Sensitivity of CO2 emission rate to temperature in cranberry soils

Dr.Wilfried Dossou-Yovo

13/02/2022

Wilfried Dossou-Yovo, Serge Étienne Parent,Noura Ziadi, Bertrand Normand, and Léon Étienne parent

```
print(Sys.Date())
```

```
## [1] "2022-10-10"
```

1. Objective

This notebook generate the result of CO2 data analysis. Data set contains a collection of soil caracteristics, measured co2 emission collected from incubation study. Soil samples was collected from two cranberry fied stand of eastern canada. Incubation study was carried out at Agriculture and Agri-food Canada(sainte-foy, quebec,qc) from February to Mai 2019. The aim of this study was to measure CO2 emission rates in cranberry soils of Eastern Canada as related to soil temperature and depth

2. Statistical questions

In addition to data exploration, this notebook will answer the following statistical questions.

1. What is the influence of soil depth and temperature on CO2 emission?
2. Can Arrhenius equation and Q10 be useful to describe temperature sensitivity of carbon decomposition across layers?

3. Packages

We need package tidyverse which loads a set of packages for easy data manipulation(Ex: dplyr) and visualization (ex: ggplot2). We also use ggpubr to customise publication ready plot, ggpmisc and grid are useful packages as extensions to ggplot2.

4. Import data

We load two data `data_pot` and `data_co2` involved in our anylisis. `data_pot` contained details about sites sampling, soil sampling(soil depth, weight, water content and bulk density), laboratory incubation temperature while `data_co2` contained details about laboratory incubation time, co2 emission and jar masson details. `data_co2` was combined with `data_pot` with `left_join` function

ID pot	Sites	Depth (cm)	Repetition	Temperature (°C)	Pot weight (g)	Soil weight (g)	Water volume (ml)	Water content (%)	Bulk density (g/cm3)
6	A9	10	1	10	244.81	110.32	42.1	10.3	0.89
21	A9	10	1	20	251.14	110.33	42.1	10.3	0.89
54	A9	10	1	30	249.63	110.33	42.1	10.3	0.89

ID potSites	Depth (cm)	Repetition	Temperature (°C)	Pot weight (g)	Soil weight (g)	Water volume (ml)	Water content (%)	Bulk density (g/cm3)
18A9	10	2	10	246.04	124.93	27.5	24.9	0.89
59A9	10	2	20	248.21	124.93	27.5	24.9	0.89
60A9	10	2	30	255.03	124.90	27.5	24.9	0.89

5. Some calculations

Several variables have been added to our data in order to proceed for analysis. The added variables are the following: `Temperature (Kelvin)`, `Molar Volume (L/mol)`, `Headspace Volume (mL)`, `Dry soil weight (g)`, `CO2 emission (ug/h/g)`, `CO2 emission (mg/kg)`, `decomposition rate K`, `lnK`and `1/T(T = Temperature(Kelvin)`

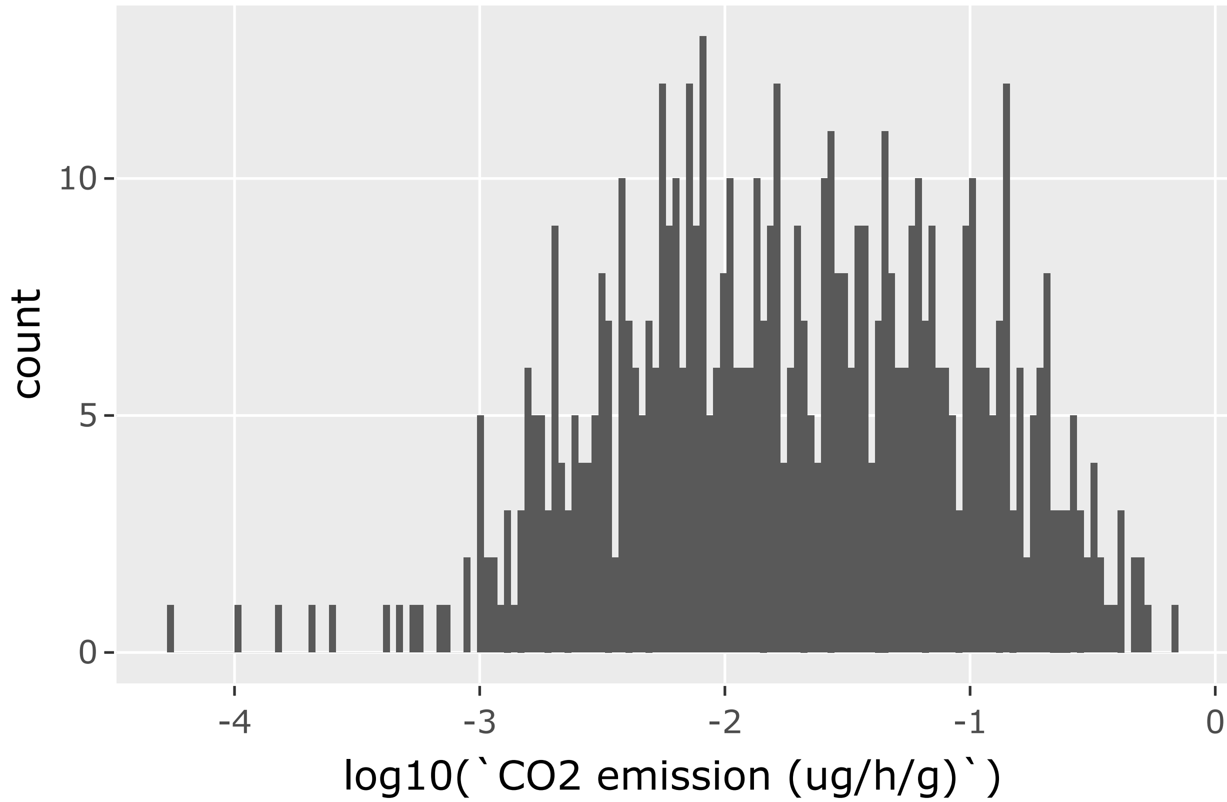
6. Exploratory data analysis

6.1 Histogram

```
New.labs <- c("10°C", "20°C", "30°C") # Change labels
names(New.labs) <- c("10", "20", "30")

New.labs_b <-  c("[0-10 cm]", "[10-20 cm]", "[20-30 cm]") # Change labels
names(New.labs_b) <- c("10", "20", "30")
```

```
library(plotly)
ggplotly(
  data_co2 |>
    ggplot() +
    geom_histogram(aes(x = log10(`CO2 emission (ug/h/g)`)), bins = 150)
)
```

```
## Warning in FUN(X[[i]], ...): NaNs produced

## Warning in FUN(X[[i]], ...): NaNs produced

## Warning: Removed 37 rows containing non-finite values (stat_bin).
```



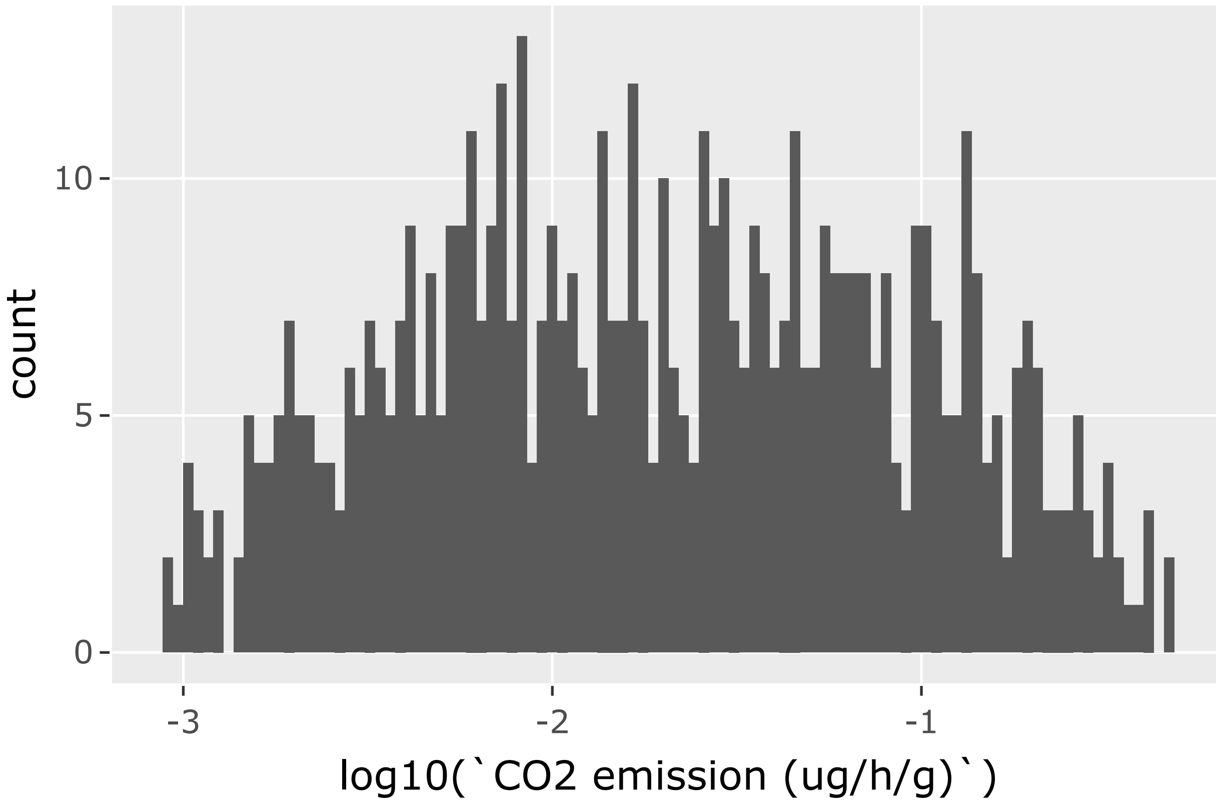
Data contains some outliers, let remove them

```
data_co2_clean <- data_co2 |>
  mutate(log_tr = log10(`CO2 emission (ug/h/g)`)) |>
  filter(log_tr > -3.06 & log_tr < -0.33) |>
  drop_na()
```

```
## Warning in mask$eval_all_mutate(quo): NaNs produced
```

Now data look well distributed

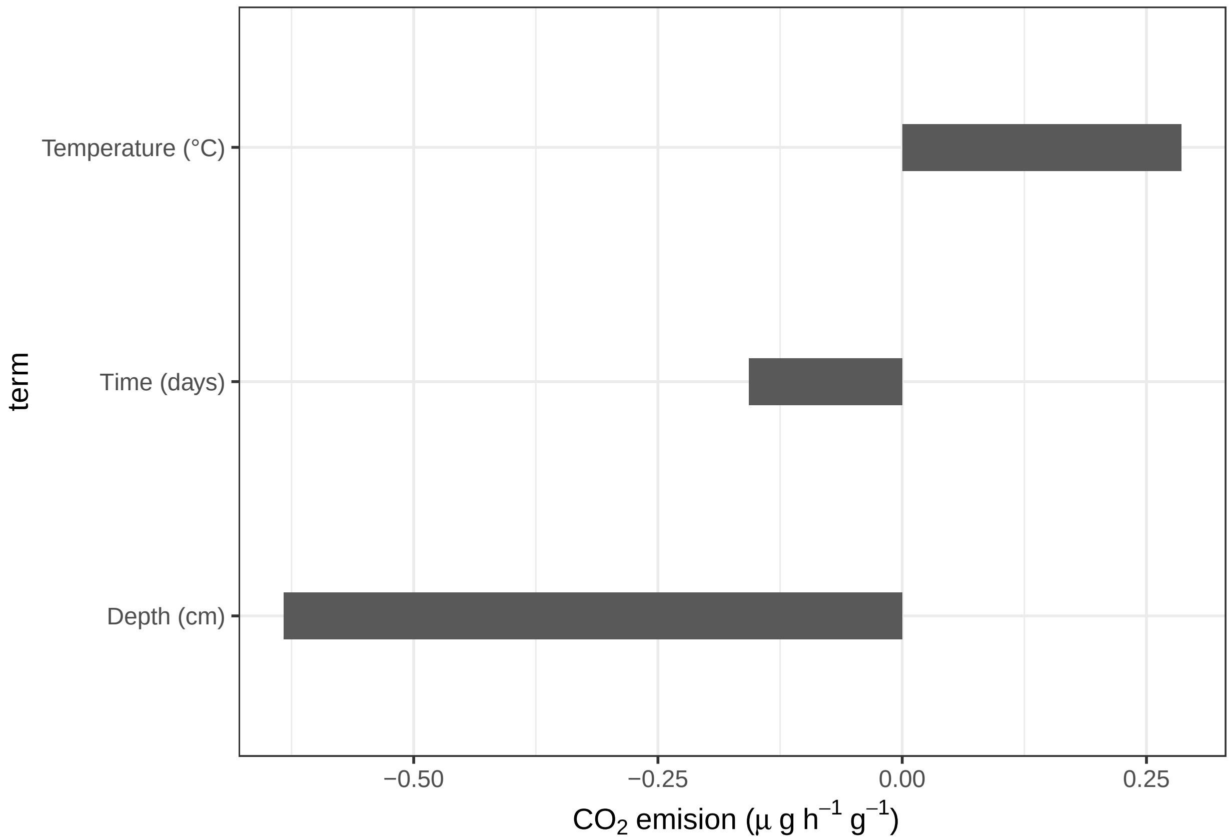
```
ggplotly(
  data_co2_clean |>
    ggplot() +
    geom_histogram(aes(x = log10(`CO2 emission (ug/h/g)`)), bins = 100)
)
```



6.2 Correlations

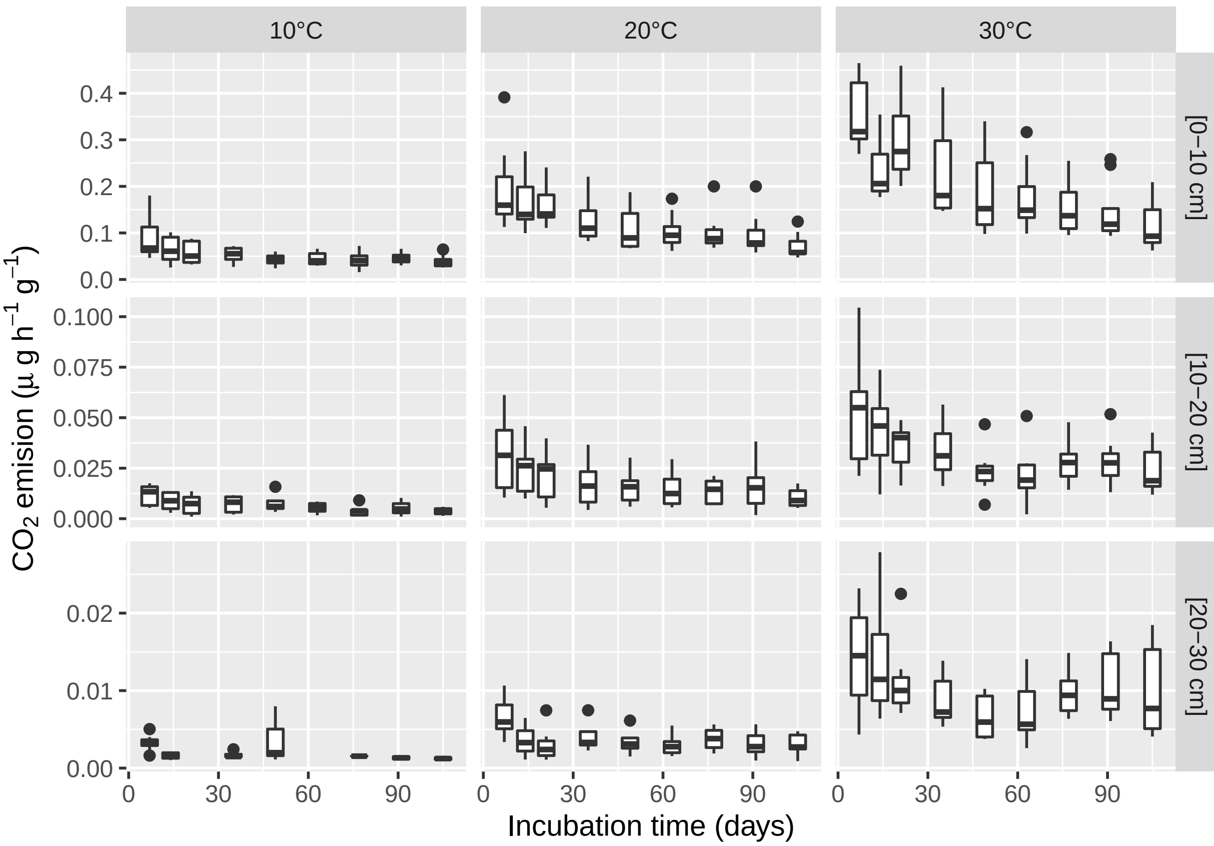
```
data_co2_clean |>
  select(`Time (days)`, `Depth (cm)`, `Temperature (°C)`, `CO2 emission (ug/h/g)`) |>
  corrr::correlate() |>
  corrr::focus(`CO2 emission (ug/h/g)`) |>
  mutate(term = fct_reorder(term, `CO2 emission (ug/h/g)`)) |>
  ggplot(aes(x = `CO2 emission (ug/h/g)`, y= term)) +
  geom_col(width = 0.2) +
  labs(x = bquote(~CO[2]~ 'emision ('*mu~'g'~ h^-1~g^-1*')')) +
  theme_bw()
```

```
##
## Correlation method: 'pearson'
## Missing treated using: 'pairwise.complete.obs'
```



6.3 Boxplot

```
options(repr.plot.width = 6, repr.plot.height = 7)
pg <- ggplot(data=data_co2_clean, aes(x = `Time (days)`,y = `CO2 emission (ug/h/g)` )) +
  geom_boxplot(aes(group = factor(`Time (days)`))) +
  facet_grid(`Depth (cm)` ~ `Temperature (°C)`,   scales = "free",
             labeller = labeller(`Depth (cm)` = New.labs_b, `Temperature (°C)` = New.labs))+
  labs(x = "Incubation time (days)", y = bquote(~CO[2]~ 'emision ('*mu~'g'~ h^-1~g^-1*')'))
pg
```




```
ggsave("figures/Boxplot.png", width = 6, height = 7, dpi = 600)# export plot high resolution
```

7.What is the influence of soil depth and temperature on CO2 emission?

7.1 Build model: linear regression

```
model_rec <-  data_co2_clean |>
  recipe(`CO2 emission (ug/h/g)` ~ ., data_co2) |>
  step_select(`CO2 emission (ug/h/g)`, `Time (days)`, Sites,
              `Depth (cm)`, `Temperature (°C)`) |>
  step_log(all_outcomes(), base = 10) |>
  step_dummy(Sites) |>
  step_normalize(all_numeric(), -all_outcomes() ) |>
  prep()

data_co2_preprocessed <-  juice(model_rec)
model_spec <- linear_reg() |>
  set_engine("lm")
```

```
model_fit <- model_spec |>
  fit(`CO2 emission (ug/h/g)` ~ ., data_co2_preprocessed)
```

Fit model

```r
tidy(model_fit)
```

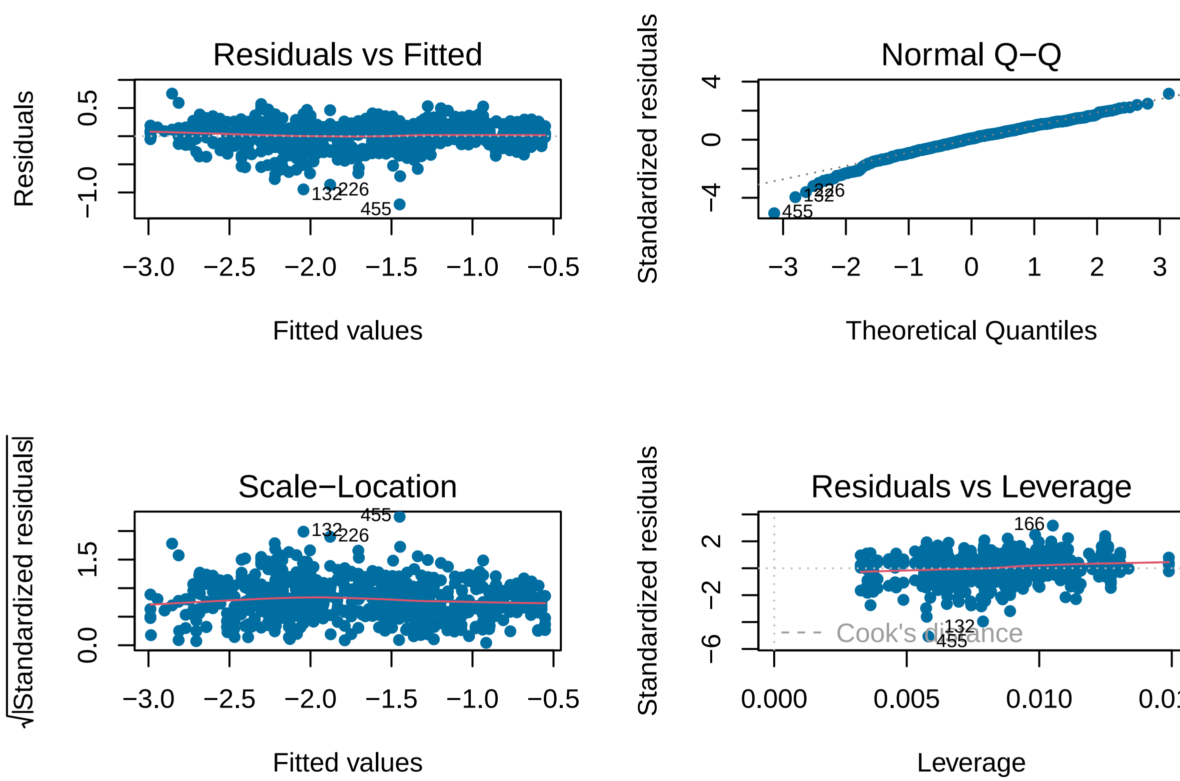
Exploring model results

```
## # A tibble: 5 x 5
##   term                  estimate std.error statistic   p.value
##   <chr>                    <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)            -1.70     0.00983   -173.    0
## 2 `Time (days)`          -0.103    0.00985    -10.5   1.22e- 23
## 3 `Depth (cm)`           -0.579    0.00993    -58.3   3.08e-247
## 4 `Temperature (°C)`      0.273    0.00992     27.6   3.81e-108
## 5 Sites_PF45             -0.0202   0.00984     -2.05 4.11e-  2
```

```r
glance(model_fit)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic   p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>     <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.868         0.867 0.240      969. 6.71e-258     4   7.36 -2.71  23.6
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

```r
par(mfrow=c(2,2)) # plot all 4 plots in one

plot(model_fit$fit,
     pch = 16,
     col = '#006EA1')
```



Inspect the model

```
prediction <-  model_fit |>
  predict(data_co2_preprocessed)
```

Prediction

```
rmse <-  data_co2_preprocessed |>
  bind_cols(prediction) |>
  rmse(`CO2 emission (ug/h/g)`, .pred)
rmse
```

collect Metrics

```
## # A tibble: 1 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard       0.239
```

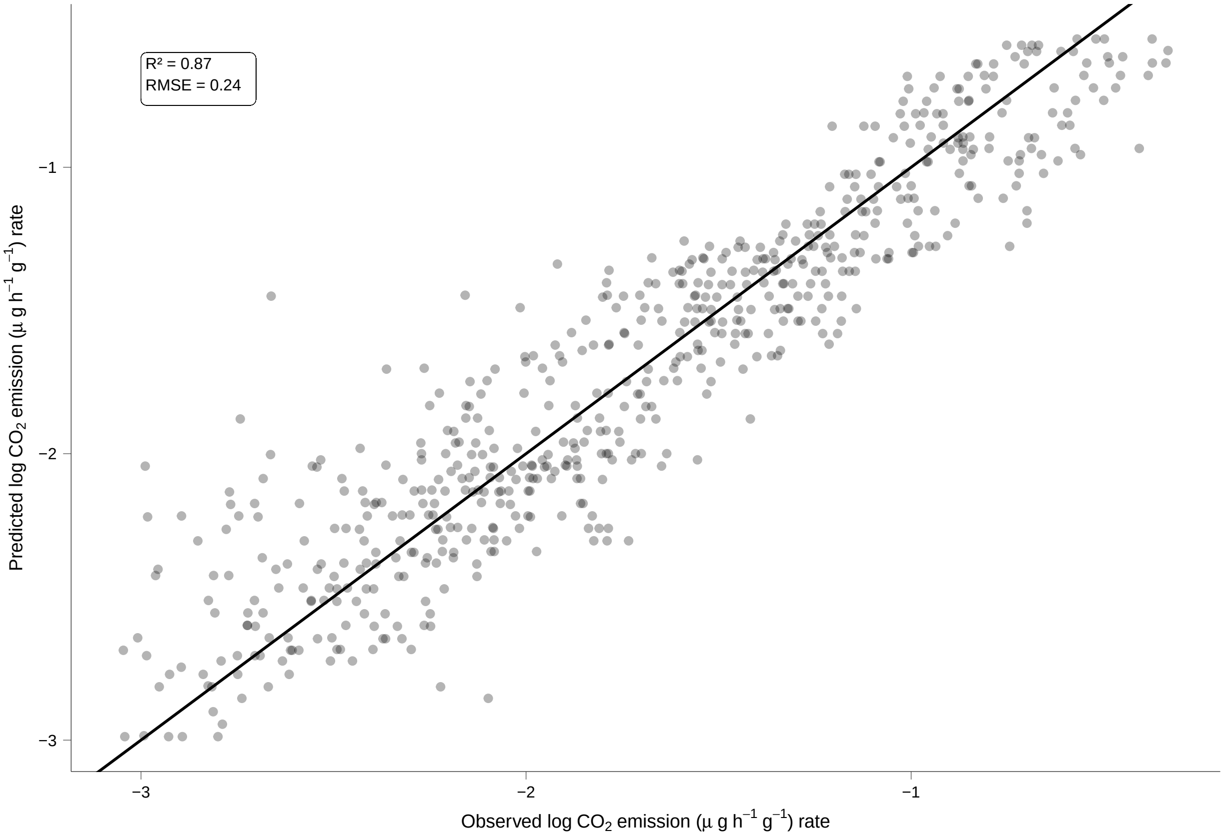
```
rmse <-  round(as.numeric(rmse[1,3]), 2)
```

```
rsq <-  data_co2_preprocessed |>
  bind_cols(prediction) |>
  rsq(`CO2 emission (ug/h/g)`, .pred)
rsq
```

```
## # A tibble: 1 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rsq     standard       0.868
```

```
rsq <- round(as.numeric(rsq[1,3]), 2)
```

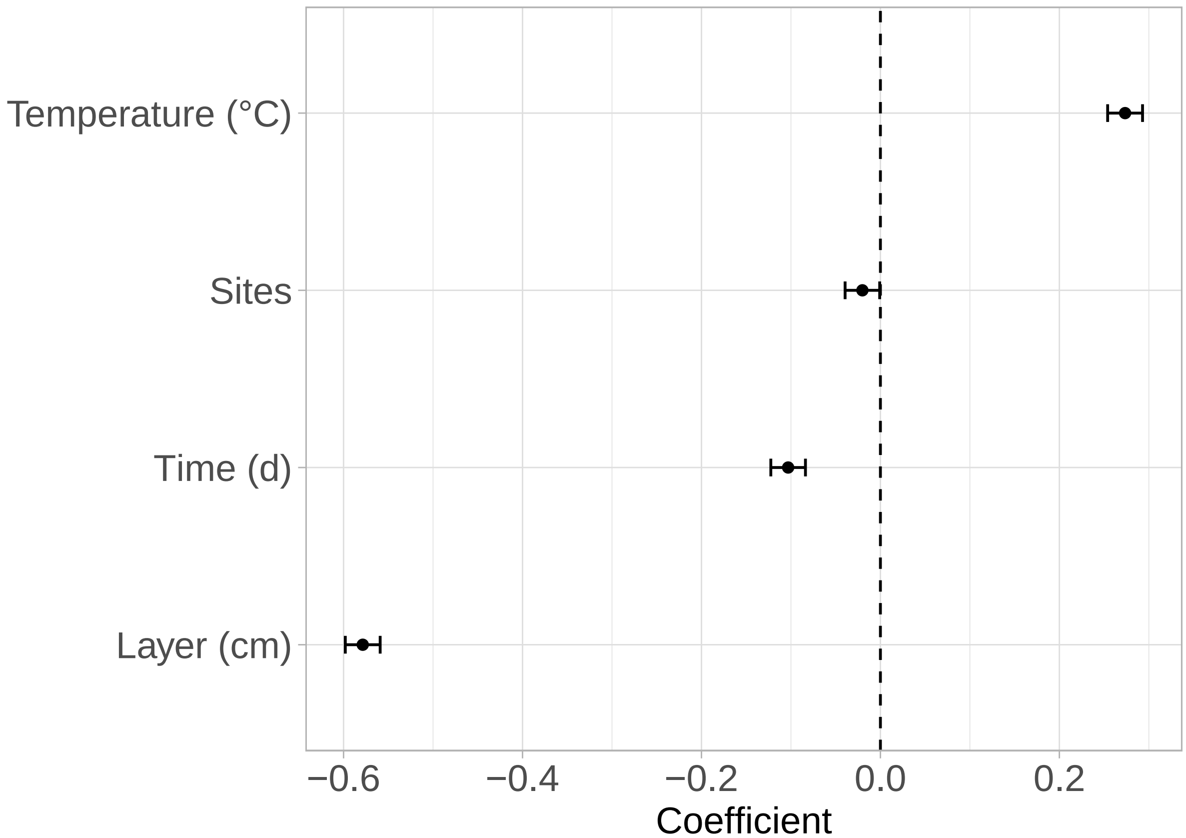
```
options(repr.plot.width = 4, repr.plot.height = 2)
px <- data_co2_preprocessed |>
  bind_cols(prediction) |>
  ggplot(aes(x = `CO2 emission (ug/h/g)`, y = .pred)) +
  geom_point(size = 1, alpha = .3) +
  geom_label(aes(x = -3, y = -.6),
             vjust = 1, hjust = 0, size = 2, label.size = 0.1,
             label = paste("R² =",  rsq, "\nRMSE =", rmse)) +
  geom_abline() +
  theme_pubr()  +
  theme(axis.title=element_text(size=7),
        axis.line = element_line(size = 0.1),
        axis.ticks = element_line(size = 0.1),
        axis.text = element_text(size = 6)) +
  labs(x= bquote("Observed log"~CO[2]~ 'emission ('*mu~'g'~ h^-1~g^-1*')'~"rate") , y = bquote("Predicte
px
```




```
ggsave("figures/Observed and predicted co2 emission.png", width = 4, height = 2.2, dpi = 600)
```

7.3 Variable coefficient and confidence intervals

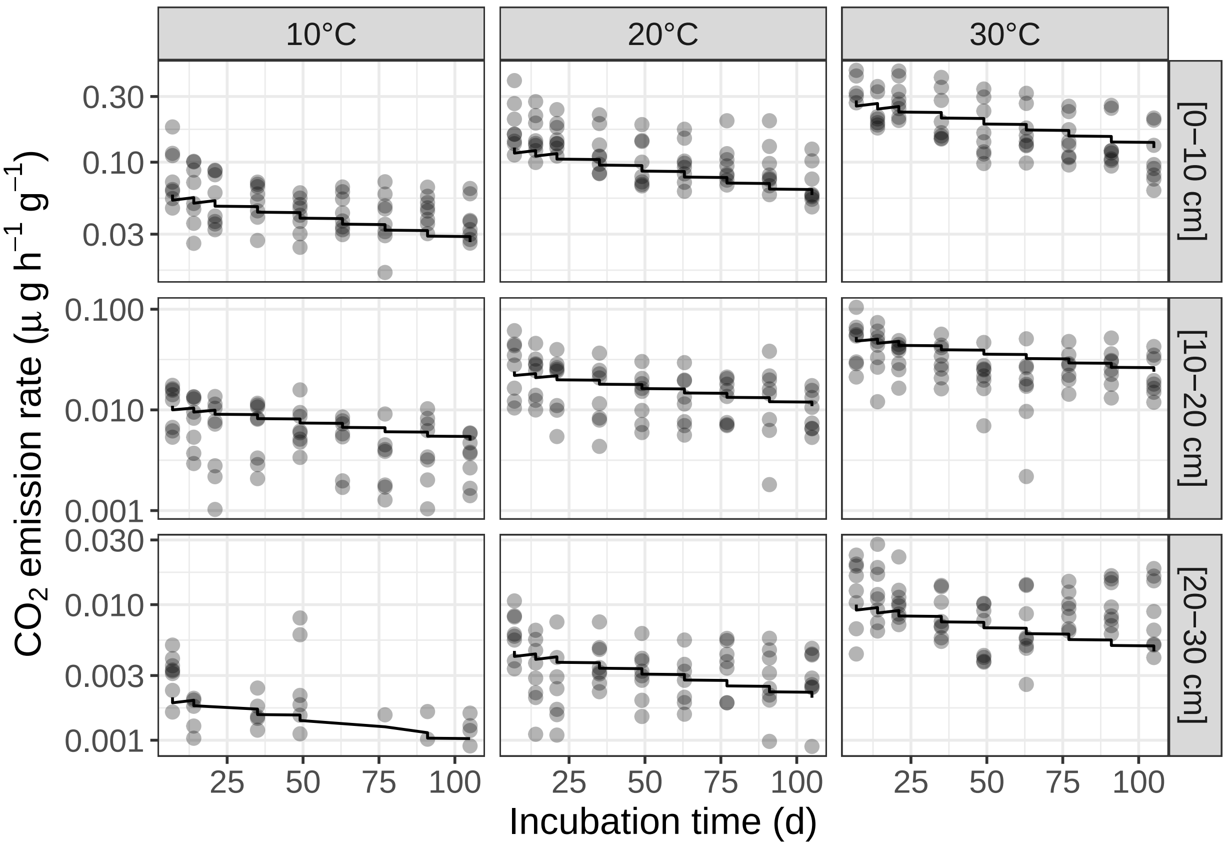
```
options(repr.plot.width = 8, repr.plot.height = 2)
term_rename <- tibble(term = c("`Time (days)`", "`Depth (cm)`", "`Temperature (°C)`", "Sites_PF45"),
                      name_corrected = c("Time (d)", "Layer (cm)", "Temperature (°C)", "Sites"))

h <- broom::tidy(model_fit, conf.int = TRUE) |>
  dplyr::filter(term != "(Intercept)") |>
  left_join(term_rename, by = "term") |>
  mutate(term_rename = fct_reorder(name_corrected, estimate)) |>
  ggplot(aes(estimate, term_rename)) +
  geom_vline(xintercept = 0, linetype = 2) +
  geom_point() +
  geom_errorbarh(aes(xmin = conf.low, xmax = conf.high), height = 0.1,
                 size=0.5) +
  labs(x = "Coefficient", y = "") +
  theme_light() +
  theme(axis.text=element_text(size=14),
        axis.title=element_text(size=14)) # Time (d)
h
```




```
ggsave("figures/Linear-model-Co2_with_site.png", width = 8, height = 2, dpi = 600)
```

7.3 Prediction model of CO2 emission in cranberry soils in three-layer positions (0-10 cm, 10-20 cm, 20-30 cm) and at three temperatures (10, 20 and 30oC).

```
New.labs <- c("10°C", "20°C", "30°C") # Change labels
names(New.labs) <- c("10", "20", "30")

New.labs_b <-  c("[0-10 cm]", "[10-20 cm]", "[20-30 cm]") # Change labels
names(New.labs_b) <- c("10", "20", "30")

options(repr.plot.width = 8, repr.plot.height = 6)
pl <- data_co2_clean |>
  bind_cols(10^prediction) |>
  ggplot(aes(x = `Time (days)`, y = `CO2 emission (ug/h/g)`)) +
  geom_point(size = 2, alpha = .3) +
  facet_grid(`Depth (cm)` ~ `Temperature (°C)`, scales = "free", labeller = labeller(`Depth (cm)` = New
  geom_line(aes(x = `Time (days)`, y = `.pred`)) +
  scale_y_log10() +
  theme_bw() +
  theme(strip.text = element_text(size = 12), axis.text=element_text(size=12),
        axis.title=element_text(size=14),
        axis.title.y = element_text(size=14)) +
  xlab("Incubation time (d)") + ylab(bquote(~CO[2]~ 'emission rate ('*mu~'g'~ h^-1~g^-1*')'))
pl
```




```
ggsave("figures/CO2 emission.png", plot= pl, width = 7, height = 5, dpi = 600)
```

8. What is the temperature sensitivity across cranberry soil layers?

8.1 Fit of Arrhenius equation

The Arrhenius equation has been used to describe temperature sensitivity to CO2 emission. The Arrhenius equation was computed as follows:

$$k = Ae^{\frac{-Ea}{RT}}$$

$$log\left(k\right) = log\left(Ae^{\frac{-Ea}{RT}}\right)$$

$$log\left(k\right) = log\left(A\right) + log\left(e^{\frac{-Ea}{RT}}\right)$$

$$log\left(k\right) = log\left(A\right) - \frac{1}{T} \times \left(\frac{Ea}{R}\right)$$

Where A is the pre-exponential factor and Ea is activation energy assumed to be independent of temperature, R is the universal gas constant and T is absolute temperature (Kelvin)

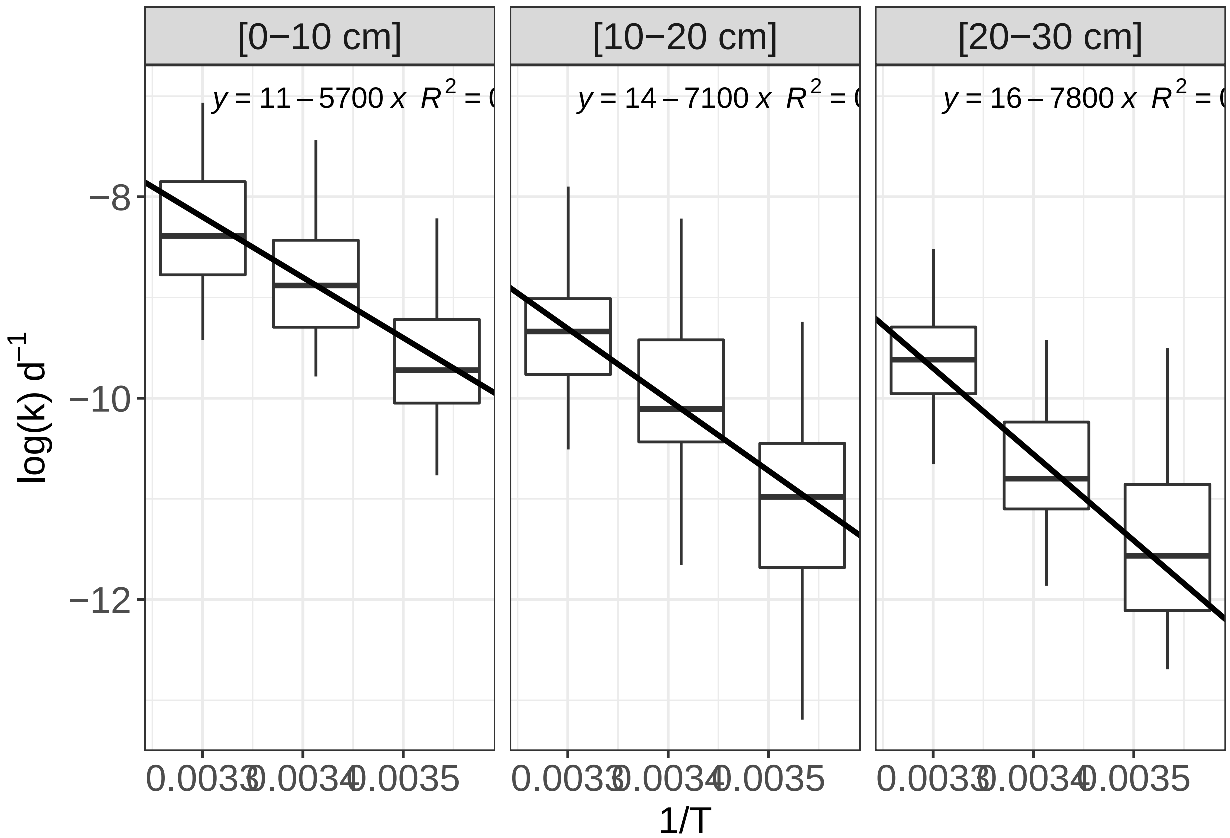
```
models_co2 <- data_co2 %>%
  group_by(`Depth (cm)`) %>%
  summarise(linmod = list(lm(lnK ~ `1/T`)))
models_co2
```

```
## # A tibble: 3 x 2
##   `Depth (cm)` linmod
##          <dbl> <list>
## 1           10 <lm>
## 2           20 <lm>
## 3           30 <lm>
```

```
linmod_coef <- list()
for (i in seq_along(models_co2$linmod)) linmod_coef[[i]] <- models_co2$linmod[[i]]$coefficients
linmod_coef <- do.call(rbind.data.frame, linmod_coef)
names(linmod_coef) <- c("Intercept", "Slope")
linmod_coef <- bind_cols(unique(data_co2["Depth (cm)"]), linmod_coef)
linmod_coef
```

```
## # A tibble: 3 x 3
##   `Depth (cm)` Intercept  Slope
##          <dbl>     <dbl>  <dbl>
## 1           10      11.6 -6002.
## 2           20      14.0 -7052.
## 3           30      18.5 -8558.
```

```
options(repr.plot.width = 12, repr.plot.height = 6)
plot_co2 <- data_co2_clean %>%
  ggplot(aes(x = `1/T`, y = lnK)) +
  facet_grid(~`Depth (cm)`, labeller = labeller(`Depth (cm)` = New.labs_b)) +
  geom_boxplot(aes(group = factor(`1/T`)), outlier.shape = NA) +
  stat_regline_equation(aes(label =  paste(..eq.label.., ..rr.label.., sep = "~~")), label.x = 0.00331,
  geom_abline(data = linmod_coef, aes(intercept = Intercept, slope = Slope), lwd = 1) +
  labs(y = bquote("log(k)" ~d^-1)) +
  theme_bw() +
  theme(strip.text = element_text(size = 14), axis.text=element_text(size=14),
        axis.title=element_text(size=14))
plot_co2
```




```
ggsave("figures/Arrhénius équation.png", plot = plot_co2, width = 8, height = 4, dpi = 600)# export plo
```

8.2 Activation Energy computation

```
Activation_energy <- tibble(
  Soil_layers = c("10", "20", "30"),
  intercept = NA,
  slope = NA,
  adj_r_sq = NA
)
```

```
lm_arrhenius <- for (i in 1:nrow(Activation_energy)) {

  lm_Activation_energy <- data_co2_clean %>%
    filter(`Depth (cm)` == Activation_energy$Soil_layers[i]) %>%
    lm(lnK ~ `1/T`, data = .)

  # intercept
  Activation_energy$intercept[i] <- coef(lm_Activation_energy)[1]

  # Slope
  Activation_energy$slope[i] <- coef(lm_Activation_energy)[2]

  # statistics
  Activation_energy$adj_r_sq[i] <- summary(lm_Activation_energy)$adj.r.squared
}
```

```r
R = 8.3144621 / 1000 # Gas constant Kj/mol/K
Activation_energy <-  Activation_energy %>%
  mutate(Ea = -slope * R) %>%
  select(Soil_layers, adj_r_sq, Ea)
Activation_energy
```

```
## # A tibble: 3 x 3
##   Soil_layers adj_r_sq    Ea
##   <chr>          <dbl> <dbl>
## 1 10             0.477  47.5
## 2 20             0.402  58.6
## 3 30             0.507  64.9
```

8.3 Computing K median in order to compute Q10 value accross soil depth

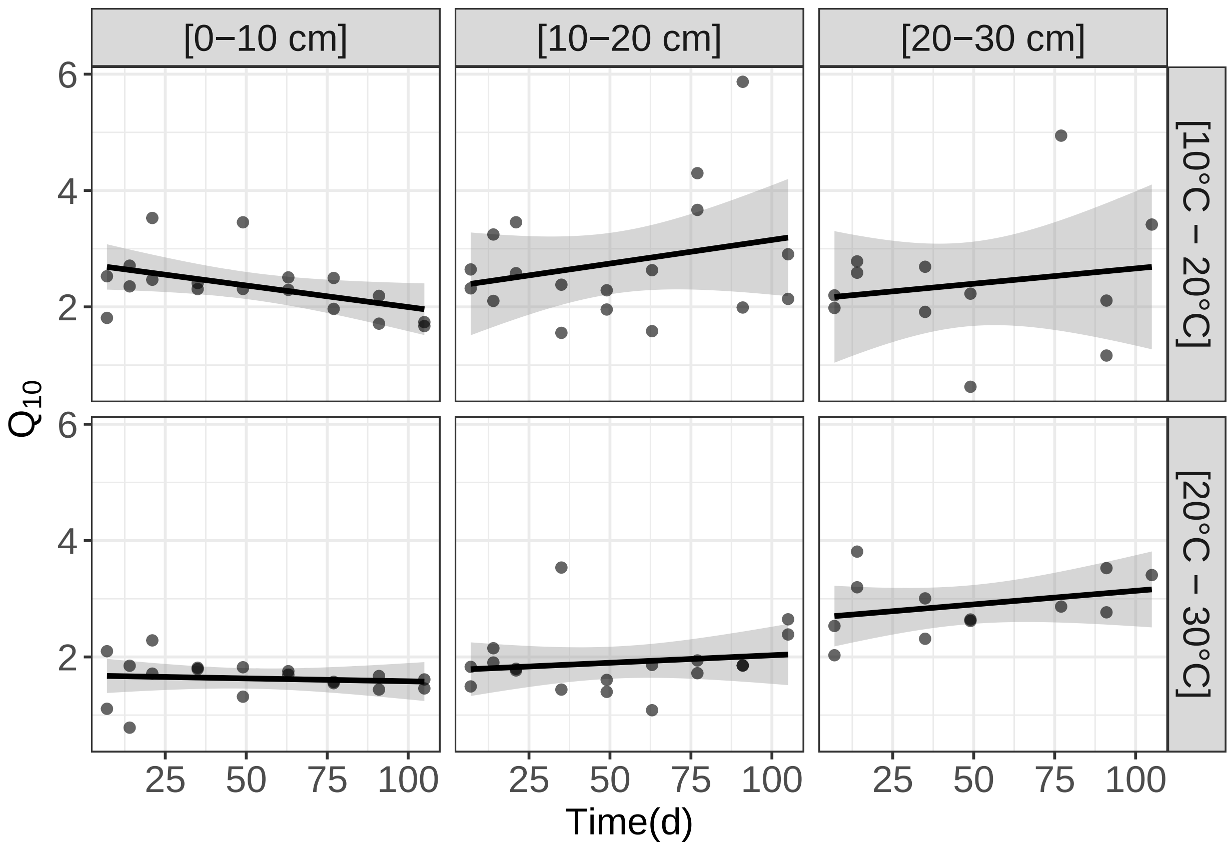
```r
K_median <- aggregate(K ~ `Sites` + `Time (days)` + `Depth (cm)` + `Temperature (°C)`, data = data_co2_c

K_median <- K_median %>%
  pivot_wider(names_from = `Temperature (°C)`, values_from = K)


K_median$Q_20_10 <- K_median$`20` / K_median$`10`
K_median$Q_30_20 <- K_median$`30` / K_median$`20`

K_median <- K_median %>%
  na.omit(K_median)
data_Q10 <- gather(data = K_median, key = `Temperature range`, value = Q10, c(`Q_20_10`, `Q_30_20`),
                   factor_key=TRUE)

stat_Q10 <- data_Q10 |>
  group_by(`Depth (cm)`) |>
  get_summary_stats(Q10)
stat_Q10
```

```
## # A tibble: 3 x 14
##   `Depth (cm)` variable     n   min   max median    q1    q3   iqr   mad  mean
##          <dbl> <chr>    <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1           10 Q10         36 0.785  3.53   1.82  1.67  2.32 0.647 0.542  1.99
## 2           20 Q10         36 1.08   5.87   2.05  1.79  2.63 0.845 0.579  2.33
## 3           30 Q10         24 0.627  4.94   2.63  2.17  3.05 0.879 0.709  2.64
## # ... with 3 more variables: sd <dbl>, se <dbl>, ci <dbl>
```

```r
New.labs_c <- c("[10°C - 20°C]", "[20°C - 30°C]") # Change labels
names(New.labs_c) <- c("Q_20_10", "Q_30_20")

options(repr.plot.width = 8, repr.plot.height = 4)
data_Q10 |>
  mutate(`Layers` = as.character(`Depth (cm)`)) |>
  ggplot(aes(x = `Time (days)`, y = `Q10`)) +
  facet_grid(`Temperature range`~`Depth (cm)`,
             labeller = labeller(`Depth (cm)` = New.labs_b,
                                 `Temperature range` = New.labs_c)) +
  geom_smooth(method = "lm", se = TRUE, color = "Black") +
  geom_point(size = 1.5, alpha = 0.6) +
```

```
  labs(x = "Time(d)", y = bquote(Q[10])) +
  theme_bw() +
  theme(strip.text = element_text(size = 14), axis.text=element_text(size=14),
        axis.title=element_text(size=14))
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
ggsave("figures/Variation of Q10 across layers.png", width = 8, height = 4, dpi = 600)# export plot hig
```

```
## `geom_smooth()` using formula 'y ~ x'
```

9. Soil description

9.1 Soil layers properties

Import data

```
data_carbon_credit <- read_csv2('data/data_carbon_credit.csv')
```

```
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
```

```
## Rows: 24 Columns: 15
## -- Column specification --------------------------------------------------------
## Delimiter: ";"
## chr  (3): Location, Layer (cm), 0_30_ID
## dbl (12): Sample, Site age, Repetition, Bulk density (kg m-3), pHCaCl2, Sand...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
data_carbon_credit <- data_carbon_credit |>
  mutate(`C:N ratio` = `Carbone (%)` / `Nitrogen (%)`)
```

Some calculations

```r
mean_sd_CoverN <- data_carbon_credit |>
  group_by(`Layer (cm)`) |>
  summarize(mean_C_over_N = mean(`C:N ratio`, na.rm = TRUE),
                        se_C_over_N = sd(`C:N ratio`, na.rm = TRUE)/sqrt(length(!is.na(`C:N ratio`))))
mean_sd_CoverN
```

```
## # A tibble: 3 x 3
##   `Layer (cm)` mean_C_over_N se_C_over_N
##   <chr>                <dbl>       <dbl>
## 1 [0-10]               20.1         1.05
## 2 [10-20]              16.0         1.91
## 3 [20-30]               9.02        1.96
```

```r
data_carbon_credit |> get_summary_stats(`C stock (kg m-3)`)
```

```
## # A tibble: 1 x 13
##   variable        n   min   max median    q1    q3   iqr   mad  mean    sd    se
##   <chr>       <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 C stock (k~    24  1.67  30.9   12.5  6.57  16.7  10.1  7.78  12.1  7.05  1.44
## # ... with 1 more variable: ci <dbl>
```

```r
data_carbon_credit |>
  group_by(`Layer (cm)`) |>
get_summary_stats(`C stock (kg m-3)`)
```

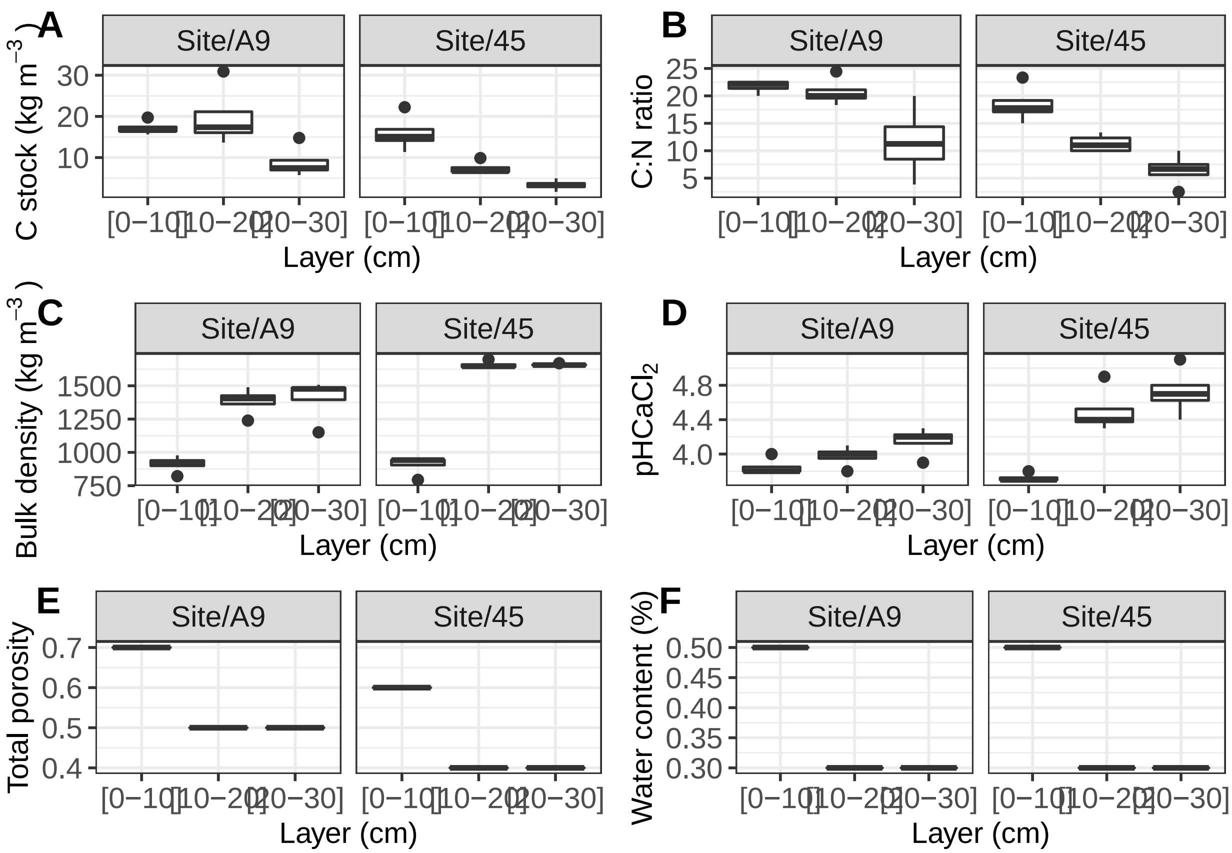
```
## # A tibble: 3 x 14
##   `Layer (cm)` variable        n   min   max median    q1    q3   iqr   mad  mean
##   <chr>        <chr>       <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 [0-10]       C stock (~      8 11.4   22.2   16.2 15.1  17.4   2.35  1.60 16.6
## 2 [10-20]      C stock (~      8  6.52  30.9   11.8  6.74 17.1  10.4   7.62 13.6
## 3 [20-30]      C stock (~      8  1.67  14.8    5.36  3.31  7.42  4.11  3.04  6.09
## # ... with 3 more variables: sd <dbl>, se <dbl>, ci <dbl>
```

```r
data_carbon_credit |>
  group_by(`Layer (cm)`) |>
get_summary_stats(`C:N ratio`)
```

```
## # A tibble: 3 x 14
##   `Layer (cm)` variable      n   min   max median    q1    q3   iqr   mad  mean
##   <chr>        <chr>     <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 [0-10]       C:N ratio     8  15    23.3   20.9  17.8  22.5  4.72  2.98 20.1
## 2 [10-20]      C:N ratio     8  10    24.4   15.8  11.5  20    8.5   6.18 16.0
## 3 [20-30]      C:N ratio     8   2.5  20      8.33  5.96 10.6  4.66  4.32  9.02
## # ... with 3 more variables: sd <dbl>, se <dbl>, ci <dbl>
```

```r
plot_desc <- function(y, ylab){
  New.labs_c <-  c("Site/A9", "Site/45") # Change labels
names(New.labs_c) <- c("Belanger/ A9", "Fortier/ 45")
  ggplot(data_carbon_credit, aes(`Layer (cm)`, y)) +
    geom_boxplot() +
facet_grid( . ~ `Location`, scales = "free", labeller = labeller(`Location` = New.labs_c)) +
    theme_bw() +
```

```
theme(strip.text = element_text(size = 11), axis.text=element_text(size=11),
        axis.title=element_text(size=11)) +
    labs(y = ylab)
  }

plot1 <- plot_desc(data_carbon_credit$`C stock (kg m-3)`, bquote("C stock (kg" ~m^-3~")"))
plot2 <- plot_desc(data_carbon_credit$`C:N ratio`, "C:N ratio")
plot3 <- plot_desc(data_carbon_credit$`Bulk density (kg m-3)`, # m-3
                   "Bulk density (kg"~m^-3~")")
plot4 <- plot_desc(data_carbon_credit$pHCaCl2, bquote(pHCaCl[2]))
plot5 <- plot_desc(data_carbon_credit$`Total porosity`, "Total porosity")
plot6 <- plot_desc(data_carbon_credit$`Water content (%)`, "Water content (%)")

options(repr.plot.width = 8, repr.plot.height = 6)
figure <- ggarrange(plot1, plot2, plot3, plot4, plot5, plot6,
                    labels = c("A", "B", "C", "D", "E", "F"), label.x = 0.05, label.y = 1.01,
                    ncol = 2, nrow = 3)
figure
```




```
ggsave("figures/Soil description.png", width = 8, height = 5, dpi = 600)# export plot high resolution
```

9.2 C:N ratio in alternate sublayers of sand and organic matter

Data loading

```
Carbon_credit <- read_csv2('data/data_carbon_sublayer.csv')
```

```
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
```

```
## Rows: 23 Columns: 20
## -- Column specification --------------------------------------------------------
## Delimiter: ";"
## chr  (6): Projet, Site, Horizon, Layers, Soil texture, Munsell_color
## dbl (14): Depht (cm), Thickness(cm), Bulk density(kg m-3), Site_age, Weigh_s...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
Carbon_credit
```

```
## # A tibble: 23 x 20
##    Projet   Site    Horizon `Depht (cm)` `Thickness(cm)` Layers `Bulk density(~`
##    <chr>    <chr>   <chr>          <dbl>           <dbl> <chr>             <dbl>
##  1 Pedology Belang~ H1               1.8             1.8 [0-1.~             913.
##  2 Pedology Belang~ H2               2.2             0.4 [1.8-~             913.
##  3 Pedology Belang~ H3               3.2             1   [2.2-~             913.
##  4 Pedology Belang~ H4               3.6             0.4 [3.2-~             913.
##  5 Pedology Belang~ H5               5.1             1.5 [3.6-~             913.
##  6 Pedology Belang~ H6               5.8             0.7 [5.1-~             913.
##  7 Pedology Belang~ H7               9.5             3.7 [5.8-~             913.
##  8 Pedology Belang~ H8              12               2   [9.5-~            1384.
##  9 Pedology Belang~ H9              12.5             0.5 [12-1~            1384.
## 10 Pedology Belang~ H10             19.2             6.7 [12.5~            1384.
## # ... with 13 more rows, and 13 more variables: `Soil texture` <chr>,
## #   Site_age <dbl>, Munsell_color <chr>, Weigh_superior_2MM <dbl>,
## #   `Weigh _0_2MM` <dbl>, Repetition <dbl>, pHCaCl2 <dbl>, CTRL_C_pourc <dbl>,
## #   CTRL_S_pourc <dbl>, CTRL_N_pourc <dbl>, C_pourc <dbl>, S_pourc <dbl>,
## #   N_pourc <dbl>
```

C:N ratio computation

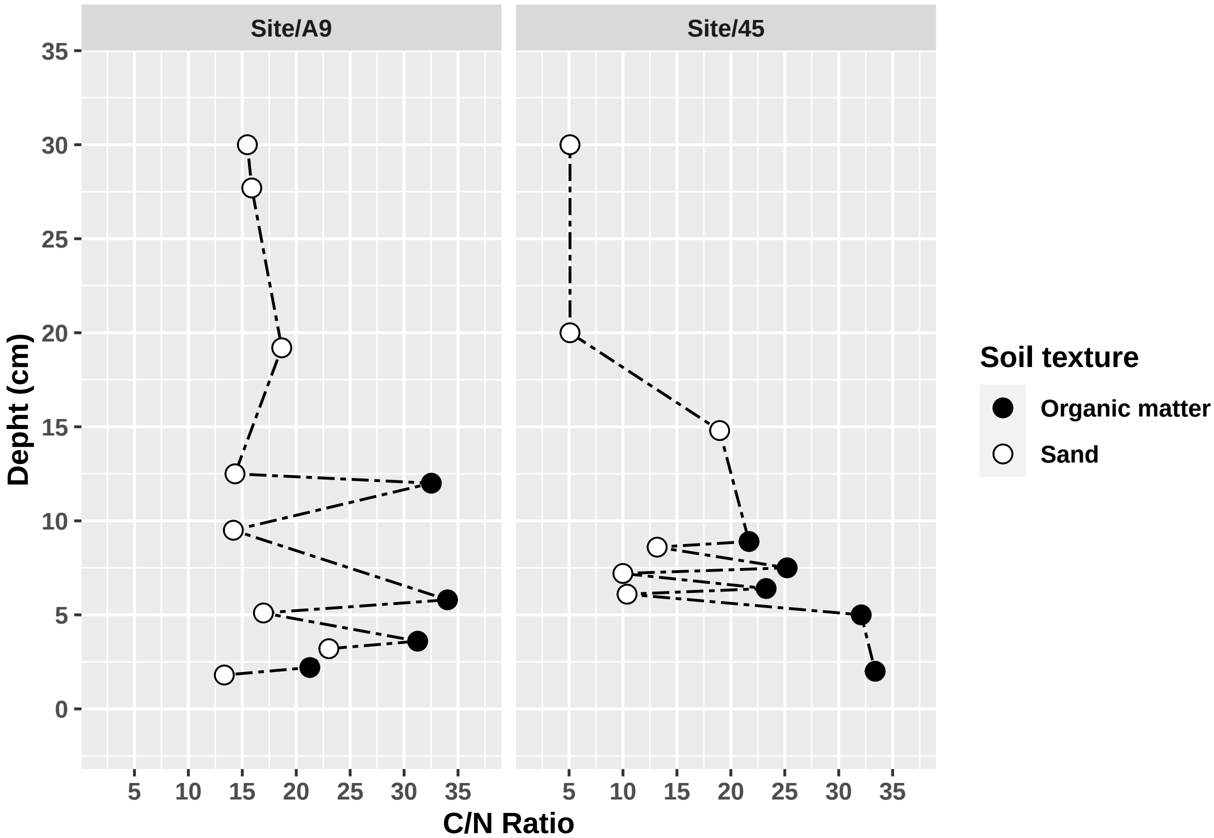
```
Carbon_credit <- Carbon_credit %>%
mutate(`C/N` = C_pourc/N_pourc)
```

Generating the plots

```
options(repr.plot.width=8, repr.plot.height=4)
New.labs_d <-  c("Site/A9", "Site/45") # Change labels
names(New.labs_d) <- c("Belanger/A9", "Fortier/45")

ggplot(data=Carbon_credit, aes(x= `Depht (cm)`, y= `C/N`)) +
  facet_grid(.~Site, labeller = labeller(`Site` = New.labs_d)) +
  geom_line(linetype = "twodash") +
  geom_point(aes(shape = `Soil texture`, fill = `Soil texture`), size = 3) +
  scale_shape_manual(values=c(21, 21))+
  scale_fill_manual(values = c("#000000", "#FFFFFF")) +
  scale_y_continuous(breaks = 5*0:1000,
                     expand = expand_scale(add = 5)) +
  scale_x_continuous(breaks = 5*0:1000,
                     expand = expand_scale(add = 5)) +
  theme(strip.text = element_text(face = "bold"), axis.text=element_text(face = "bold"),
        axis.title=element_text(face = "bold") , legend.title= element_text(face = "bold"),
        legend.text = element_text(face = "bold")) +
  labs(y= "C/N Ratio") +
```

```
  coord_flip()
```

```
## Warning: `expand_scale()` is deprecated; use `expansion()` instead.

## Warning: `expand_scale()` is deprecated; use `expansion()` instead.
```




```
ggsave("figures/(C_over_N).png", width = 8, height = 4, dpi = 800)
```