```r
#Packages
library(dplyr)
library(ggtext)
library(ggpubr)
library(ggplot2)
library(car)
library(emmeans)
library(multcomp)
library(DHARMa)

# Part 1 ------------------------------------------------------------------

mating_df <- read.csv("part1.csv")

mating_summary <- mating_df %>%
  summarise(
    n_pre = sum(is.finite(pre)),
    mean_pre = mean(pre, na.rm = TRUE),
    sd_pre = sd(pre, na.rm = TRUE),
    se_pre = sd_pre / sqrt(n_pre),

    n_cop = sum(is.finite(cop)),
    mean_cop = mean(cop, na.rm = TRUE),
    sd_cop = sd(cop, na.rm = TRUE),
    se_cop = sd_cop / sqrt(n_cop)
  )
mating_summary



# Part 2: Choice ----------------------------------------------------------
kin_df <- read.csv("part2.csv")

kin_df <- kin_df %>%
  mutate(
    sex    = factor(sex, levels = c("Male", "Female")),
    kin     = factor(kin, levels = c("Y", "N"))
  )

table(kin_df$sex, kin_df$kin)

# Females
female_model <- glm(
  I(kin == "Y") ~ 1,
  family = binomial(link = "logit"),
  data = filter(kin_df, sex == "Female")
)

# Males
male_model <- glm(
  I(kin == "Y") ~ 1,
  family = binomial(link = "logit"),
  data = filter(kin_df, sex == "Male")
)

summary(female_model)
summary(male_model)

#check for overdispersion
dispersion_ratio <- deviance(male_model) / df.residual(male_model)
dispersion_ratio
dispersion_ratio <- deviance(female_model) / df.residual(female_model)
dispersion_ratio

kin_model <- glm(
  I(kin == "Y") ~ sex,
  family = binomial,
  data = kin_df
)
```

```
summary(kin_model)
dispersion_ratio <- deviance(kin_model) / df.residual(kin_model)
dispersion_ratio

Anova(kin_model)

###Plots###

# Sample size for each sex
n_df <- kin_df %>%
  count(sex, name = "N") %>%
  mutate(sex_label = paste0(sex, " (n = ", N, ")"))

# Calculate choice proportions
plot_df <- kin_df %>%
  count(sex, kin, name = "n") %>%
  group_by(sex) %>%
  mutate(
    percentage = n / sum(n) * 100,
    value = ifelse(kin == "Y", -percentage / 2, percentage / 2),
    label = paste0(round(percentage, 1), "%")
  ) %>%
  ungroup() %>%
  left_join(n_df, by = "sex")

# Plot

plot_df$sex_label <- gsub(
  "n =",
  "<i>n</i> =",
  plot_df$sex_label,
  fixed = TRUE
)

ggplot(plot_df, aes(x = sex_label, y = value, fill = kin)) +
  geom_col(width = 0.4, colour = "black", linewidth = 0.3) +
  geom_text(
    aes(y = value / 2, label = label),
    size = 5,
    family = "Helvetica"
  ) +
  geom_hline(yintercept = 0, colour = "black", linewidth = 0.4) +
  coord_flip() +
  scale_y_continuous(
    breaks = seq(-50, 50, 10),
    labels = function(x) abs(x) * 2,
    limits = c(-50, 50),
    expand = c(0, 0)
  ) +
  scale_fill_manual(
    values = c("Y" = "white", "N" = "grey60"),
    labels = c("Y" = "Sibling", "N" = "Unrelated"),
    name = "Mate choice"
  ) +
  labs(
    x = NULL,
    y = "Proportion of choices (%)"
  ) +
  theme_classic2(base_size = 18, base_family = "Helvetica") +
  theme(
    axis.text.x = element_text(colour = "black", size = 17),
    axis.text.y = element_markdown(colour = "black", size = 17),
    axis.title = element_text(size = 18),
    legend.position = "right",
    legend.text = element_text(size = 17),
    legend.title = element_text(size = 17),
    plot.margin = margin(
      0.5, 0.5, 0.5, 0.5,
      "cm"
```

```
    )
  )


# Part 2: Duration ----------------------------------------------------

kin_summary <- kin_df %>%
  group_by(sex, kin) %>%
  summarise(
    n_pre = sum(is.finite(pre)),
    mean_pre = mean(pre, na.rm = TRUE),
    sd_pre = sd(pre, na.rm = TRUE),
    se_pre = sd_pre / sqrt(n_pre),

    n_cop = sum(is.finite(cop)),
    mean_cop = mean(cop, na.rm = TRUE),
    sd_cop = sd(cop, na.rm = TRUE),
    se_cop = sd_cop / sqrt(n_cop)
  )
View(kin_summary)


par(mfrow = c(1, 2))
hist(kin_df$pre, main = "Pre", xlab = "Pre")
hist(kin_df$cop, main = "Cop", xlab = "Cop")

# Premating duration
pre_model <- lm(pre ~ sex * kin, data = kin_df)

# Copulation duration
cop_model <- lm(cop ~ sex * kin, data = kin_df)

anova(pre_model)
anova(cop_model)

par(mfrow = c(2, 2))
plot(pre_model)

par(mfrow = c(2, 2))
plot(cop_model)

emm_pre <- emmeans(pre_model, ~ sex * kin)
letters_pre <- cld(
  emm_pre,
  adjust = "holm",
  Letters = letters
)
letters_pre

emm_cop <- emmeans(cop_model, ~ sex * kin)
letters_cop <- cld(
  emm_cop,
  adjust = "holm",
  Letters = letters
)
letters_cop


# Part 3: Choice ------------------------------------------------------------
vir_df <- read.csv("part3a.csv")

vir_df <- vir_df %>%
  mutate(
    sex    = factor(sex, levels = c("Male", "Female")),
    virgin     = factor(virgin, levels = c("Y", "N"))
  )

table(vir_df$sex, vir_df$virgin)
```

```
# Females
female_modelv <- glm(
  I(virgin == "Y") ~ 1,
  family = binomial(link = "logit"),
  data = filter(vir_df, sex == "Female")
)

# Males
male_modelv <- glm(
  I(virgin == "Y") ~ 1,
  family = binomial(link = "logit"),
  data = filter(vir_df, sex == "Male")
)

summary(female_modelv)
summary(male_modelv)

#check for overdispersion
dispersion_ratio <- deviance(male_modelv) / df.residual(male_modelv)
dispersion_ratio
dispersion_ratio <- deviance(female_modelv) / df.residual(female_modelv)
dispersion_ratio

vir_model <- glm(
  I(virgin == "Y") ~ sex,
  family = binomial,
  data = vir_df
)
summary(vir_model)
dispersion_ratio <- deviance(vir_model) / df.residual(vir_model)
dispersion_ratio

Anova(vir_model)

###Plots###

# Sample size for each sex
vn_df <- vir_df %>%
  count(sex, name = "N") %>%
  mutate(sex_label = paste0(sex, " (n = ", N, ")"))

# Calculate choice proportions
vplot_df <- vir_df %>%
  count(sex, virgin, name = "n") %>%
  group_by(sex) %>%
  mutate(
    percentage = n / sum(n) * 100,
    value = ifelse(virgin == "Y", -percentage / 2, percentage / 2),
    label = paste0(round(percentage, 1), "%")
  ) %>%
  ungroup() %>%
  left_join(vn_df, by = "sex")

# Plot
library(ggtext)
vplot_df$sex_label <- gsub(
  "n =",
  "<i>n</i> =",
  vplot_df$sex_label,
  fixed = TRUE
)

#adjusting the % position on the plot for males
vplot_df <- vplot_df %>%
  mutate(
    label_inside = percentage >= 12,
    label_y = ifelse(
      label_inside,
      value / 2,
```

```
      value + ifelse(value < 0, -2, 2)
    ),
    label_hjust = ifelse(
      label_inside, 0.5,
      ifelse(value < 0, 1, 0)
    )
  )

ggplot(vplot_df, aes(x = sex_label, y = value, fill = virgin)) +
  geom_col(width = 0.4, colour = "black", linewidth = 0.3) +
  geom_text(
    data = vplot_df,
    aes(
      y = label_y,
      label = label,
      hjust = label_hjust
    ),
    size = 5,
    family = "Helvetica"
  ) +
  geom_text(
    data = distinct(vplot_df, sex_label),
    aes(x = sex_label, y = 20, label = "*"),
    inherit.aes = FALSE,
    size = 7,
    family = "Helvetica"
  ) +
  geom_hline(yintercept = 0, colour = "black", linewidth = 0.4) +
  coord_flip() +
  scale_y_continuous(
    breaks = seq(-50, 50, 10),
    labels = function(x) abs(x) * 2,
    limits = c(-50, 50),
    expand = c(0, 0)
  ) +
  scale_fill_manual(
    values = c("Y" = "white", "N" = "grey60"),
    labels = c("Y" = "Virgin", "N" = "Mated"),
    name = "Mating status"
  ) +
  labs(
    x = NULL,
    y = "Proportion of choices (%)"
  ) +
  theme_classic2(base_size = 18, base_family = "Helvetica") +
  theme(
    axis.text.x = element_text(colour = "black", size = 17),
    axis.text.y = element_markdown(colour = "black", size = 17),
    axis.title = element_text(size = 18),
    legend.position = "right",
    legend.text = element_text(size = 17),
    legend.title = element_text(size = 17),
    plot.margin = margin(
      0.5, 0.5, 0.5, 0.5,
      "cm"
    )
  )


# Part 3: Duration ------------------------------------------------------

vir_summary <- vir_df %>%
  group_by(sex, virgin) %>%
  summarise(
    n_pre = sum(is.finite(pre)),
    mean_pre = mean(pre, na.rm = TRUE),
    sd_pre = sd(pre, na.rm = TRUE),
    se_pre = sd_pre / sqrt(n_pre),
```

```
    n_cop = sum(is.finite(cop)),
    mean_cop = mean(cop, na.rm = TRUE),
    sd_cop = sd(cop, na.rm = TRUE),
    se_cop = sd_cop / sqrt(n_cop)
  )
View(vir_summary)

par(mfrow = c(1, 2))
hist(vir_df$pre, main = "Pre", xlab = "Pre")
hist(vir_df$cop, main = "Cop", xlab = "Cop")

# Premating latency
prev_model <- lm(pre ~ sex * virgin, data = vir_df)

# Copulation duration
copv_model <- lm(cop ~ sex * virgin, data = vir_df)

anova(prev_model)
anova(copv_model)

par(mfrow = c(2, 2))
plot(prev_model)

par(mfrow = c(2, 2))
plot(copv_model)

emm_prev <- emmeans(prev_model, ~ sex * virgin)
letters_prev <- cld(
  emm_prev,
  adjust = "holm",
  Letters = letters
)
letters_prev

emm_copv <- emmeans(copv_model, ~ sex * virgin)
letters_copv <- cld(
  emm_copv,
  adjust = "holm",
  Letters = letters
)
letters_copv


# Part 3: Reproduction ----------------------------------------------------
fec_df <- read.csv("part3b.csv")

fec_df <- fec_df %>%
  mutate(
    treat    = factor(treat, levels = c("Once", "Twice", "Unlimited"))
  )

table(fec_df$treat)

fec_summary <- fec_df %>%
  group_by(treat) %>%
  summarise(
    n_pre = sum(is.finite(pre)),
    mean_pre = mean(pre, na.rm = TRUE),
    sd_pre = sd(pre, na.rm = TRUE),
    se_pre = sd_pre / sqrt(n_pre),

    n_ovi = sum(is.finite(ovi)),
    mean_ovi = mean(ovi, na.rm = TRUE),
    sd_ovi = sd(ovi, na.rm = TRUE),
    se_ovi = sd_ovi / sqrt(n_ovi),

    n_fec = sum(is.finite(fec)),
    mean_fec = mean(fec, na.rm = TRUE),
    sd_fec = sd(fec, na.rm = TRUE),
```

```
    se_fec = sd_fec / sqrt(n_fec),

    n_daily = sum(is.finite(daily)),
    mean_daily = mean(daily, na.rm = TRUE),
    sd_daily = sd(daily, na.rm = TRUE),
    se_daily = sd_daily / sqrt(n_daily)
  )
View(fec_summary)

par(mfrow = c(2, 2))
hist(fec_df$pre, main = "Pre", xlab = "Pre")
hist(fec_df$ovi, main = "Ovi", xlab = "Ovi")
hist(fec_df$daily, main = "daily", xlab = "daily")
hist(fec_df$fec, main = "fec", xlab = "fec")

#Pre-oviposition
m_pre <- glm(pre ~ treat, family = Gamma(link = "log"), data = fec_df)
summary(m_pre)

res_pre <- simulateResiduals(m_pre)
plot(res_pre)

Anova(m_pre)

emm_preo <- emmeans(m_pre, ~ treat)
letters_preo <- cld(
  emm_preo,
  adjust = "holm",
  Letters = letters
)
letters_preo

#Oviposiiton
m_ovi <- lm(ovi ~ treat, data = fec_df)
summary(m_ovi)
anova(m_ovi)

par(mfrow = c(2, 2))
plot(m_ovi)
par(mfrow = c(1, 1))

emm_ovi <- emmeans(m_ovi, ~ treat)
letters_ovi <- cld(
  emm_ovi,
  adjust = "holm",
  Letters = letters
)
letters_ovi

#Daily oviposition
m_daily <- lm(daily ~ treat, data = fec_df)
summary(m_daily)
anova(m_daily)

par(mfrow = c(2, 2))
plot(m_daily)
par(mfrow = c(1, 1))

emm_daily <- emmeans(m_daily, ~ treat)
letters_daily <- cld(
  emm_daily,
  adjust = "holm",
  Letters = letters
)
letters_daily

#Fecundity
m_fec <- lm(fec ~ treat, data = fec_df)
summary(m_fec)
```

```
anova(m_fec)

par(mfrow = c(2, 2))
plot(m_fec)
par(mfrow = c(1, 1))

emm_fec <- emmeans(m_fec, ~ treat)
letters_fec <- cld(
  emm_fec,
  adjust = "holm",
  Letters = letters
)
letters_fec

#Correlation
once_df <- subset(fec_df, treat == "Once")

# Pearson correlation tests
cor_ovi <- cor.test(once_df$cop, once_df$ovi, method = "pearson")
cor_fec <- cor.test(once_df$cop, once_df$fec, method = "pearson")

#Plot
make_label <- function(test) {
  p_label <- if (test$p.value < 0.001) {
    "italic(P) < 0.001"
  } else {
    paste0("italic(P) == ", sprintf("%.3f", test$p.value))
  }

  paste0(
    'Pearson*"\\u2019s"~italic(r) == ',
    sprintf("%.2f", unname(test$estimate)),
    "*','~~",
    p_label
  )
}

p1 <- ggplot(once_df, aes(x = cop, y = ovi)) +
  geom_point(size = 2.5, colour = "black") +
  geom_smooth(
    method = "lm", se = TRUE,
    colour = "grey40", fill = "grey70", alpha = 0.35
  ) +
  annotate(
    "text", x = Inf, y = Inf,
    label = make_label(cor_ovi),
    parse = TRUE,
    hjust = 1.1, vjust = 1.2,
    size = 5, family = "Helvetica"
  ) +
  labs(x = "Copulation duration (min)", y = "Oviposition period (day)") +
  theme_classic2(base_family = "Helvetica", base_size = 14)+
  theme(
    axis.text.x = element_text(colour = "black", size = 17),
    axis.text.y = element_markdown(colour = "black", size = 17),
    axis.title = element_text(size = 18),
    legend.position = "right",
    legend.text = element_text(size = 17),
    legend.title = element_text(size = 17),
    plot.margin = margin(
      0.8, 0.5, 0.5, 0.5,
      unit = "cm"
    )
  )

p2 <- ggplot(once_df, aes(x = cop, y = fec)) +
  geom_point(size = 2.5, colour = "black") +
  geom_smooth(
    method = "lm", se = TRUE,
```

```
    colour = "grey40", fill = "grey70", alpha = 0.35
  ) +
  annotate(
    "text", x = Inf, y = Inf,
    label = make_label(cor_fec),
    parse = TRUE,
    hjust = 1.1, vjust = 1.2,
    size = 5, family = "Helvetica"
  ) +
  labs(x = "Copulation duration (min)", y = "Fecundity (number of eggs)") +
  theme_classic2(base_family = "Helvetica", base_size = 14) +
  theme(
    axis.text.x = element_text(colour = "black", size = 17),
    axis.text.y = element_markdown(colour = "black", size = 17),
    axis.title = element_text(size = 18),
    legend.position = "right",
    legend.text = element_text(size = 17),
    legend.title = element_text(size = 17),
    plot.margin = margin(
      0.8, 0.5, 0.5, 0.5,
      unit = "cm"
    )
  )

fig <- ggarrange(
  p1, p2,
  ncol = 2, nrow = 1,
  labels = c("A", "B"),
  font.label = list(family = "Helvetica", size = 17, face = "bold"),
  align = "hv"
)

fig
```