

Spring 2023 data analysis

import data file

```
abs_raw <- read_excel("2023-03-8 third trial 72hr.xlsx",
                      range = "B36:CU901")
colnames(abs_raw)[2] <- "temp"

lum_raw <- read_excel("2023-03-8 third trial 72hr.xlsx",
                      range = "B905:CU1770")
colnames(lum_raw)[2] <- "temp"
```

Declare well

```
# separate well by strain
MK62 = c("B2", "B3", "B4", "B5", "B6", "B7", "B8", "B9", "B10")
MK63 = c("C3", "C4", "C5", "C6", "C7", "C8", "C9", "C10")
MK64 = c("D2", "D4", "D6", "D8", "D9", "D10")
MK65 = c("E2", "E3", "E4", "E5", "E6", "E7", "E8", "E9", "E10")
MK66 = c("F2", "F3", "F4", "F5", "F6", "F7", "F8", "F9", "F10")
MK67 = c("G2", "G3", "G4", "G5", "G7", "G8", "G9", "G10")

# well that contains no bacteria but get contaminate
contam = c("A4", "A5", "F1", "H1", "B1")

# well that have condensation at the start
conden = c("A11", "H12", "G12", "D1", "C2", "D3", "D5", "D7", "G6", "F12", "H4")

# well that contains no bacteria
control = c("A1", "A2", "A3", "A6", "A7", "A8", "A9", "A10", "A12", "B12", "C1", "C12", "D12", "E1", "E2")
```

normalized time/pivot table

```
# normalized time into minutes
abs_with_time <- abs_raw %>%
  mutate(Time = Time[1] %% Time, # use time interval to normalize time
        Time = as.numeric(Time, unit = "hours"))

# pivot absorbance table
abs_df <- abs_with_time %>%
  pivot_longer(
    cols = !Time & !temp,
    names_to = "well",
    values_to = "absorbance"
  )

# normalized time into minutes
lum_with_time <- lum_raw %>%
  mutate(Time = Time[1] %% Time,
```

```

Time = as.numeric(Time, unit = "hours"))

# pivot luminescence table
lum_df <- lum_with_time %>%
  pivot_longer(
    cols = !Time & !temp,
    names_to = "well",
    values_to = "luminescence"
  )

```

Combine abs and lum table and declare strain/species/promoter group

```

# combine abs and lum dataframe
abs_lum <- cbind(abs_df, lum_df[4])

# assign strain/species/promoter group to each well
group_abs_lum <- abs_lum %>%
  mutate(# declare strain
        strain = case_when(
          well %in% MK62 ~ "MK62",
          well %in% MK63 ~ "MK63",
          well %in% MK64 ~ "MK64",
          well %in% MK65 ~ "MK65",
          well %in% MK66 ~ "MK66",
          well %in% MK67 ~ "MK67",
          well %in% contam ~ "contaminate",
          well %in% condensated ~ "condensated",
          well %in% control ~ "control"),
        # declare species
        species = case_when(
          well %in% MK62 ~ "E. coli",
          well %in% MK63 ~ "E. coli",
          well %in% MK64 ~ "E. coli",
          well %in% MK65 ~ "E. clocae",
          well %in% MK66 ~ "E. clocae",
          well %in% MK67 ~ "E. clocae",
          well %in% contam ~ "contaminate",
          well %in% condensated ~ "condensated",
          well %in% control ~ "control"),
        # declare promoter
        promoter = case_when(
          well %in% MK62 ~ "no promoter",
          well %in% MK63 ~ "E. coli promoter",
          well %in% MK64 ~ "E. clocae promoter",
          well %in% MK65 ~ "no promoter",
          well %in% MK66 ~ "E. coli promoter",
          well %in% MK67 ~ "E. clocae promoter",
          well %in% contam ~ "contaminate",
          well %in% condensated ~ "condensated",
          well %in% control ~ "control")))

```

normalized absorbance and luminescence with background

```

# get background abs and lum at each time
envi_mean <- group_abs_lum %>%

```

```

filter(strain == "control") %>%
group_by(Time) %>%
summarize(mean_abs = mean(absorbance),
          mean_lum = mean(luminescence))
# combine background abs and lum data with experiment data
normalize_envi <- left_join(group_abs_lum, envi_mean, by = "Time")

# normalize abs and lum with background
normalize_abs_lum <- normalize_envi %>%
  mutate(absorbance = absorbance - mean_abs,
        luminescence = luminescence - mean_lum)

```

normalized luminescence with absorbance

```

# normalised the luminescence to the absorbance
normalized_lum <- normalize_abs_lum %>%
  mutate(normalized = luminescence/absorbance)

```

Choose only used well, don't use the well that has contamination potential or have condensation at the start

```

correct_well_data <- normalized_lum %>%
  filter(strain != "contaminate") %>%
  filter(strain != "condensated")

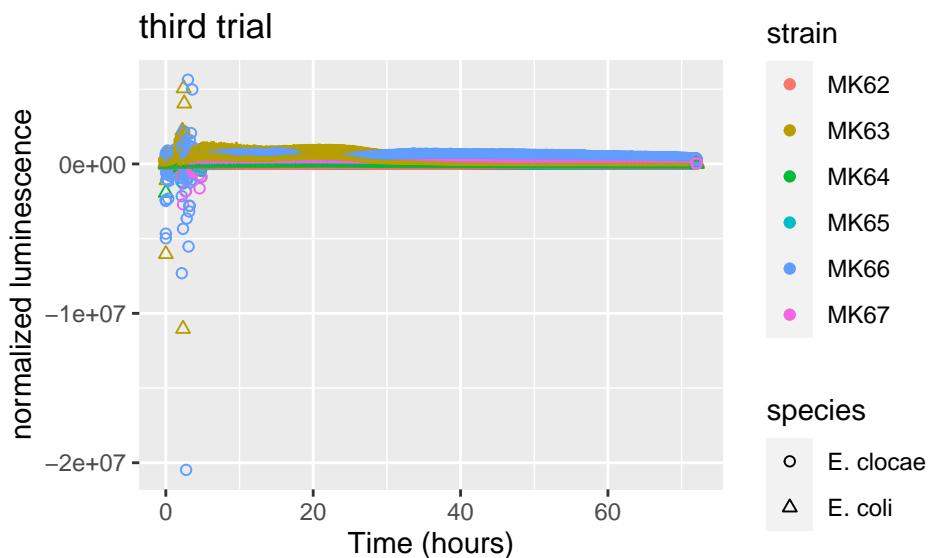
```

Plot normalized luminescence out

```

correct_well_data %>%
  filter(strain != "control") %>%
  ggplot() +
  geom_point(mapping = aes(x = Time, y = normalized, shape = species, color = strain)) +
  scale_shape_manual(values=c(1, 2, 3)) +
  xlab("Time (hours)") +
  ylab("normalized luminescence") +
  ggtitle("third trial")

```



Plot raw abs out

```

ggplot(data = correct_well_data) +
  geom_point(mapping = aes(x = Time, y = absorbance, color = strain, shape = promoter))+

```

```

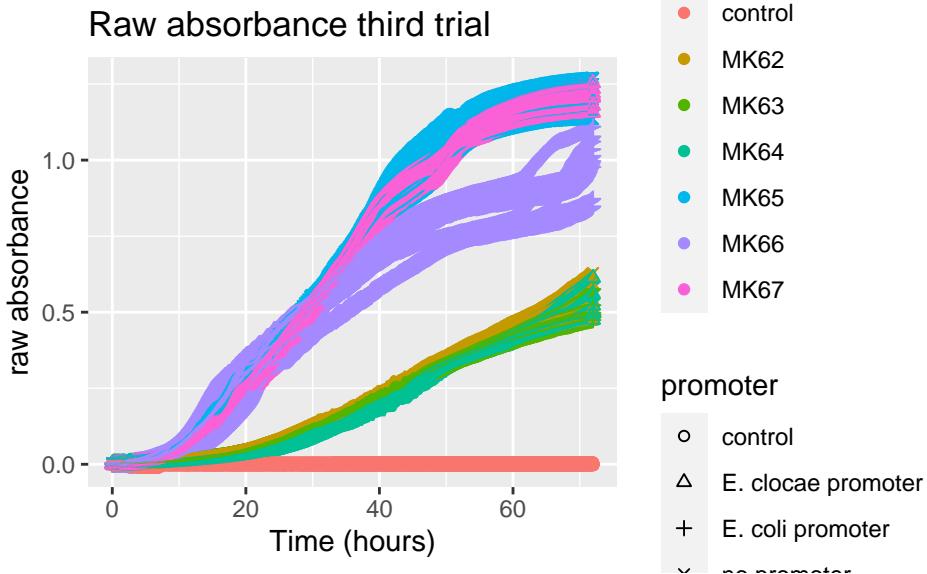
scale_shape_manual(values=c(1, 2, 3, 4))+  

xlab("Time (hours)") +  

ylab("raw absorbance") +  

ggtitle("Raw absorbance third trial")

```



Plot raw lum out

```

ggplot(data = correct_well_data) +  

  geom_point(mapping = aes(x = Time, y = luminescence, shape = species, color = strain)) +  

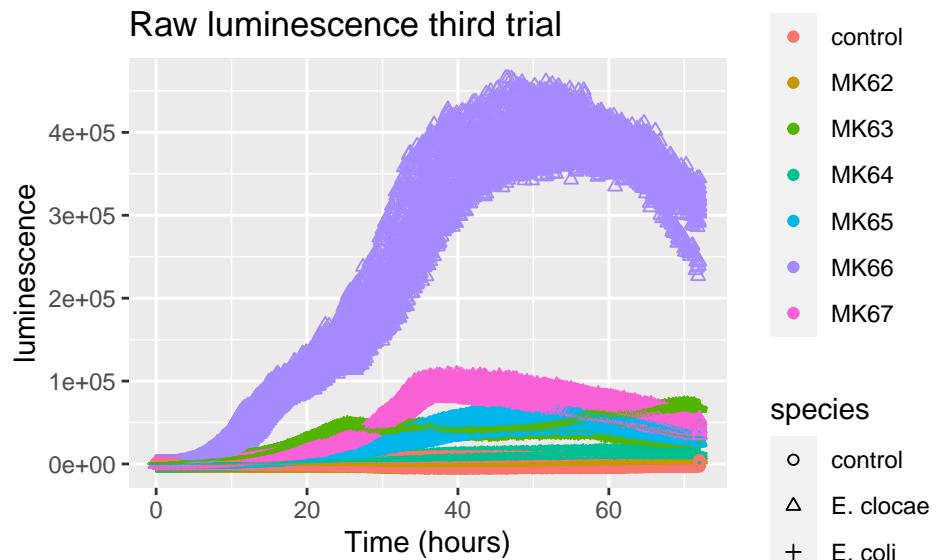
  scale_shape_manual(values=c(1, 2, 3))+  

  xlab("Time (hours)") +  

  ylab("luminescence") +  

  ggtitle("Raw luminescence third trial")

```



plot normalized luminescence log scale

```

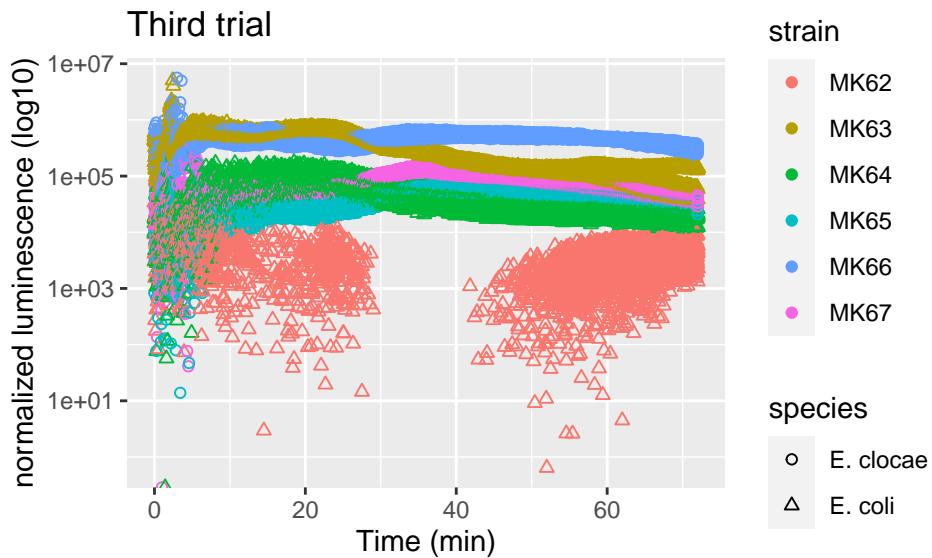
correct_well_data %>%
  filter(strain != "control") %>%

```

```

ggplot() +
  geom_point(mapping = aes(x = Time, y = normalized, shape = species, color = strain)) +
  scale_shape_manual(values=c(1, 2, 3))+
  scale_y_continuous(trans = 'log10') +
  xlab("Time (min)") +
  ylab("normalized luminescence (log10)") +
  ggtitle("Third trial")

```

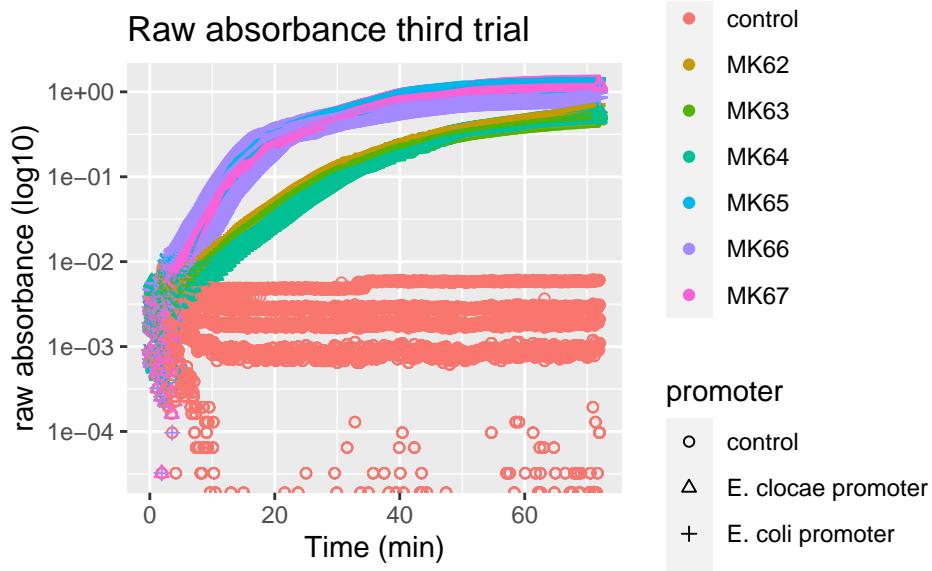


Plot raw abs log scale

```

ggplot(data = correct_well_data) +
  geom_point(mapping = aes(x = Time, y = absorbance, color = strain, shape = promoter))+ 
  scale_shape_manual(values=c(1, 2, 3, 4))+
  scale_y_continuous(trans = 'log10') +
  xlab("Time (min)") +
  ylab("raw absorbance (log10)") +
  ggtitle("Raw absorbance third trial")

```



Plot raw lum log scale

```
ggplot(data = correct_well_data) +  
  geom_point(mapping = aes(x = Time, y = luminescence, shape = species, color = strain)) +  
  scale_shape_manual(values=c(1, 2, 3))+  
  scale_y_continuous(trans = 'log10') +  
  xlab("Time (min)") +  
  ylab("luminescence (log10)") +  
  ggtitle("Raw luminescence third trial")
```

