
    y = "Distance mesurée (cm)") +
  theme_minimal()
```



```

## Erreurs de mesure et coefficient de détermination
## Calcul de l'erreur de mesure (RMSE et MAE)
RMSE_mesure <- rmse(donnees_detecteur_obstacle$real_distance_cm,
                    donnees_detecteur_obstacle$measured_distance_cm)
MAE_mesure <- mae(donnees_detecteur_obstacle$real_distance_cm,
                 donnees_detecteur_obstacle$measured_distance_cm)

## Coefficient de détermination
R2_mesure <- cor(donnees_detecteur_obstacle$real_distance_cm,
                 donnees_detecteur_obstacle$measured_distance_cm)^2

## Affichage des résultats
print(paste("La RMSE est de: ", RMSE_mesure))

## [1] "La RMSE est de:  10.109098750681"
print(paste("La MAE est de: ", MAE_mesure))

## [1] "La MAE est de:  8.47959183673469"
print(paste("Le coefficient de détermination est de: ", R2_mesure))

## [1] "Le coefficient de détermination est de:  0.998530428038407"

```