Pseudomonas Epiphytic Growth and Virulence Analysis

2024 Summer

Contents

Load Libraries

```
pacman::p_load(ggplot2, readxl, RColorBrewer, paletteer, viridis, wesanderson, ggbeeswarm,
    dplyr, tidyverse, devtools, emmeans, cowplot, knitr, survival, here, tibble,
    survminer, lubridate, formatR, gridExtra, ggsurvfit, gtsummary, tidycmprsk, install = FALSE)
```

Load Data

```r
pseud_epi_growth_2024summer_R <- read_excel("data/pseud_epi_growth_2024summer_R.xlsx")

# Define the desired order of the treatments
desired_order <- c("194", "200", "204", "205", "215", "216", "220", "221", "227",
    "228", "B728a", "Cit7", "pisi", "Control")

# Convert the treatment variable to a factor with the specified order
pseud_epi_growth_2024summer_R$strain <- factor(pseud_epi_growth_2024summer_R$strain,
    levels = desired_order)

aphid_virulence_data <- read_csv("data/virulence_new_all.csv")
```

```
## Rows: 4973 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): date, treatment, replicate
## dbl (3): individual, censored, time
##
## 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
# Convert the treatment variable to a factor with the specified order
aphid_virulence_data$treatment <- factor(aphid_virulence_data$treatment, levels = desired_order)
```

Load colors

```r
# Define the desired order of the treatments
strains <- c("194", "200", "204", "205", "215", "216", "220", "221", "227", "228",
    "B728a", "Cit7", "pisi", "Control")

# Convert the strain variable to a factor with the specified order
pseud_epi_growth_2024summer_R$strain <- factor(pseud_epi_growth_2024summer_R$strain,
    levels = strains)


# Extract the default ggplot colors
default_colors <- (scales::hue_pal())(length(strains) - 1)

# Create a named vector for strain colors, ensuring 'Control' is black
strain_colors <- setNames(c(default_colors, "black"), strains)

# Load species colors
species_colors <- c(`P. fluorescens` = "dodgerblue2", `P. syringae` = "springgreen3",
    `P. paralactis` = "sienna4")
```

Goals

- Create Kaplan-Meier curve for Pseud. virulence data

- Use stats (Wilcox?) to determine statistical significance of each strain
- Compare virulence data with epiphitic growth ability

Virulence Analysis

Note: *In order to help me with this analysis, I am using the following sites - Survival Analysis in R and Hazard Ratio: Interpretation & Definition.*

Caclulcate survival probabilities for each strain and create dataframe

```r
# Fit the survival model
km_fit <- survfit(Surv(time, censored) ~ treatment, data = aphid_virulence_data)

# Extract survival probabilities at specific time points
time_points <- c(24, 48, 72)
km_summary <- summary(km_fit, times = time_points)

# Initialize empty lists to store the results
times_list <- list()
treatment_list <- list()
surv_prob_list <- list()

# Loop over each treatment group and extract survival probabilities at
# specified time points
for (i in 1:length(km_fit$strata)) {
    treatment_name <- names(km_fit$strata)[i]
    for (t in time_points) {
        idx <- which(km_summary$time == t & km_summary$strata == treatment_name)
        if (length(idx) > 0) {
            times_list <- c(times_list, t)
            treatment_list <- c(treatment_list, treatment_name)
            surv_prob_list <- c(surv_prob_list, km_summary$surv[idx])
        } else {
            times_list <- c(times_list, t)
            treatment_list <- c(treatment_list, treatment_name)
            surv_prob_list <- c(surv_prob_list, NA)
        }
    }
}

# Create the data frame
surv_probs <- data.frame(time = unlist(times_list), treatment = unlist(treatment_list),
    surv_prob = unlist(surv_prob_list))

# Replace 'treatment=' with an empty string
surv_probs$treatment <- gsub("treatment=", "", surv_probs$treatment)

# Print the data frame
print(surv_probs)
```

```
##    time treatment  surv_prob
```

```
## 1    24        194 0.77891156
## 2    48        194 0.18707483
## 3    72        194 0.12244898
## 4    24        200 0.53159851
## 5    48        200 0.05576208
## 6    72        200 0.02230483
## 7    24        204 0.54929577
## 8    48        204 0.08450704
## 9    72        204 0.05281690
## 10   24        205 0.56949153
## 11   48        205 0.11525424
## 12   72        205 0.07118644
## 13   24        215 0.79513889
## 14   48        215 0.44791667
## 15   72        215 0.22222222
## 16   24        216 0.56250000
## 17   48        216 0.27430556
## 18   72        216 0.15625000
## 19   24        220 0.35135135
## 20   48        220 0.19256757
## 21   72        220 0.17229730
## 22   24        221 0.65543071
## 23   48        221 0.21722846
## 24   72        221 0.09363296
## 25   24        227 0.40000000
## 26   48        227 0.22033898
## 27   72        227 0.16949153
## 28   24        228 0.65671642
## 29   48        228 0.15298507
## 30   72        228 0.10447761
## 31   24      B728a 0.68041237
## 32   48      B728a 0.40549828
## 33   72      B728a 0.10996564
## 34   24       Cit7 0.75609756
## 35   48       Cit7 0.37630662
## 36   72       Cit7 0.17073171
## 37   24       pisi 0.88250653
## 38   48       pisi 0.79634465
## 39   72       pisi 0.74151436
## 40   24    Control 0.94777397
## 41   48    Control 0.87243151
## 42   72    Control 0.81678082
```

```r
# subset data by time
surv_probs_72 <- surv_probs %>%
    filter(time == 72)

# Convert the treatment variable to a factor with the specified order
surv_probs_72$treatment <- factor(surv_probs_72$treatment, levels = desired_order)

# If you need to save it to a file, you can use the following command
# write.csv(surv_probs, 'survival_probabilities.csv', row.names = FALSE)
```

Post-hoc test for survival

```
# Perform pairwise log-rank tests between treatment groups and the control
# group
pairwise_tests <- pairwise_survdiff(Surv(time, censored) ~ treatment, data = aphid_virulence_data)

# Print the pairwise tests
print(pairwise_tests)
```

```
##
##  Pairwise comparisons using Log-Rank test
##
## data:  aphid_virulence_data and treatment
##
##            194      200      204      205      215      216      220      221      227
## 200     2.3e-12  -        -        -        -        -        -        -        -
## 204     1.6e-08  0.26489  -        -        -        -        -        -        -
## 205     3.7e-06  0.04009  0.38289  -        -        -        -        -        -
## 215     4.6e-05  < 2e-16  < 2e-16  7.8e-16  -        -        -        -        -
## 216     0.20589  3.0e-06  0.00054  0.00915  1.3e-05  -        -        -        -
## 220     1.2e-05  0.34987  0.95899  0.61067  7.5e-10  0.04787  -        -        -
## 221     0.06597  1.7e-06  0.00063  0.01560  3.0e-08  0.57084  0.05899  -        -
## 227     0.00048  0.05275  0.34910  0.75895  1.7e-08  0.14997  0.65606  0.23547  -
## 228     0.02457  2.3e-05  0.00265  0.03730  5.6e-09  0.52684  0.02668  0.75895  0.15756
## B728a   0.30241  3.8e-15  2.0e-10  1.5e-07  0.00193  0.15635  0.00089  0.00913  0.00558
## Cit7    0.97833  6.8e-10  1.0e-06  8.4e-05  0.00044  0.38289  0.00217  0.11439  0.01459
## pisi    < 2e-16  < 2e-16  < 2e-16  < 2e-16  < 2e-16  < 2e-16  < 2e-16  < 2e-16  < 2e-16
## Control < 2e-16  < 2e-16  < 2e-16  < 2e-16  < 2e-16  < 2e-16  < 2e-16  < 2e-16  < 2e-16
##            228      B728a    Cit7     pisi
## 200     -        -        -        -
## 204     -        -        -        -
## 205     -        -        -        -
## 215     -        -        -        -
## 216     -        -        -        -
## 220     -        -        -        -
## 221     -        -        -        -
## 227     -        -        -        -
## 228     -        -        -        -
## B728a   0.00265  -        -        -
## Cit7    0.05646  0.46912  -        -
## pisi    < 2e-16  < 2e-16  < 2e-16  -
## Control < 2e-16  < 2e-16  < 2e-16  0.00042
##
## P value adjustment method: BH
```

Plot Surival Dataframe at different times

```
surv_plot_72 <- ggplot(data = surv_probs_72, aes(x = treatment, y = (1 - surv_prob),
    color = treatment, fill = treatment)) + geom_col() + labs(title = "72 hours",
    x = "-80 Strain #", y = "Aphid Mortality at 72 Hours (%)") + theme_minimal() +
    theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 45,
```

```
        hjust = 1), legend.position = "none") + scale_fill_manual(values = strain_colors) +
    scale_color_manual(values = strain_colors)
```

```
surv_plot_72
```

72 hours
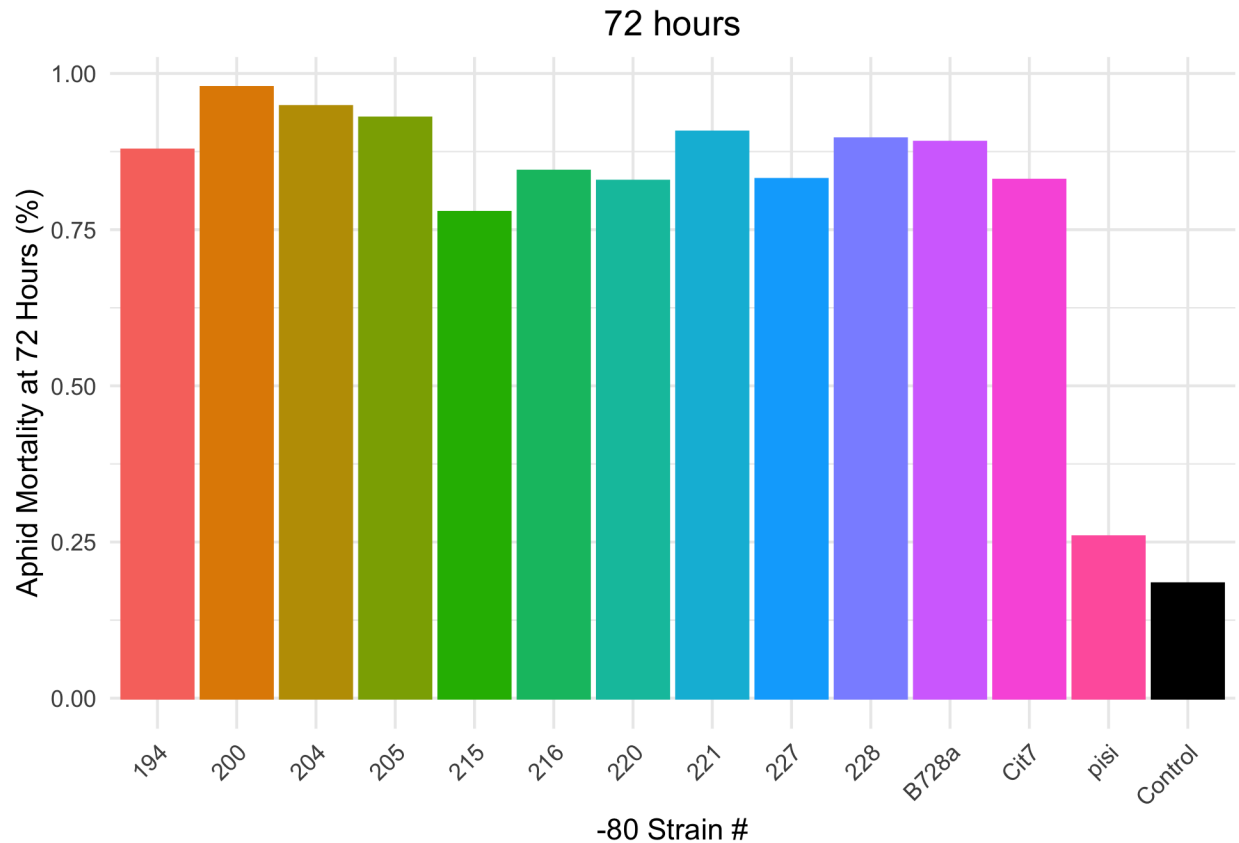
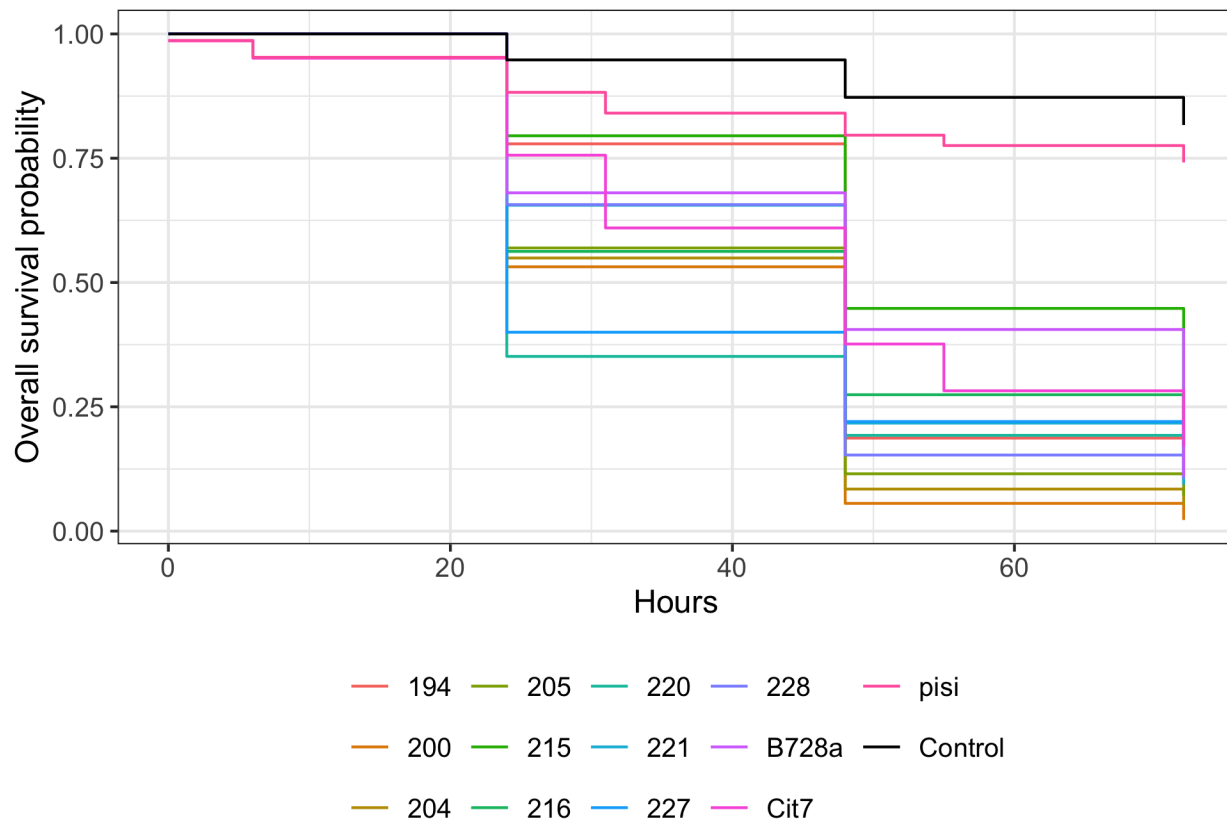**Make Kaplan-Meier Plot**

Cohort Survival Curve

```
survfit2(Surv(time, censored) ~ treatment, data = aphid_virulence_data) %>%
    ggsurvfit() + labs(x = "Hours", y = "Overall survival probability") + scale_fill_manual(values = st:
    scale_color_manual(values = strain_colors)
```

```
## Warning: No shared levels found between 'names(values)' of the manual scale and the
## data's fill values.
```

Epiphytic Growth Analysis

Calculate mean/variance epiphytic growth ability

```r
# Replace NA with a lower value or remove them for visualization Remove rows
# with NA in CFU_per_10_leafdiscs
epi_growth_clean <- pseud_epi_growth_2024summer_R %>%
    filter(!is.na(CFU_per_10_leafdiscs))

# Convert CFU_per_10_leafdiscs to numeric, handling scientific notation
epi_growth_clean$CFU_per_10_leafdiscs <- as.numeric(gsub("<", "", epi_growth_clean$CFU_per_10_leafdiscs
```

```
## Warning: NAs introduced by coercion
```

```r
# Subset data to remove unfinished strains and extract the letter part from the
# 'plant_rep' column
epi_growth_clean <- epi_growth_clean %>%
    mutate(plant_rep_letter = substr(plant_rep, 1, 1))

# Calculate the mean and SD CFU per strain for each plant rep letter and retain
# the specified columns
epi_growth_mean_per_rep <- epi_growth_clean %>%
```

```r
    group_by(strain, plant_rep_letter) %>%
    summarise(mean_CFU_per_10_leafdiscs = mean(CFU_per_10_leafdiscs, na.rm = TRUE),
        sd_CFU = sd(CFU_per_10_leafdiscs, na.rm = TRUE), sample = first(sample),
        person = first(person), block = first(block), species = first(species), plant_rep = first(plant
        .groups = "drop")

# Calculate the mean and SD CFU per strain for each plant rep letter and retain
# the specified columns
epi_growth_mean_per_strain <- epi_growth_clean %>%
    group_by(strain) %>%
    summarise(mean_CFU_per_10_leafdiscs = mean(CFU_per_10_leafdiscs, na.rm = TRUE),
        sd_CFU = sd(CFU_per_10_leafdiscs, na.rm = TRUE), sample = first(sample),
        person = first(person), block = first(block), species = first(species), plant_rep = first(plant
        .groups = "drop")

epi_growth_mean_per_strain
```

```
## # A tibble: 13 x 8
##    strain mean_CFU_per_10_leafdiscs sd_CFU sample person block species plant_rep
##    <fct>                      <dbl>  <dbl> <chr>  <chr>  <chr> <chr>   <chr>
##  1 194                      156472. 1.43e5 194-A1 Havi   2-1   P. par~ A1
##  2 200                      261214. 1.81e5 200-A1 Havi   2-6   P. flu~ A1
##  3 204                      186888. 1.89e5 204-A1 Havi   2-3   P. flu~ A1
##  4 205                      207750  2.40e5 205-A1 Sara   3     P. flu~ A1
##  5 215                         1000  0     215-A1 Sara   3     P. syr~ A1
##  6 216                         6788. 7.61e3 216-A1 Sara   3     P. flu~ A1
##  7 220                      172543. 1.89e5 220-A1 Sara   5     P. par~ A1
##  8 221                      259556. 1.93e5 221-A1 Havi   2-7   P. flu~ A1
##  9 227                       82872. 1.53e5 227-A1 Sara   5     P. flu~ A1
## 10 228                       22875  2.37e4 228-A1 Sara   5     P. flu~ A1
## 11 B728a                     71144. 1.36e5 B728a~ Sara   4     P. syr~ A1
## 12 Cit7                      61906. 1.17e5 Cit7-~ Havi   2-3   P. syr~ A1
## 13 pisi                      11607. 3.51e4 pisi-~ Havi   2-4   P. syr~ A1
```

Plot Epiphytic Growth

```r
epi_growth_strain <- ggplot(data = epi_growth_mean_per_rep, aes(x = strain, y = mean_CFU_per_10_leafdis
  geom_boxplot(outlier.shape = NA, alpha = 0.3) +
  geom_beeswarm(stroke = 0.5, size = 0.8, alpha = 0.8) +
  labs(title = "CFU per 10 Leaf Discs Across Samples",
      x = "-80 Strain #",
      y = "CFU per 10 Leaf Discs (Log10 Transformed)") +
  theme_minimal() +
  scale_y_log10() +   # Apply log transformation to y-axis
  theme(plot.title = element_text(hjust = 0.5),
      axis.text.x = element_text(angle = 45, hjust = 1), legend.position = "none")

epi_growth_strain
```

CFU per 10 Leaf Discs Across Samples
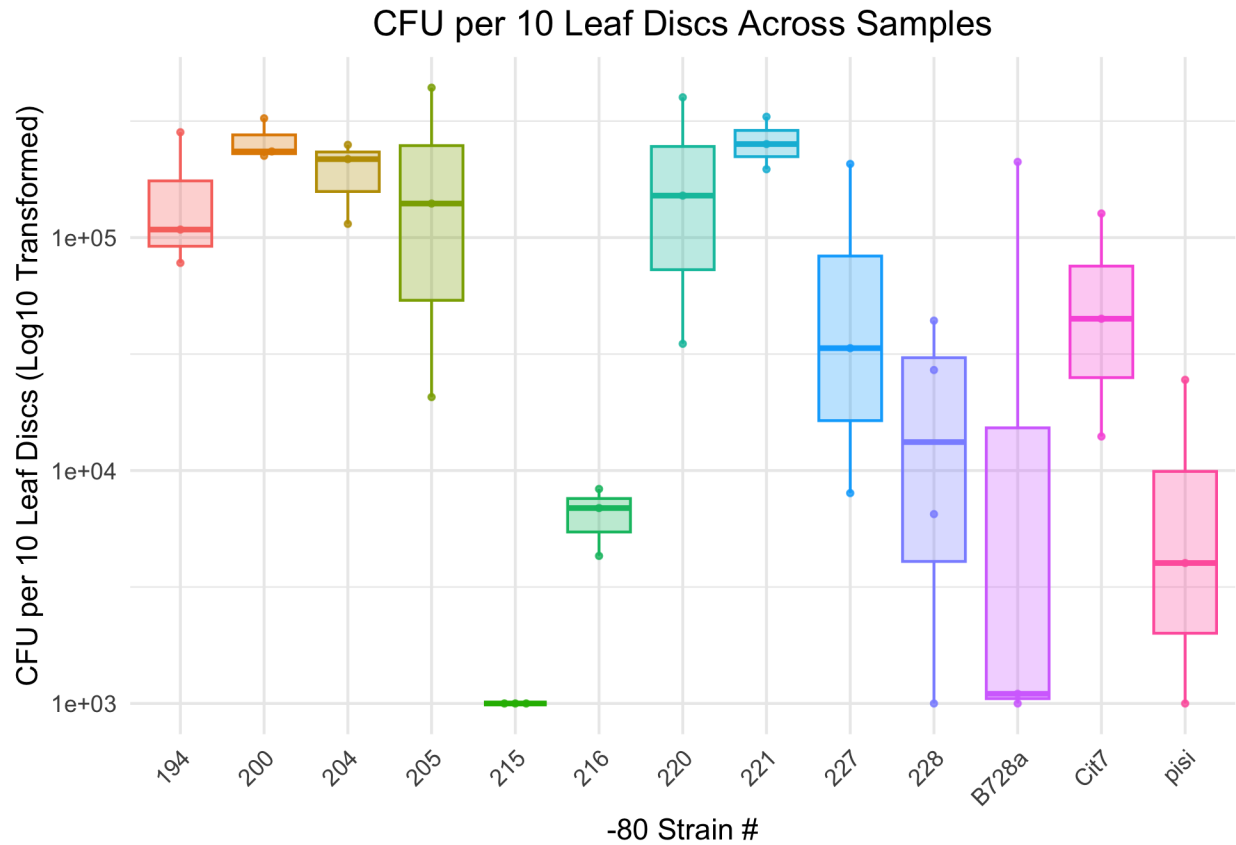
Epiphutic growth stats

```
epi_growth_mod = lm(mean_CFU_per_10_leafdiscs ~ strain, data = epi_growth_mean_per_rep)

emmeans(epi_growth_mod, pairwise ~ strain)
```

```
## $emmeans
##   strain emmean     SE df lower.CL upper.CL
##   194    156472 58153 27    37151   275793
##   200    261214 58153 27   141893   380535
##   204    193911 58153 27    74590   313232
##   205    200222 58153 27    80901   319543
##   215      1000 58153 27  -118321   120321
##   216      6511 58153 27  -112810   125832
##   220    195467 58153 27    76146   314788
##   221    259556 58153 27   140234   378877
##   227     82872 58153 27   -36449   202193
##   228     19625 50362 27   -83710   122960
##   B728a   71144 58153 27   -48177   190466
##   Cit7    61906 58153 27   -57416   181227
##   pisi     9839 58153 27  -109482   129160
##
## Confidence level used: 0.95
```

```
## 
## $contrasts
##  contrast    estimate    SE  df t.ratio p.value
##  194 - 200    -104742 82241 27  -1.274  0.9829
##  194 - 204     -37439 82241 27  -0.455  1.0000
##  194 - 205     -43750 82241 27  -0.532  1.0000
##  194 - 215     155472 82241 27   1.890  0.7869
##  194 - 216     149961 82241 27   1.823  0.8218
##  194 - 220     -38994 82241 27  -0.474  1.0000
##  194 - 221    -103083 82241 27  -1.253  0.9849
##  194 - 227      73600 82241 27   0.895  0.9992
##  194 - 228     136847 76930 27   1.779  0.8434
##  194 - B728a    85328 82241 27   1.038  0.9970
##  194 - Cit7     94567 82241 27   1.150  0.9926
##  194 - pisi    146633 82241 27   1.783  0.8415
##  200 - 204      67303 82241 27   0.818  0.9997
##  200 - 205      60992 82241 27   0.742  0.9999
##  200 - 215     260214 82241 27   3.164  0.1318
##  200 - 216     254703 82241 27   3.097  0.1503
##  200 - 220      65747 82241 27   0.799  0.9998
##  200 - 221       1658 82241 27   0.020  1.0000
##  200 - 227     178342 82241 27   2.169  0.6194
##  200 - 228     241589 76930 27   3.140  0.1381
##  200 - B728a   190069 82241 27   2.311  0.5282
##  200 - Cit7    199308 82241 27   2.423  0.4582
##  200 - pisi    251375 82241 27   3.057  0.1624
##  204 - 205      -6311 82241 27  -0.077  1.0000
##  204 - 215     192911 82241 27   2.346  0.5064
##  204 - 216     187400 82241 27   2.279  0.5489
##  204 - 220      -1556 82241 27  -0.019  1.0000
##  204 - 221     -65644 82241 27  -0.798  0.9998
##  204 - 227     111039 82241 27   1.350  0.9733
##  204 - 228     174286 76930 27   2.266  0.5573
##  204 - B728a   122767 82241 27   1.493  0.9456
##  204 - Cit7    132006 82241 27   1.605  0.9132
##  204 - pisi    184072 82241 27   2.238  0.5748
##  205 - 215     199222 82241 27   2.422  0.4589
##  205 - 216     193711 82241 27   2.355  0.5003
##  205 - 220       4756 82241 27   0.058  1.0000
##  205 - 221     -59333 82241 27  -0.721  0.9999
##  205 - 227     117350 82241 27   1.427  0.9601
##  205 - 228     180597 76930 27   2.348  0.5052
##  205 - B728a   129078 82241 27   1.569  0.9246
##  205 - Cit7    138317 82241 27   1.682  0.8853
##  205 - pisi    190383 82241 27   2.315  0.5258
##  215 - 216      -5511 82241 27  -0.067  1.0000
##  215 - 220    -194467 82241 27  -2.365  0.4945
##  215 - 221    -258556 82241 27  -3.144  0.1372
##  215 - 227     -81872 82241 27  -0.996  0.9979
##  215 - 228     -18625 76930 27  -0.242  1.0000
##  215 - B728a   -70144 82241 27  -0.853  0.9995
##  215 - Cit7    -60906 82241 27  -0.741  0.9999
##  215 - pisi     -8839 82241 27  -0.107  1.0000
##  216 - 220    -188956 82241 27  -2.298  0.5368
```

```
##  216 - 221     -253044 82241 27  -3.077  0.1563
##  216 - 227      -76361 82241 27  -0.928  0.9989
##  216 - 228      -13114 76930 27  -0.170  1.0000
##  216 - B728a    -64633 82241 27  -0.786  0.9998
##  216 - Cit7     -55394 82241 27  -0.674  1.0000
##  216 - pisi      -3328 82241 27  -0.040  1.0000
##  220 - 221      -64089 82241 27  -0.779  0.9998
##  220 - 227      112594 82241 27   1.369  0.9704
##  220 - 228      175842 76930 27   2.286  0.5443
##  220 - B728a    124322 82241 27   1.512  0.9409
##  220 - Cit7     133561 82241 27   1.624  0.9068
##  220 - pisi     185628 82241 27   2.257  0.5626
##  221 - 227      176683 82241 27   2.148  0.6323
##  221 - 228      239931 76930 27   3.119  0.1441
##  221 - B728a    188411 82241 27   2.291  0.5410
##  221 - Cit7     197650 82241 27   2.403  0.4705
##  221 - pisi     249717 82241 27   3.036  0.1688
##  227 - 228       63247 76930 27   0.822  0.9997
##  227 - B728a     11728 82241 27   0.143  1.0000
##  227 - Cit7      20967 82241 27   0.255  1.0000
##  227 - pisi      73033 82241 27   0.888  0.9993
##  228 - B728a    -51519 76930 27  -0.670  1.0000
##  228 - Cit7     -42281 76930 27  -0.550  1.0000
##  228 - pisi       9786 76930 27   0.127  1.0000
##  B728a - Cit7     9239 82241 27   0.112  1.0000
##  B728a - pisi    61306 82241 27   0.745  0.9999
##  Cit7 - pisi     52067 82241 27   0.633  1.0000
##
## P value adjustment: tukey method for comparing a family of 13 estimates
```

```
hist(resid(epi_growth_mod))
```

Histogram of resid(epi_growth_mod)
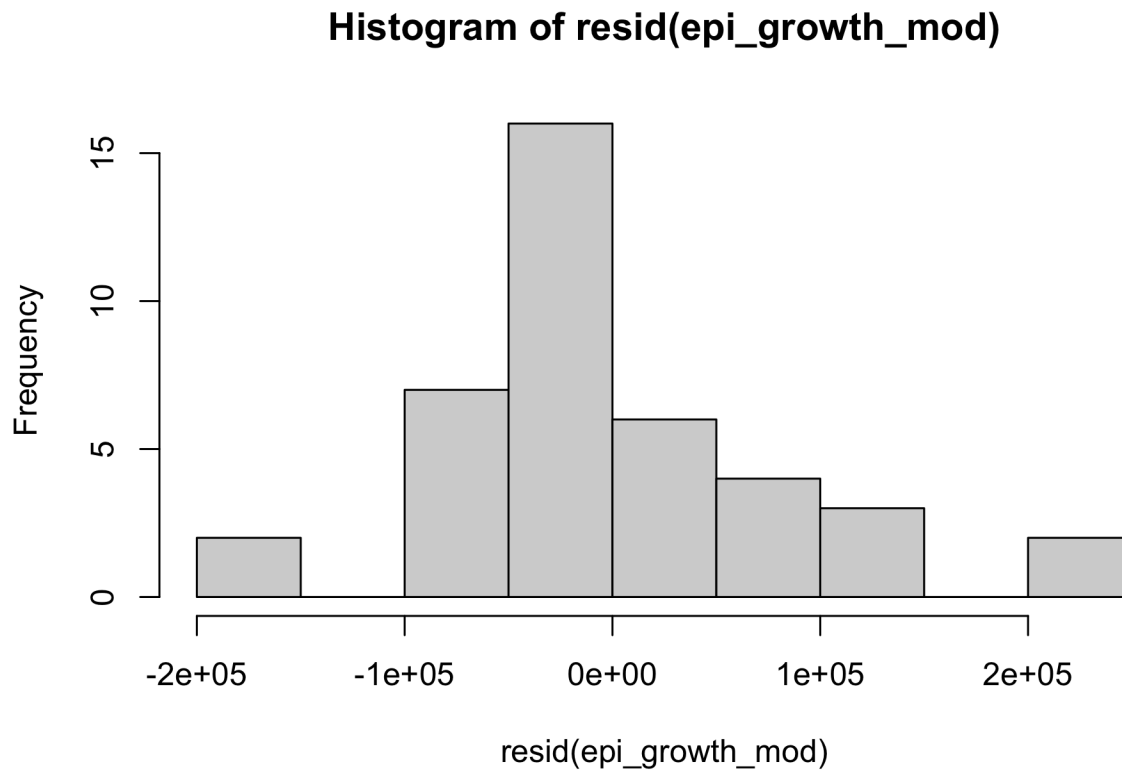
Combine epiphytic and virulence data

```
# Assuming strain_stats has a column 'strain' and surv_probs has a column
# 'treatment' Rename columns if necessary to match the key for joining
epi_growth_mean_per_strain <- epi_growth_mean_per_strain %>%
    rename(treatment = strain)

# Combine strain_stats and surv_probs using left_join
epi_virulence_data <- left_join(epi_growth_mean_per_strain, surv_probs, by = "treatment")

# Print the combined data
print(epi_virulence_data)
```

```
## # A tibble: 39 x 10
##    treatment mean_CFU_per_10_leaf~1 sd_CFU sample person block species plant_rep
##    <chr>                     <dbl>  <dbl> <chr>  <chr>  <chr> <chr>   <chr>
## 1 194                     156472. 1.43e5 194-A1 Havi   2-1   P. par~ A1
## 2 194                     156472. 1.43e5 194-A1 Havi   2-1   P. par~ A1
## 3 194                     156472. 1.43e5 194-A1 Havi   2-1   P. par~ A1
## 4 200                     261214. 1.81e5 200-A1 Havi   2-6   P. flu~ A1
## 5 200                     261214. 1.81e5 200-A1 Havi   2-6   P. flu~ A1
## 6 200                     261214. 1.81e5 200-A1 Havi   2-6   P. flu~ A1
## 7 204                     186888. 1.89e5 204-A1 Havi   2-3   P. flu~ A1
```

```
##  8 204                      186888. 1.89e5 204-A1 Havi   2-3   P. flu~ A1
##  9 204                      186888. 1.89e5 204-A1 Havi   2-3   P. flu~ A1
## 10 205                      207750  2.40e5 205-A1 Sara   3     P. flu~ A1
## # i 29 more rows
## # i abbreviated name: 1: mean_CFU_per_10_leafdiscs
## # i 2 more variables: time <dbl>, surv_prob <dbl>
```

```r
subset_data_72 <- epi_virulence_data %>%
    filter(time == 72)

# Calculate correlation coefficient between survival probability and epiphytic
# growth ability
correlation_72 <- -1 * cor(subset_data_72$surv_prob, subset_data_72$mean_CFU_per_10_leafdiscs,
    method = "pearson")
```

Plot epi virulence plot

```r
# Create scatter plot for 72 hours
epi_virulence_plot_strains <- ggplot(subset_data_72, aes(x = mean_CFU_per_10_leafdiscs, y = 1-surv_prob
  geom_point(aes(color = treatment, shape = species)) +
  geom_smooth(method = "lm", se = TRUE, color = "black", fill = "black", alpha = 0.15, linewidth = 0.5)
  labs(title = "72 Hours",
       x = "Mean CFU per 10 Leaf Discs (Log Transformed)",
       y = "Aphid Mortality at 72 Hours (%)",
       caption = paste("Pearson's Correlation Coefficient:", round(correlation_72, 2))) +
  scale_x_continuous(trans = "log10") +  # Apply log transformation to x-axis
  theme_minimal()

epi_virulence_plot_strains
```

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

72 Hours

Pearson's Correlation Coefficient: 0.51
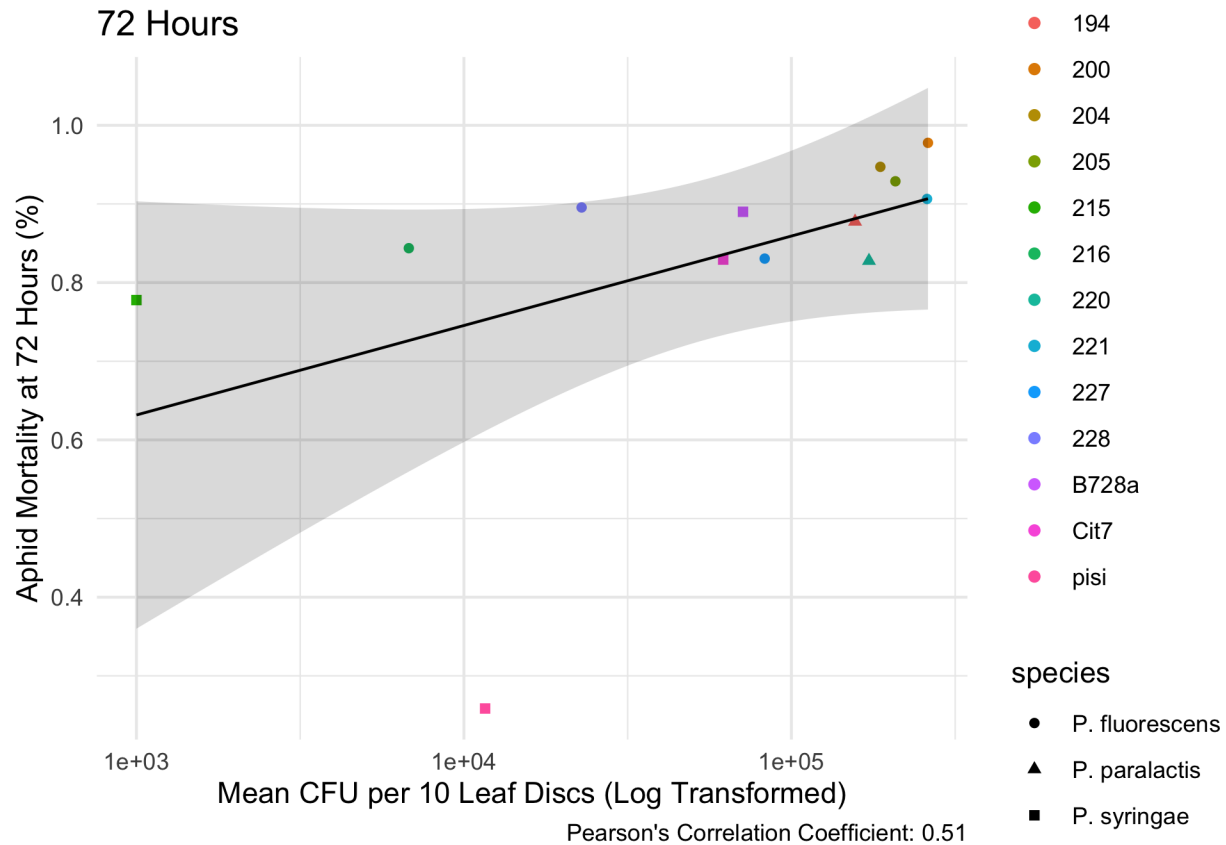
Session Information

```
devtools::session_info()
```

```
## ─ Session info ─────────────────────────────────────────────────────────
##  setting  value
##  version  R version 4.4.0 (2024-04-24)
##  os       macOS Ventura 13.4
##  system   x86_64, darwin20
##  ui       X11
##  language (EN)
##  collate  en_US.UTF-8
##  ctype    en_US.UTF-8
##  tz       America/New_York
##  date     2024-06-24
##  pandoc   3.1.11 @ /Applications/RStudio.app/Contents/Resources/app/quarto/bin/tools/x86_64/ (via rma
##
## ─ Packages ─────────────────────────────────────────────────────────────
##  ! package     * version date (UTC) lib source
##  P abind         1.4-5   2016-07-21 [?] RSPM
```

```
##  P backports        1.4.1   2021-12-13 [?] CRAN (R 4.4.0)
##  P beeswarm         0.4.0   2021-06-01 [?] CRAN (R 4.4.0)
##  P bit              4.0.5   2022-11-15 [?] CRAN (R 4.4.0)
##  P bit64            4.0.5   2020-08-30 [?] CRAN (R 4.4.0)
##  P broom            1.0.6   2024-05-17 [?] CRAN (R 4.4.0)
##  P broom.helpers    1.15.0  2024-04-05 [?] CRAN (R 4.4.0)
##  P cachem           1.0.8   2023-05-01 [?] CRAN (R 4.4.0)
##  P car              3.1-2   2023-03-30 [?] RSPM
##  P carData          3.0-5   2022-01-06 [?] RSPM
##  P cellranger       1.1.0   2016-07-27 [?] CRAN (R 4.4.0)
##  P cli              3.6.2   2023-12-11 [?] CRAN (R 4.4.0)
##  P colorspace       2.1-0   2023-01-23 [?] CRAN (R 4.4.0)
##  P cowplot        * 1.1.3   2024-01-22 [?] CRAN (R 4.4.0)
##  P crayon           1.5.2   2022-09-29 [?] CRAN (R 4.4.0)
##  P data.table       1.15.4  2024-03-30 [?] CRAN (R 4.4.0)
##  P devtools       * 2.4.5   2022-10-11 [?] RSPM
##  P digest           0.6.35  2024-03-11 [?] CRAN (R 4.4.0)
##  P dplyr          * 1.1.4   2023-11-17 [?] CRAN (R 4.4.0)
##  P ellipsis         0.3.2   2021-04-29 [?] RSPM
##  P emmeans        * 1.10.2  2024-05-20 [?] RSPM
##  P estimability     1.5.1   2024-05-12 [?] RSPM
##  P evaluate         0.23    2023-11-01 [?] CRAN (R 4.4.0)
##  P fansi            1.0.6   2023-12-08 [?] CRAN (R 4.4.0)
##  P farver           2.1.2   2024-05-13 [?] CRAN (R 4.4.0)
##  P fastmap          1.1.1   2023-02-24 [?] CRAN (R 4.4.0)
##  P forcats        * 1.0.0   2023-01-29 [?] CRAN (R 4.4.0)
##  P formatR        * 1.14    2023-01-17 [?] RSPM
##  P fs               1.6.4   2024-04-25 [?] CRAN (R 4.4.0)
##  P generics         0.1.3   2022-07-05 [?] CRAN (R 4.4.0)
##  P ggbeeswarm     * 0.7.2   2023-04-29 [?] CRAN (R 4.4.0)
##  P ggplot2        * 3.5.1   2024-04-23 [?] CRAN (R 4.4.0)
##  P ggpubr         * 0.6.0   2023-02-10 [?] RSPM
##  P ggsignif         0.6.4   2022-10-13 [?] RSPM
##  P ggsurvfit      * 1.1.0   2024-05-08 [?] CRAN (R 4.4.0)
##  P glue             1.7.0   2024-01-09 [?] CRAN (R 4.4.0)
##  P gridExtra      * 2.3     2017-09-09 [?] RSPM
##  P gt               0.10.1  2024-01-17 [?] CRAN (R 4.4.0)
##  P gtable           0.3.5   2024-04-22 [?] CRAN (R 4.4.0)
##  P gtsummary      * 1.7.2   2023-07-15 [?] CRAN (R 4.4.0)
##  P here           * 1.0.1   2020-12-13 [?] CRAN (R 4.4.0)
##  P hms              1.1.3   2023-03-21 [?] CRAN (R 4.4.0)
##  P htmltools        0.5.8.1 2024-04-04 [?] CRAN (R 4.4.0)
##  P htmlwidgets      1.6.4   2023-12-06 [?] CRAN (R 4.4.0)
##  P httpuv           1.6.15  2024-03-26 [?] RSPM
##  P km.ci            0.5-6   2022-04-06 [?] RSPM
##  P KMsurv           0.1-5   2012-12-03 [?] RSPM
##  P knitr          * 1.46    2024-04-06 [?] CRAN (R 4.4.0)
##  P labeling         0.4.3   2023-08-29 [?] CRAN (R 4.4.0)
##  P later            1.3.2   2023-12-06 [?] RSPM
##  P lattice          0.22-6  2024-03-20 [?] CRAN (R 4.4.0)
##  P lifecycle        1.0.4   2023-11-07 [?] CRAN (R 4.4.0)
##  P lubridate      * 1.9.3   2023-09-27 [?] CRAN (R 4.4.0)
##  P magrittr         2.0.3   2022-03-30 [?] CRAN (R 4.4.0)
##  P Matrix           1.7-0   2024-03-22 [?] CRAN (R 4.4.0)
```

```
##  P memoise         2.0.1   2021-11-26 [?] CRAN (R 4.4.0)
##  P mgcv            1.9-1   2023-12-21 [?] CRAN (R 4.4.0)
##  P mime            0.12    2021-09-28 [?] CRAN (R 4.4.0)
##  P miniUI          0.1.1.1 2018-05-18 [?] RSPM
##  P munsell         0.5.1   2024-04-01 [?] CRAN (R 4.4.0)
##  P mvtnorm         1.2-5   2024-05-21 [?] RSPM
##  P nlme            3.1-164 2023-11-27 [?] CRAN (R 4.4.0)
##  P pacman          0.5.1   2019-03-11 [?] CRAN (R 4.4.0)
##  P paletteer     * 1.6.0   2024-01-21 [?] RSPM
##  P pillar          1.9.0   2023-03-22 [?] CRAN (R 4.4.0)
##  P pkgbuild        1.4.4   2024-03-17 [?] RSPM
##  P pkgconfig       2.0.3   2019-09-22 [?] CRAN (R 4.4.0)
##  P pkgload         1.3.4   2024-01-16 [?] RSPM
##  P profvis         0.3.8   2023-05-02 [?] RSPM
##  P promises        1.3.0   2024-04-05 [?] RSPM
##  P purrr         * 1.0.2   2023-08-10 [?] CRAN (R 4.4.0)
##  P R6              2.5.1   2021-08-19 [?] CRAN (R 4.4.0)
##  P RColorBrewer  * 1.1-3   2022-04-03 [?] CRAN (R 4.4.0)
##  P Rcpp            1.0.12  2024-01-09 [?] CRAN (R 4.4.0)
##  P readr         * 2.1.5   2024-01-10 [?] CRAN (R 4.4.0)
##  P readxl        * 1.4.3   2023-07-06 [?] CRAN (R 4.4.0)
##  P rematch2        2.1.2   2020-05-01 [?] CRAN (R 4.4.0)
##  P remotes         2.5.0   2024-03-17 [?] CRAN (R 4.4.0)
##    renv            1.0.7   2024-04-11 [1] CRAN (R 4.4.0)
##  P rlang           1.1.3   2024-01-10 [?] CRAN (R 4.4.0)
##  P rmarkdown       2.26    2024-03-05 [?] CRAN (R 4.4.0)
##  P rprojroot       2.0.4   2023-11-05 [?] CRAN (R 4.4.0)
##  P rstatix         0.7.2   2023-02-01 [?] RSPM
##  P rstudioapi      0.16.0  2024-03-24 [?] CRAN (R 4.4.0)
##  P scales          1.3.0   2023-11-28 [?] CRAN (R 4.4.0)
##  P sessioninfo     1.2.2   2021-12-06 [?] RSPM
##  P shiny           1.8.1.1 2024-04-02 [?] RSPM
##  P stringi         1.8.3   2023-12-11 [?] CRAN (R 4.4.0)
##  P stringr       * 1.5.1   2023-11-14 [?] CRAN (R 4.4.0)
##  P survival      * 3.6-4   2024-04-24 [?] CRAN (R 4.4.0)
##  P survminer     * 0.4.9   2021-03-09 [?] RSPM
##  P survMisc        0.5.6   2022-04-07 [?] RSPM
##  P tibble        * 3.2.1   2023-03-20 [?] CRAN (R 4.4.0)
##  P tidycmprsk    * 1.0.0   2023-10-30 [?] CRAN (R 4.4.0)
##  P tidyr         * 1.3.1   2024-01-24 [?] CRAN (R 4.4.0)
##  P tidyselect      1.2.1   2024-03-11 [?] CRAN (R 4.4.0)
##  P tidyverse     * 2.0.0   2023-02-22 [?] CRAN (R 4.4.0)
##  P timechange      0.3.0   2024-01-18 [?] CRAN (R 4.4.0)
##  P tzdb            0.4.0   2023-05-12 [?] CRAN (R 4.4.0)
##  P urlchecker      1.0.1   2021-11-30 [?] RSPM
##  P usethis       * 2.2.3   2024-02-19 [?] RSPM
##  P utf8            1.2.4   2023-10-22 [?] CRAN (R 4.4.0)
##  P vctrs           0.6.5   2023-12-01 [?] CRAN (R 4.4.0)
##  P vipor           0.4.7   2023-12-18 [?] CRAN (R 4.4.0)
##  P viridis       * 0.6.5   2024-01-29 [?] RSPM
##  P viridisLite   * 0.4.2   2023-05-02 [?] CRAN (R 4.4.0)
##  P vroom           1.6.5   2023-12-05 [?] CRAN (R 4.4.0)
##  P wesanderson   * 0.3.7   2023-10-31 [?] RSPM
##  P withr           3.0.0   2024-01-16 [?] CRAN (R 4.4.0)
```

```
##  P xfun           0.43    2024-03-25 [?] CRAN (R 4.4.0)
##  P xml2           1.3.6   2023-12-04 [?] CRAN (R 4.4.0)
##  P xtable         1.8-4   2019-04-21 [?] RSPM
##  P yaml           2.3.8   2023-12-11 [?] CRAN (R 4.4.0)
##  P zoo            1.8-12  2023-04-13 [?] RSPM
##
##  [1] /Users/zahavahrojer/Desktop/Cornell/Hendry Lab/pseud-epi-growth/renv/library/macos/R-4.4/x86_64-
##  [2] /Users/zahavahrojer/Library/Caches/org.R-project.R/R/renv/sandbox/macos/R-4.4/x86_64-apple-darw:
##
##  P -- Loaded and on-disk path mismatch.
##
##  -----------------------------------------------------------------------------
```