TRhizo-localAdaptation

Cockerham's Test

David Murray-Stoker

Contents

Load Packages & Data

```
## Load the tidyverse
library(tidyverse)

## Packages for analysis
library(broom)
library(lme4)
library(lmerTest)

## Read in data
biomass.data <- read_rds(file = "data/cleaned_biomass_data.rds")
nodule.data <- read_rds(file = "data/cleaned_nodule_data.rds")
```

Quantify Genetic Variation

Data Management

```
## Aboveground biomass data
# Local
aboveground.biomass.local.data <- biomass.data %>%
  select(Population, Microbiome, Block, Aboveground_Biomass) %>%
  filter(Microbiome == "Local")
# Rural
aboveground.biomass.rural.data <- biomass.data %>%
  select(Population, Microbiome, Block, Aboveground_Biomass) %>%
  filter(Microbiome == "Nonlocal_R")
# Urban
aboveground.biomass.urban.data <- biomass.data %>%
  select(Population, Microbiome, Block, Aboveground_Biomass) %>%
  filter(Microbiome == "Nonlocal_U")


## Nodule density
# Local | Ambient N
nodule.density.local.ambient.N.data <- nodule.data %>%
  filter(Nitrogen == "Ambient_N") %>%
  select(Population, Microbiome, Block, Nodule_Density) %>%
  filter(Microbiome == "Local")
# Rural | Ambient N
nodule.density.rural.ambient.N.data <- nodule.data %>%
  filter(Nitrogen == "Ambient_N") %>%
  select(Population, Microbiome, Block, Nodule_Density) %>%
  filter(Microbiome == "Nonlocal_R")
# Urban | Ambient N
nodule.density.urban.ambient.N.data <- nodule.data %>%
  filter(Nitrogen == "Ambient_N") %>%
  select(Population, Microbiome, Block, Nodule_Density) %>%
  filter(Microbiome == "Nonlocal_U")

# Local | N Addition
nodule.density.local.N.addition.data <- nodule.data %>%
  filter(Nitrogen == "N_Addition") %>%
  select(Population, Microbiome, Block, Nodule_Density) %>%
  filter(Microbiome == "Local")
# Rural | N Addition
nodule.density.rural.N.addition.data <- nodule.data %>%
  filter(Nitrogen == "N_Addition") %>%
  select(Population, Microbiome, Block, Nodule_Density) %>%
  filter(Microbiome == "Nonlocal_R")
# Urban | N Addition
nodule.density.urban.N.addition.data <- nodule.data %>%
  filter(Nitrogen == "N_Addition") %>%
  select(Population, Microbiome, Block, Nodule_Density) %>%
  filter(Microbiome == "Nonlocal_U")


## Fixing nodule density data
```

```r
# Local
fixing.nodule.density.local.data <- nodule.data %>%
  select(Population, Microbiome, Block, Fixing_Nodule_Density) %>%
  filter(Microbiome == "Local")
# Rural
fixing.nodule.density.rural.data <- nodule.data %>%
  select(Population, Microbiome, Block, Fixing_Nodule_Density) %>%
  filter(Microbiome == "Nonlocal_R")
# Urban
fixing.nodule.density.urban.data <- nodule.data %>%
  select(Population, Microbiome, Block, Fixing_Nodule_Density) %>%
  filter(Microbiome == "Nonlocal_U")
```

Aboveground Biomass

```
## Aboveground biomass
# Local
aboveground.biomass.local.Vg.LMM <- lmer(
  sqrt(Aboveground_Biomass) ~ Block + (1 | Population),
  data = aboveground.biomass.local.data,
  REML = TRUE
)

aboveground.biomass.local.Vg.df <- data.frame(VarCorr(aboveground.biomass.local.Vg.LMM)) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Local") %>%
  add_column(Nitrogen = "Combined")

# Rural
aboveground.biomass.rural.Vg.LMM <- lmer(
  sqrt(Aboveground_Biomass) ~ Block + (1 | Population),
  data = aboveground.biomass.rural.data,
  REML = TRUE
)

aboveground.biomass.rural.Vg.df <- data.frame(VarCorr(aboveground.biomass.rural.Vg.LMM)) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Rural") %>%
  add_column(Nitrogen = "Combined")

# Urban
aboveground.biomass.urban.Vg.LMM <- lmer(
  sqrt(Aboveground_Biomass) ~ Block + (1 | Population),
  data = aboveground.biomass.urban.data,
  REML = TRUE
)

aboveground.biomass.urban.Vg.df <- data.frame(VarCorr(aboveground.biomass.urban.Vg.LMM)) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Urban") %>%
  add_column(Nitrogen = "Combined")
```

Nodule Density

```
## Nodule density
# Ambient N | Local
nodule.density.local.ambient.N.Vg.LMM <- lmer(
  log(Nodule_Density + 1) ~ Block + (1 | Population),
  data = nodule.density.local.ambient.N.data,
  REML = TRUE
)

nodule.density.local.ambient.N.Vg.df <- data.frame(
  VarCorr(nodule.density.local.ambient.N.Vg.LMM)
) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Local") %>%
  add_column(Nitrogen = "Ambient_N")

# Ambient N | Rural
nodule.density.rural.ambient.N.Vg.LMM <- lmer(
  log(Nodule_Density + 1) ~ Block + (1 | Population),
  data = nodule.density.rural.ambient.N.data,
  REML = TRUE
)

nodule.density.rural.ambient.N.Vg.df <- data.frame(
  VarCorr(nodule.density.rural.ambient.N.Vg.LMM)
) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Rural") %>%
  add_column(Nitrogen = "Ambient_N")

# Ambient N | Urban
nodule.density.urban.ambient.N.Vg.LMM <- lmer(
  log(Nodule_Density + 1) ~ Block + (1 | Population),
  data = nodule.density.urban.ambient.N.data,
  REML = TRUE
)

nodule.density.urban.ambient.N.Vg.df <- data.frame(
  VarCorr(nodule.density.urban.ambient.N.Vg.LMM)
) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Urban") %>%
  add_column(Nitrogen = "Ambient_N")


# N Addition | Local
nodule.density.local.N.addition.Vg.LMM <- lmer(
  log(Nodule_Density + 1) ~ Block + (1 | Population),
  data = nodule.density.local.N.addition.data,
  REML = TRUE
)

nodule.density.local.N.addition.Vg.df <- data.frame(
```

```
  VarCorr(nodule.density.local.N.addition.Vg.LMM)
) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Local") %>%
  add_column(Nitrogen = "N_Addition")

# N Addition | Rural
nodule.density.rural.N.addition.Vg.LMM <- lmer(
  log(Nodule_Density + 1) ~ Block + (1 | Population),
  data = nodule.density.rural.N.addition.data,
  REML = TRUE
)

nodule.density.rural.N.addition.Vg.df <- data.frame(
  VarCorr(nodule.density.rural.N.addition.Vg.LMM)
) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Rural") %>%
  add_column(Nitrogen = "N_Addition")

# N Addition | Urban
nodule.density.urban.N.addition.Vg.LMM <- lmer(
  log(Nodule_Density + 1) ~ Block + (1 | Population),
  data = nodule.density.urban.N.addition.data,
  REML = TRUE
)

nodule.density.urban.N.addition.Vg.df <- data.frame(
  VarCorr(nodule.density.urban.N.addition.Vg.LMM)
) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Urban") %>%
  add_column(Nitrogen = "N_Addition")
```

Fixing Nodule Density

```
## Fixing nodule density
# Local
fixing.nodule.density.local.Vg.LMM <- lmer(
  log(Fixing_Nodule_Density + 1) ~ Block + (1 | Population),
  data = fixing.nodule.density.local.data,
  REML = TRUE
)

fixing.nodule.density.local.Vg.df <- data.frame(VarCorr(fixing.nodule.density.local.Vg.LMM)) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Local") %>%
  add_column(Nitrogen = "Combined")

# Rural
fixing.nodule.density.rural.Vg.LMM <- lmer(
  log(Fixing_Nodule_Density + 1) ~ Block + (1 | Population),
  data = fixing.nodule.density.rural.data,
  REML = TRUE
)

fixing.nodule.density.rural.Vg.df <- data.frame(VarCorr(fixing.nodule.density.rural.Vg.LMM)) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Rural") %>%
  add_column(Nitrogen = "Combined")

# Urban
fixing.nodule.density.urban.Vg.LMM <- lmer(
  log(Fixing_Nodule_Density + 1) ~ Block + (1 | Population),
  data = fixing.nodule.density.urban.data,
  REML = TRUE
)

fixing.nodule.density.urban.Vg.df <- data.frame(VarCorr(fixing.nodule.density.urban.Vg.LMM)) %>%
  filter(grp == "Population") %>%
  add_column(Microbiome = "Urban") %>%
  add_column(Nitrogen = "Combined")
```

Export Genetic Variation Data

```
## Combine and export data for management in Excel in prep for Cockerham's test
# Bind aboveground biomass data
aboveground.biomass.combined.Vg.data <- bind_rows(
  aboveground.biomass.local.Vg.df,
  aboveground.biomass.rural.Vg.df,
  aboveground.biomass.urban.Vg.df
)

# Bind nodule density data
nodule.density.combined.Vg.data <- bind_rows(
  nodule.density.local.ambient.N.Vg.df,
  nodule.density.rural.ambient.N.Vg.df,
  nodule.density.urban.ambient.N.Vg.df,
  nodule.density.local.N.addition.Vg.df,
  nodule.density.rural.N.addition.Vg.df,
  nodule.density.urban.N.addition.Vg.df
)

# Bind fixing nodule density data
fixing.nodule.density.combined.Vg.data <- bind_rows(
  fixing.nodule.density.local.Vg.df,
  fixing.nodule.density.rural.Vg.df,
  fixing.nodule.density.urban.Vg.df
)

## Bind all data together
combined.Vg.data <- bind_rows(
  aboveground.biomass.combined.Vg.data,
  nodule.density.combined.Vg.data,
  fixing.nodule.density.combined.Vg.data
) %>%
  add_column(Trait = c(
    rep("Aboveground_Biomass", 3),
    rep("Nodule_Density", 6),
    rep("Fixing_Nodule_Density", 3)
  )) %>%
  select(Trait, Microbiome:Nitrogen, vcov) %>%
  rename(Vg = vcov)

## Export data to prepare for Cockerham's test
write_csv(combined.Vg.data, file = "data/cockerham_data_uncleaned.csv")
```

Genetic Correlations

Data Management

```
## Format data to calculate correlations
full.data <- nodule.data %>%
  full_join(
    biomass.data %>% select(UID, Aboveground_Biomass),
    by = c("UID")
  ) %>%
  select(
    Population:Nitrogen, Aboveground_Biomass, Nodule_Density:Fixing_Nodule_Density
  ) %>%
  drop_na()

## Local by Rural data
local.by.rural.data <- full.data %>%
  filter(Microbiome != "Nonlocal_U") %>%
  group_by(Population, Microbiome) %>%
  select(-Nodule_Density) %>%
  summarise(
    AG_Biomass = mean(Aboveground_Biomass),
    Fix_Nod_Density = mean(Fixing_Nodule_Density),
    .groups = "keep"
  ) %>%
  ungroup() %>%
  drop_na() %>%
  pivot_wider(names_from = Microbiome, values_from = AG_Biomass:Fix_Nod_Density)

## Local by Urban data
local.by.urban.data <- full.data %>%
  filter(Microbiome != "Nonlocal_R") %>%
  group_by(Population, Microbiome) %>%
  select(-Nodule_Density) %>%
  summarise(
    AG_Biomass = mean(Aboveground_Biomass),
    Fix_Nod_Density = mean(Fixing_Nodule_Density),
    .groups = "keep"
  ) %>%
  ungroup() %>%
  drop_na() %>%
  pivot_wider(names_from = Microbiome, values_from = AG_Biomass:Fix_Nod_Density)

## Rural by Urban data
rural.by.urban.data <- full.data %>%
  filter(Microbiome != "Local") %>%
  group_by(Population, Microbiome) %>%
  select(-Nodule_Density) %>%
  summarise(
    AG_Biomass = mean(Aboveground_Biomass),
    Fix_Nod_Density = mean(Fixing_Nodule_Density),
    .groups = "keep"
  ) %>%
  ungroup() %>%
```

```r
  drop_na() %>%
  pivot_wider(names_from = Microbiome, values_from = AG_Biomass:Fix_Nod_Density)

## Nodule density data (N-treatment specific)
# Local by Rural data
nodule.density.local.by.rural.data <- full.data %>%
  filter(Microbiome != "Nonlocal_U") %>%
  group_by(Population, Microbiome, Nitrogen) %>%
  summarise(
    Nod_Density = mean(Nodule_Density),
    .groups = "keep"
  ) %>%
  ungroup() %>%
  pivot_wider(names_from = Microbiome:Nitrogen, values_from = Nod_Density) %>%
  drop_na()

# Local by Urban data
nodule.density.local.by.urban.data <- full.data %>%
  filter(Microbiome != "Nonlocal_R") %>%
  group_by(Population, Microbiome, Nitrogen) %>%
  summarise(
    Nod_Density = mean(Nodule_Density),
    .groups = "keep"
  ) %>%
  ungroup() %>%
  pivot_wider(names_from = Microbiome:Nitrogen, values_from = Nod_Density) %>%
  drop_na()

# Rural by Urban data
nodule.density.rural.by.urban.data <- full.data %>%
  filter(Microbiome != "Local") %>%
  group_by(Population, Microbiome, Nitrogen) %>%
  summarise(
    Nod_Density = mean(Nodule_Density),
    .groups = "keep"
  ) %>%
  ungroup() %>%
  pivot_wider(names_from = Microbiome:Nitrogen, values_from = Nod_Density) %>%
  drop_na()
```

Correlations

Aboveground Biomass & Fixing Nodule Density

Table 1: Fitness correlations between local and nonlocal-rural for aboveground biomass and fixing nodule density.

	AG_Biomass_Nonlocal_R	AG_Biomass_Local	Fix_Nod_Density_Nonlocal_R
AG_Biomass_Nonlocal_R	1.000	0.561	0.104
AG_Biomass_Local	0.561	1.000	0.235
Fix_Nod_Density_Nonlocal_R	0.104	0.235	1.000
Fix_Nod_Density_Local	0.089	0.135	0.251

Table 2: Fitness correlations between local and nonlocal-urban for aboveground biomass and fixing nodule density.

	AG_Biomass_Nonlocal_U	AG_Biomass_Local	Fix_Nod_Density_Nonlocal_U
AG_Biomass_Nonlocal_U	1.000	0.495	0.123
AG_Biomass_Local	0.495	1.000	0.180
Fix_Nod_Density_Nonlocal_U	0.123	0.180	1.000
Fix_Nod_Density_Local	0.264	0.135	0.305

Table 3: Fitness correlations between nonlocal-rural and nonlocal-urban for aboveground biomass and fixing nodule density.

	AG_Biomass_Nonlocal_R	AG_Biomass_Nonlocal_U	Fix_Nod_Density_Nonloc
AG_Biomass_Nonlocal_R	1.000	0.611	
AG_Biomass_Nonlocal_U	0.611	1.000	
Fix_Nod_Density_Nonlocal_R	0.104	0.139	
Fix_Nod_Density_Nonlocal_U	0.177	0.123	-

Nodule Density

Table 4: Fitness correlations between local and nonlocal-rural by nitrogen treatment for nodule density.

	Nonlocal_R_Ambient_N	Nonlocal_R_N_Addition	Local_Ambient_N	Local_N_Add
Nonlocal_R_Ambient_N	1.000	0.046	0.053	
Nonlocal_R_N_Addition	0.046	1.000	0.044	
Local_Ambient_N	0.053	0.044	1.000	
Local_N_Addition	0.344	0.311	0.127	

Table 5: Fitness correlations between local and nonlocal-urban by nitrogen treatment for nodule density.

	Nonlocal_U_Ambient_N	Nonlocal_U_N_Addition	Local_Ambient_N	Local_N_Add
Nonlocal_U_Ambient_N	1.000	0.362	0.153	
Nonlocal_U_N_Addition	0.362	1.000	0.011	
Local_Ambient_N	0.153	0.011	1.000	
Local_N_Addition	0.394	0.254	0.120	

Table 6: Fitness correlations between nonlocal-rural and nonlocal-urban by nitrogen treatment for nodule density.

	Nonlocal_R_Ambient_N	Nonlocal_R_N_Addition	Nonlocal_U_Ambient_N	Nonloca
Nonlocal_R_Ambient_N	1.000	0.084	-0.282	
Nonlocal_R_N_Addition	0.084	1.000	0.320	
Nonlocal_U_Ambient_N	-0.282	0.320	1.000	
Nonlocal_U_N_Addition	-0.189	-0.050	0.362	

Cockerham's Test

```
## Cockerham data
cockerham.data <- read_csv(
  "data/cockerham_data_cleaned.csv",
  col_types = c("ffnnn"),
  show_col_types = FALSE
)
```

Set Functions

```
## Function for imperfect correlation (page 88, Cockerham 1963)
imperfect_correlation <- function(Vg1, Vg2, Rg) {
  out <- 2 * sqrt(Vg1) * sqrt(Vg2) * (1 - Rg)

  return(out)
}

## Function for heterogeneous variances (page 88, Cockerham 1963)
heterogeneous_variances <- function(Vg1, Vg2) {
  out <- ((sqrt(Vg1) - sqrt(Vg2))^2)

  return(out)
}

## Calculate imperfect correlation values for each row
cockerham.data$Crossing <- sapply(1:nrow(cockerham.data), FUN = function(r) {
  Vg1 <- cockerham.data$Vg1[r]
  Vg2 <- cockerham.data$Vg2[r]
  Rg <- cockerham.data$Rg[r]

  crossing.value <- imperfect_correlation(Vg1, Vg2, Rg)

  return(crossing.value)
})

## Calculate heterogeneous variance values for each row
cockerham.data$Heterogeneous_Variances <- sapply(1:nrow(cockerham.data), FUN = function(r) {
  Vg1 <- cockerham.data$Vg1[r]
  Vg2 <- cockerham.data$Vg2[r]

  heterogeneous.variance.value <- heterogeneous_variances(Vg1, Vg2)

  return(heterogeneous.variance.value)
})
```

Calculations for Cockerham's Test

Aboveground Biomass

```
## Group by trait and calculate sum of imperfect correlation values
# Divide by number of environments*(number of environments - 1) as per Cockerham 1963
aboveground.biomass.crossing <- cockerham.data %>%
  filter(Trait == "Aboveground_Biomass") %>%
  summarise(Crossing_Sum = sum(Crossing) / (3 * (3 - 1)))

## Group by trait and calculate sum of heterogeneous variances values
# Divide by number of environments*(number of environments - 1) as per Cockerham 1963
aboveground.biomass.heterogeneous.variance <- cockerham.data %>%
  filter(Trait == "Aboveground_Biomass") %>%
  summarise(Hetero_Variance_Sum = sum(Heterogeneous_Variances) / (3 * (3 - 1)))

## Make tibble for aboveground biomass calculation
aboveground.biomass.cockerham.data <- tibble(
  Trait = "Aboveground_Biomass",
  aboveground.biomass.crossing,
  aboveground.biomass.heterogeneous.variance
) %>%
  mutate(Total_Variance = Crossing_Sum + Hetero_Variance_Sum) %>%
  mutate(Percent_Crossing = 100 * Crossing_Sum / Total_Variance)
```

Nodule Density

```
## Group by trait and calculate sum of imperfect correlation values
# Divide by number of environments*(number of environments - 1) as per Cockerham 1963
nodule.density.crossing <- cockerham.data %>%
  filter(Trait == "Nodule_Density") %>%
  summarise(Crossing_Sum = sum(Crossing) / (6 * (6 - 1)))

## Group by trait and calculate sum of heterogeneous variances values
# Divide by number of environments*(number of environments - 1) as per Cockerham 1963
nodule.density.heterogeneous.variance <- cockerham.data %>%
  filter(Trait == "Nodule_Density") %>%
  summarise(Hetero_Variance_Sum = sum(Heterogeneous_Variances) / (6 * (6 - 1)))

## Make tibble for Nodule Density calculation
nodule.density.cockerham.data <- tibble(
  Trait = "Nodule_Density",
  nodule.density.crossing,
  nodule.density.heterogeneous.variance
) %>%
  mutate(Total_Variance = Crossing_Sum + Hetero_Variance_Sum) %>%
  mutate(Percent_Crossing = 100 * Crossing_Sum / Total_Variance)
```

Fixing Nodule Density

```
## Group by trait and calculate sum of imperfect correlation values
# Divide by number of environments*(number of environments - 1) as per Cockerham 1963
fixing.nodule.density.crossing <- cockerham.data %>%
  filter(Trait == "Fixing_Nodule_Density") %>%
  summarise(Crossing_Sum = sum(Crossing) / (3 * (3 - 1)))

## Group by trait and calculate sum of heterogeneous variances values
# Divide by number of environments*(number of environments - 1) as per Cockerham 1963
fixing.nodule.density.heterogeneous.variance <- cockerham.data %>%
  filter(Trait == "Fixing_Nodule_Density") %>%
  summarise(Hetero_Variance_Sum = sum(Heterogeneous_Variances) / (3 * (3 - 1)))

## Make tibble for Nodule Density calculation
fixing.nodule.density.cockerham.data <- tibble(
  Trait = "Fixing_Nodule_Density",
  fixing.nodule.density.crossing,
  fixing.nodule.density.heterogeneous.variance
) %>%
  mutate(Total_Variance = Crossing_Sum + Hetero_Variance_Sum) %>%
  mutate(Percent_Crossing = 100 * Crossing_Sum / Total_Variance)
```

Results of Cockerham's Tests

Table 7: Results of Cockerham's test for aboveground biomass, nodule density, and fixing nodule density.

Trait	Crossing_Sum	Hetero_Variance_Sum	Total_Variance	Percent_Crossing
Aboveground_Biomass	0.00177	0.00005	0.00181	97.38075
Nodule_Density	0.00375	0.00057	0.00432	86.90567
Fixing_Nodule_Density	0.00190	0.00004	0.00194	97.87555

R Session Information

Table 8: Packages required for data management and analysis.

Package	Loaded Version	Date
broom	1.0.5	2023-06-09
dplyr	1.1.2	2023-04-20
forcats	1.0.0	2023-01-29
ggplot2	3.4.2	2023-04-03
kableExtra	1.3.4	2021-02-20
knitr	1.43	2023-05-25
lme4	1.1-34	2023-07-04
lmerTest	3.1-3	2020-10-23
lubridate	1.9.2	2023-02-10
Matrix	1.6-0	2023-07-08
purrr	1.0.1	2023-01-10
readr	2.1.4	2023-02-10
stringr	1.5.0	2022-12-02
tibble	3.2.1	2023-03-20
tidyr	1.3.0	2023-01-24
tidyverse	2.0.0	2023-02-22