

```

----
title: "Supplementary Materials 1: The balance of nature: Critical Role of Species Environmental Responses for Stability"
author: "Til Hämig, Francesco Polazzo, Owen L. Petchey, Frank Pennekamp"
date: ``r format(Sys.time(), '%d %B, %Y')``
output:
  bookdown::html_document2:
    toc: true
    toc_float:
      collapsed: true
      smooth_scroll: true
    code_folding: hide
    keep_md: no
    fig_caption: false
editor_options:
  markdown:
    wrap: 72
----

```

```

```{r setup, include=FALSE}
rm(list=ls())
library("ggplot2")
library("tidyverse")
library("gridExtra")
#library(rEDM)
library("parallel")
library("pracma")
library("purrr")
library("signal")
library("ggfortify")
library("data.table")
library("patchwork")
library("codyn")
library("ggrepel")
library("lme4")
library("lmerTest")
library("MuMIn")
library("RColorBrewer")
library("broom")
library("relaimpo")
library("lavaanPlot")
library(ggbeeswarm)
library(performance)
library(broom)
library(kableExtra)
library(gt)
library(lavaan)

```

```

#devtools::install_github("canankarakoc/r_package_EDMhelper/EDMhelper")

```

```

```{r include = FALSE, echo = FALSE}
knitr::opts_chunk$set(echo = FALSE,
                      include = TRUE,
                      warning = FALSE,
                      message = FALSE,
                      cache = FALSE)
```

```

## # Introduction

The purpose of this document is to provide a reproducible record of all analyses and figures in the main article. The main article is focused on the effect of response diversity on community stability in fluctuating environments. We are going to look at the effect of the distribution of species responses, richness, temperature and nutrients on community temporal stability. Specifically, we are going to look at the effect of fundamental imbalance (our measurement of the distribution of species' fundamental responses) on temporal stability. Species' fundamental responses to environmental change can stabilise community biomass in two ways: through response diversity and/or through population stability. Thus, as response diversity is thought to stabilize temporal stability of aggregate community properties via asynchrony, we are going to look at the relationship between response diversity and asynchrony. Subsequently, we are going to look at the relationship between population stability and temporal stability of community biomass. Finally, we use a structural equation model to test the direct and indirect effects of balance on temporal stability of community biomass.

In this document, we also analyse the predictive power of imbalance and divergence on temporal stability, and we compare the unique explanatory power of imbalance and divergence. We also look at the interaction between divergence and richness, and between imbalance and richness. Finally, we assess variable importance using the relative importance of predictors in the full model. This part of the document (section 5), is not included in the main article, but it is an important part of the analysis that justify adopting imbalance instead of divergence, which was the original metric used to design the experiment.

This document is produced by an Rmarkdown file that includes code to reproduce from data all results presented in the main article.

## # Load datasets, Data wrangling and Imbalance calculation

```

```{r warning=FALSE, results='hide'}

divergence_df <- read_csv("Data/divergence_df.csv")
load("Data/dens_biomass_poly.RData")

dd_all_pred<-read.csv("Data/morph_dd_pred.csv")
dd_all_pred_nonoise<-read.csv("Data/morph_dd_pred_nonoise.csv")

load("Data/ciliate_traits.Rdata")

df_slopes <- read_csv("Data/df_slopes_cor.csv")

# needs to have id_new variable
ciliate_traits <- ciliate_traits %>%
  dplyr::mutate(
    # Remove dots from the date
    cleaned_date = gsub("\\.", "", date),
    # Extract the part of id after the underscore

```

```

    id_suffix = sub(".*(.*)", "\\1", id),
    # Combine cleaned_date, id_suffix, and species_initial into a new variable
    id_new = paste0(cleaned_date, id_suffix, composition)
  ) %>%
  # Optionally, remove the intermediate columns to clean up
  dplyr::select(-cleaned_date, -id_suffix, -new_id)

uniqueN(ciliate_traits$id_new) == nrow(ciliate_traits) # all unique ;)

id_dd <- full_join(dd_all_pred, dplyr::select(ciliate_traits, id_new, biomass), join_by("id_new"))

## add day variable

# create a day variable from the date variable

id_dd <- dplyr::mutate(id_dd, date = as.Date(date, format = "%d.%m.%y"))

earliest_date <- min(id_dd$date)
days_since_earliest <- as.numeric(id_dd$date - earliest_date) + 1
id_dd <- id_dd %>% dplyr::mutate(day = days_since_earliest)

# create a summarised df on microcosm level with each species separate
# Make sure, that we have n_frames and not N_frames
names(id_dd)[names(id_dd) == "N_frames"] <- "n_frames"

# extrapolation_factor <- 9.301902 # for 16 x magnification
# extrapolation_factor <- 9.828125 # for 25 x magnification
video_biomass_species <- c("C", "P", "S", "D", "L", "T")

biomasses <- id_dd %>%
  dplyr::group_by(day, temperature, nutrients, sample_ID, composition, predict_spec) %>% # group by xxx
  dplyr::summarize(
    biomass = sum(biomass * n_frames, na.rm = TRUE) / (1 * 125) # if not 3 videos corrections is done below with dens_factor
  ) %>%
  dplyr::mutate(
    biomass = biomass * extrapolation_factor,
  )

biomasses <- biomasses %>% dplyr::mutate(biomass = biomass * 1000)

dd_ts_id <- biomasses

# fill up missing dates with biomass <- 0

fill_dd <- expand_grid(sample_ID = unique(dd_ts_id$sample_ID), day = unique(dd_ts_id$day), predict_spec = unique(dd_ts_id$predict_spec))
complete_ts <- full_join(fill_dd, dd_ts_id, join_by(sample_ID, day, predict_spec))

complete_ts$biomass[is.na(complete_ts$biomass)] <- 0
complete_ts <- complete_ts %>% dplyr::mutate(composition = sub("_.*", "", sample_ID))
complete_ts <- complete_ts %>%
  dplyr::mutate(temperature = sapply(strsplit(as.character(sample_ID), "_"), function(x) paste(x[3], x[4], sep = "-")))
complete_ts <- dplyr::mutate(complete_ts, nutrients = gsub(".*Nut(?:.*)_", "\\1", sample_ID))

# Now remove wrong combinations of composition and predict_spec / predict_spec

complete_ts <- complete_ts %>%
  rowwise() %>%
  dplyr::filter(predict_spec %in% unlist(strsplit(composition, ""))) %>%
  ungroup()
complete_ts <- dplyr::mutate(complete_ts, temperature = as.character(temperature),
  nutrients = as.character(nutrients),
  richness = nchar(composition))

complete_ts <- complete_ts %>% group_by(sample_ID, composition, day) %>% dplyr::mutate(tot_biomass = sum(biomass))
complete_ts <- complete_ts %>% dplyr::mutate(biom_contribution = biomass / tot_biomass)

df_biomass_mod <- complete_ts

complete_ts <- complete_ts %>% dplyr::mutate(temperature = paste0(temperature, " °C"),
  nutrients = paste0(nutrients, " g/L"))

# introduce slopes of
names(df_slopes)[names(df_slopes) == "species_initial"] <- "predict_spec"

slope_ts <- full_join(dplyr::select(df_slopes, nutrients, predict_spec, temperature, slope), complete_ts)
slope_ts <- slope_ts %>% dplyr::mutate(w_slope = biom_contribution * slope,
  sign = sign(slope))

slope_ts <- slope_ts %>% group_by(sample_ID, temperature, nutrients, richness, composition, day, tot_biomass) %>% dplyr::summarize(
  sum_w_slopes = abs(sum(w_slope)),
  mean_abs_slope = mean(abs(slope)),
  sum_abs_slope = sum(abs(slope)),
  abs_sum_slope = abs(sum(slope)),
  symmetry = abs(sum(sign)))

slope_ts <- slope_ts %>% dplyr::mutate(richness = as.factor(richness))

## create new variable where it checks, where the last observation = 0 is; with complete_ts
aggr_ts <- slope_ts %>%
  group_by(sample_ID) %>%
  arrange(day) %>%
  mutate(
    # Create a flag for non-zero tot_biomass
    non_zero_biomass = tot_biomass != 0,
    # Find the last non-zero day
    last_non_zero_day = ifelse(any(non_zero_biomass), max(day[non_zero_biomass], na.rm = TRUE), NA),
    # Find the first zero day after the last non-zero day
    first_zero_day = ifelse(
      !is.na(last_non_zero_day),
      min(day[!non_zero_biomass & day > last_non_zero_day], na.rm = TRUE),

```

```

    NA
  ),
  # Flag for days after the first zero day
  is_after_first_zero_day = ifelse(!is.na(first_zero_day), day > first_zero_day, FALSE)
) %>%
ungroup()

aggr_ts<-aggr_ts%>%mutate(rep_var=sub("_[^_]+$", "", sample_ID))

biomass_ts<-
aggr_ts%>%dplyr::group_by(day,temperature,nutrients,richness)%>%summarize(tot_biom=mean(tot_biomass),se_tot_biom=sd(tot_biomass)/sqrt(
...

# Biomass

Let's have a look at the biomass dynamics in the different environmental treatments.

### tot biomass plot

```{r plot_biomass, fig.align="center", fig.height=8, fig.width=15}

plot_biomass<-ggplot(data=biomass_ts)+
 geom_ribbon(aes(x=day,y=tot_biom,fill=richness,color=richness,ymin=tot_biom-
se_tot_biom,ymax=tot_biom+se_tot_biom),show.legend=F,alpha=0.5)+
 geom_line(method="loess",se=F,aes(x=day,y=tot_biom,color=richness),size=0.8)+
 scale_color_viridis_d(option = "inferno", begin = 0.2, end = 0.8)+
 scale_fill_viridis_d(option = "inferno", begin = 0.2, end = 0.8)+
 theme_classic()+
 #geom_text_repel(data=biomass_ts%>%dplyr::filter(day==58,tot_biom!=0),aes(x=day,y=tot_biom,group=composition,label =
composition), size = 4,max.overlaps = 20) +
 facet_wrap(temperature~nutrients,scale="free_y")+
 theme(axis.title.x = element_text(size = 16),
 axis.title.y = element_text(size = 16),
 strip.background = element_rect(fill = "grey80", color = NA),
 strip.text = element_text(size = 12))+
 ylab(expression("Community biomass [mg ml"^-1 * "]"))

plot_biomass

Figure 1 : Community total biomass during the experiment in different environmental treatments. Different color represent
richness levels.

Main Results

We now look at the main results of the experiment. We are going to look first at the effect of richness, temperature and nutrients
on community temporal stability. Then, we are going to look at the relationship between divergence (original response diversity
metric) and temporal stability. Finally, we are going to look at the relationship between imbalance and temporal stability.

In the whole analysis, we calculated the temporal stability of total community biomass as the inverse of the coefficient of
variation (ICV) (i.e. $\frac{\sigma}{\mu}$).
```{r results='hide', echo=FALSE, warning=FALSE}
## discard all the rows that are part of time stretches where entire microcosms has (seemingly) gone extinct
%>%dplyr::filter(is_after_first_zero_day==FALSE)

complete_aggr<-aggr_ts %>%group_by(composition,nutrients,temperature,sample_ID)%>%reframe(
  avg_w_sumslopes=mean(sum_w_slopes,na.rm = T),
  abs_sum_slope=mean(abs_sum_slope),
  magnitude=mean(mean_abs_slope),
  symmetry=mean(symmetry),
  sum_abs_slope=mean(sum_abs_slope),
  CV=sd(tot_biomass)/mean(tot_biomass),
  mean_biomass=mean(tot_biomass))

complete_aggr<-full_join(divergence_df,complete_aggr,by=join_by(composition,nutrients,temperature))%>%
  mutate(richness=as.factor(richness))

...

### Effect of T, N and R
```{r boxplots_TNR, results='hide', echo=FALSE, warning=FALSE, fig.align="center", fig.height=8, fig.width=20}
effects of temperature; nutrient and richness, now with complete time series

Boxplot and jitter for temperature
plot_T_complete <- ggplot(data = complete_aggr, aes(x = temperature, y = log10(1 / CV))) +
 geom_boxplot(outlier.shape = NA, aes(group = temperature, color = temperature), alpha = 0.8) +
 geom_jitter(aes(color = as.factor(temperature)), width = 0.2, size = 4, alpha = 0.65) +
 theme_classic(base_size = 25) +
 xlab("Temperature regime") +
 labs(color = "Temperature regime", tag = "(b)") +
 theme(legend.position = "none", axis.title.y = element_blank()) +
 scale_color_viridis_d(option = "inferno", begin = 0.2, end = 0.8) +
 scale_y_continuous(
 breaks = log10(c(0.1, 0.2, 0.4, 0.8, 1.6, 3.2)),
 labels = c("0.1", "0.2", "0.4", "0.8", "1.6", "3.2")
)

Boxplot and jitter for nutrients
plot_N_complete <- ggplot(data = complete_aggr, aes(x = nutrients, y = log10(1 / CV))) +
 geom_boxplot(outlier.shape = NA, aes(group = nutrients, color = nutrients), alpha = 0.8) +
 geom_jitter(aes(color = as.factor(nutrients)), width = 0.2, size = 4, alpha = 0.65) +
 theme_classic(base_size = 25) +
 xlab("Nutrients") +
 theme(legend.position = "none", axis.title.y = element_blank()) +
 scale_color_viridis_d(option = "inferno", begin = 0.2, end = 0.8) +
 scale_y_continuous(
 breaks = log10(c(0.1, 0.2, 0.4, 0.8, 1.6, 3.2)),

```

```

 labels = c("0.1", "0.2", "0.4", "0.8", "1.6", "3.2")
) +
labs(tag = "(c)")

Boxplot and jitter for richness
plot_R_complete <- ggplot(data = complete_aggr, aes(x = richness, y = log10(1 / CV))) +
 geom_boxplot(outlier.shape = NA, aes(group = richness, colour = richness), alpha = 0.8) +
 geom_jitter(aes(color = as.factor(richness)), width = 0.2, size = 4, alpha = 0.65) +
 theme_classic(base_size = 25) +
 xlab("Richness") +
 ylab("Temporal stability\nlog(mean/s.d.)") +
 theme(legend.position = "none") +
 scale_color_viridis_d(option = "inferno", begin = 0.2, end = 0.8) +
 scale_y_continuous(
 breaks = log10(c(0.1, 0.2, 0.4, 0.8, 1.6, 3.2)),
 labels = c("0.1", "0.2", "0.4", "0.8", "1.6", "3.2")
) +
 labs(tag = "(a)")

Combine plots in a single row
combined_plot <- plot_R_complete | plot_T_complete | plot_N_complete

Add global label for the y-axis
combined_plot <- combined_plot

Display the final combined plot
print(combined_plot)

#ggsave("figures_ms/fig.1.png", plot = combined_plot, width = 20, height = 8, dpi = 600)

...

**Figure 2*: Effects of richness (a), temperature (b), and nutrients (c) on community total biomass temporal stability.

```

We can see that richness does not have a clear effect on community temporal stability, while stability was higher at lower temperature, and nutrients increased community temporal stability.

### ### Effect of Divergence

We look at the relationship between divergence (our original response diversity metric) and stability

```

```{r divergence_CV, results='hide', echo=FALSE, warning=FALSE, fig.align="center", fig.height=10, fig.width=16}

plot_CV_divergence<-complete_aggr%>%ggplot(aes(x=divergence,y=log10(1/CV),color=richness))+
  geom_point(size = 3.5, alpha = 0.65)+
  geom_smooth(method="lm",aes(x=divergence,y=log10(1/CV),color=richness),show.legend=F, linewidth = 2)+theme_classic()+
  scale_color_viridis_d(option = "inferno", begin = 0.2, end = 0.8)+
  scale_y_continuous(
    breaks = log10(c(0.1,0.2, 0.4, 0.8, 1.6, 3.2)),
    labels = c("0.1","0.2"," 0.4", "0.8", "1.6", "3.2")
  ) +
  theme(
    axis.title.x = element_text(size = 25),
    axis.title.y = element_text(size = 25),
    axis.text = element_text(size = 14), # Axis text size
    legend.title = element_text(size = 20), # Legend title size
    legend.text = element_text(size = 20), # Legend text size
    strip.background = element_rect(fill = "grey80", color = NA),
    strip.text = element_text(size = 22)) +
  facet_wrap(~richness, labeller = as_labeller(function(x) paste0("richness = ", x)))+
  ylab("Temporal stability\nlog(mean/s.d.)")+
  xlab("divergence")

plot_CV_divergence

...

```

****Figure 3*: Relationship between Divergence and temporal stability of total community biomass.**

Divergence is positively related to temporal stability, suggesting that response diversity promotes stability. However, the relationship between divergence and stability becomes weaker as richness increases. We think that this is due to divergence considering only the responses of the 2 most "responding" species. Thus, when species richness increases, disregarding the responses of the other species in the community except the 2 responding the most makes the relationship between response diversity and stability weaker.

This is why, after running the experiment, we developed another metric to measure the distribution of species' responses, which we called ****imbalance****, and that is presented in the main text of the publication.

Imbalance has several desirable features that make it a more suitable metric than divergence: Independence of richness, higher predictive power, and accounts for the responses of all species in the community (as opposed to divergence that accounts for only the 2 most "responding" species).

Here, we provide extensive evidence of why imbalance is a better metric to measure response diversity than divergence, and thus justifying focusing the analysis around imbalance.

Comparing Divergence and Imbalance

Predictive power of Divergence and Imbalance

We first compare how well divergence and imbalance predict stability (predictive power).

```

```{r results='hide', echo=FALSE, warning=FALSE, message=FALSE}

complete_aggr <- complete_aggr %>% mutate(stability = 1/CV) %>%
 dplyr::rename(balance_f = abs_sum_slope,
 balance_r = avg_w_sumslopes)

summary(lm_full)

```

```

anova(lm_full)
```

### Imbalance
```{r model_check1, fig.align="center", fig.height=9, fig.width=12, include=TRUE, echo=TRUE}
#
mod1 <- lm(data=complete_aggr, log10(stability)~log10(balance_f))

Check model assumptions
#check_model(mod1)
```

### Divergence
```{r model_check2, include=TRUE, echo=TRUE, fig.align="center", fig.height=9, fig.width=12}
mod2 <- lm(data=complete_aggr, log10(stability)~(divergence))

Check model assumptions
#check_model(mod2)
```

**Table 1**: Comparison of model performance of divergence and imbalance as predictors of stability. Model 1 has imbalance as predictor and model 2 has divergence as predictor.
```{r}
compare performance tables of mod1 and mod2 specifying
performance(mod1) %>%
 bind_rows(performance(mod2), .id = "model") %>%
 kable() %>%
 kable_styling(full_width = F)
```

A model with imbalance as predictor performs better than one with divergence as predictor, and it explains more of the variance in stability than divergence.

Moreover, from **Figure 3**, it looks like divergence declines in performance as richness increases. Let's test this analytically. To do that we build a linear model having stability as response variable and either log10(imbalance) or divergence as predictor for each richness level. We then extract the R squared of the models and their *standardised* estimates. (standardized estimates were calculated centering divergence and imbalance using the function scale()).
```{r include = TRUE, echo=TRUE}
getting model estimates for each richness level
lm_divergence_richness_E <- complete_aggr %>%
 nest(data = ~richness) %>%
 mutate(
 model = map(data, ~ lm(log10(stability) ~ scale(divergence), data = .x)),
 results = map(model, broom::tidy)
) %>%
 unnest(results) %>% dplyr::filter(term=="scale(divergence)")

getting model R squared for each richness level
lm_divergence_richness_R <- complete_aggr %>%
 nest(data = ~richness) %>%
 mutate(
 model = map(data, ~ lm(log10(stability) ~ scale(divergence), data = .x)),
 results = map(model, broom::glance)
) %>%
 unnest(results)
```

```{r include = TRUE, echo=TRUE}
getting model estimates for each richness level
lm_balance_richness_E <- complete_aggr %>%
 nest(data = ~richness) %>%
 mutate(
 model = map(data, ~ lm(log10(stability) ~ scale(log10(balance_f)), data = .x)),
 results = map(model, broom::tidy)
) %>%
 unnest(results) %>% dplyr::filter(term=="scale(log10(balance_f))")

getting model R squared for each richness level
lm_balance_richness_R <- complete_aggr %>%
 nest(data = ~richness) %>%
 mutate(
 model = map(data, ~ lm(log10(stability) ~ scale(log10(balance_f)), data = .x)),
 results = map(model, broom::glance)
) %>%
 unnest(results)
```

```{r R_squared, fig.align="center", fig.height=5, fig.width=15}
plot how the r squared changes with richness for divergence and balance with legend and lines with diamond instead of points
a <- ggplot(data = lm_divergence_richness_R, aes(x = richness, y = r.squared, color = "Divergence")) +
 geom_point(shape = 18, cex = 5) +
 geom_point(data = lm_balance_richness_R, aes(x = richness, y = r.squared, color = "Imbalance"), shape = 18, cex = 5) +

```

```

labs(x = "Richness", y = "R squared", color = "Model") +
theme_minimal(base_size = 20) +
scale_color_manual(values = c("Divergence" = "blue", "Imbalance" = "red"))+ labs(tag = "(a)")

plot how the estimates changes with richness for divergence and balance with legend and lines
b <- ggplot(data = lm_divergence_richness_E, aes(x = richness, y = estimate, color = "Divergence")) +
 geom_point(shape = 18, cex = 5) +
 geom_point(data = lm_balance_richness_E, aes(x = richness, y = abs(estimate), color = "Imbalance"), shape = 18, cex = 5) +
 labs(x = "Richness", y = "Estimate", color = "Model") +
 theme_minimal(base_size = 20) +
 scale_color_manual(values = c("Divergence" = "blue", "Imbalance" = "red")) + labs(tag = "(b)")

a + b
```


**Figure 4**: Performance comparison of divergence vs imbalance. In (a), the R squared of linear models for divergence and imbalance are shown for each richness level. In (b), the estimates of the linear models for divergence and imbalance are shown for each richness level.


```

We can see that the R squared of divergence as predictor of stability becomes smaller as richness increases, while the R squared of imbalance as predictor of stability does not (actually increases slightly).

Comparing unique explanatory power of imbalance and divergence

Now we build a linear model where stability is modeled as a function of imbalance and divergence. Then, we compared the variance explained by the full model compared to a model containing either only imbalance or only divergence.

Full model – imbalance and divergence

```

```{r model_check3, fig.align="center", fig.height=9, fig.width=12, include=TRUE, echo=TRUE}
lm_div_balance <- lm(data=complete_aggr, log10(stability)~log10(balance_f)+divergence)

Check model assumptions
check_model(lm_div_balance)
```

```

model with only divergence

```

```{r model_check4, fig.align="center", fig.height=9, fig.width=12, include=TRUE, echo=TRUE}
lm_div <- lm(data=complete_aggr, log10(stability)~divergence)

Check model assumptions
check_model(lm_div)
```

```

model with only imbalance

```

```{r model_check5, fig.align="center", fig.height=9, fig.width=12, include=TRUE, echo=TRUE}
lm_balance <- lm(data=complete_aggr, log10(stability)~log10(balance_f))

Check model assumptions
check_model(lm_balance)
```

```

Comparison full model vs divergence only and imbalance only

****Table 2**:** Comparison of model performance of divergence, imbalance and both as predictors of stability. Model 1 has both imbalance and divergence as predictors, model 2 has divergence as predictor, and model 3 has imbalance as predictor.

```

```{r}
compare performance tables of lm_div_balance, lm_div and lm_balance
performance(lm_div_balance) %>%
 bind_rows(performance(lm_div), .id = "model") %>%
 bind_rows(., performance(lm_balance), .id = "model") %>%
 kable() %>%
 kable_styling(full_width = F)
```

```

Comparison full model vs imbalance only

****Table 3**:** Anova table: a model with both imbalance and divergence as predictors is not significantly different from a model with only imbalance as predictor.

```

```{r include = TRUE, echo=TRUE}
anova1 <- anova(lm_div_balance, lm_balance)

Convert to tidy format
anova_tidy1 <- broom::tidy(anova1)
Display the tidy ANOVA table using gt with formatted p-values and adjusted size
anova_tidy1 %>%
 gt() %>%
 cols_label(
 term = "Term",
 sumsq = "Sum of Squares",
 df = "DF",
 statistic = "F Statistic",
 p.value = "p-value"
) %>%
 fmt_number(
 columns = vars(p.value),
 decimals = 3
) %>%

```

```

 tab_style(
 style = cell_text(weight = "bold"),
 locations = cells_body(
 columns = vars(p.value),
 rows = p.value < 0.05
)
) %>%
 tab_options(
 table.width = px(600), # Adjust table width (e.g., 400px)
 table.font.size = px(12), # Adjust font size (e.g., 12px)
 data_row.padding = px(10) # Adjust row padding (e.g., 4px for more compact rows)
)

...

Comparision full model vs imbalance only and divergence only

Table 4: Anova table: a model with both imbalance and divergence as predictors is significantly better from a model with only
divergence as predictor.
```{r include = TRUE, echo=TRUE}
anova2 <- anova(lm_div_balance, lm_div)

anova_tidy2 <- broom::tidy(anova2)
# Display the tidy ANOVA table using gt with formatted p-values and adjusted size
anova_tidy2 %>%
  gt() %>%
  cols_label(
    term = "Term",
    sumsq = "Sum of Squares",
    df = "DF",
    statistic = "F Statistic",
    p.value = "p-value"
  ) %>%
  fmt_number(
    columns = vars(p.value),
    decimals = 3
  ) %>%
  tab_style(
    style = cell_text(weight = "bold"),
    locations = cells_body(
      columns = vars(p.value),
      rows = p.value < 0.05
    )
  ) %>%
  tab_options(
    table.width = px(600),          # Adjust table width (e.g., 400px)
    table.font.size = px(12),       # Adjust font size (e.g., 12px)
    data_row.padding = px(10)      # Adjust row padding (e.g., 4px for more compact rows)
  )

...

Overall, imbalance explains more of the variance in stability than divergence, and there is virtually no difference between a model
containing only imbalance and the full model.

## Interaction divergence and richness

Richness had to be transformed to numeric and to be centered to avoid collinearity with divergence
```{r model_check6, fig.align="center", fig.height=9, fig.width=12, include=TRUE, echo=TRUE}

lm_rich_div <- lm(data=complete_aggr, log10(stability)~divergence*scale(as.numeric(richness)))

check model assumptions
check_model(lm_rich_div)
```

**Table 5**: Type III anova table of the model with divergence and richness as predictors of stability.
```{r include = TRUE, echo=TRUE}
anova3 <- car::Anova(lm_rich_div, type = "III")

anova_tidy3 <- broom::tidy(anova3)
Display the tidy ANOVA table using gt with formatted p-values and adjusted size
anova_tidy3 %>%
 gt() %>%
 cols_label(
 term = "Term",
 sumsq = "Sum of Squares",
 df = "DF",
 statistic = "F Statistic",
 p.value = "p-value"
) %>%
 fmt_number(
 columns = vars(p.value),
 decimals = 3
) %>%
 tab_style(
 style = cell_text(weight = "bold"),
 locations = cells_body(
 columns = vars(p.value),
 rows = p.value < 0.05
)
) %>%
 tab_options(

```

```

 table.width = px(600), # Adjust table width (e.g., 400px)
 table.font.size = px(12), # Adjust font size (e.g., 12px)
 data_row.padding = px(10) # Adjust row padding (e.g., 4px for more compact rows)
)
}

```

Divergence significantly interact with richness, suggesting that the relationship between divergence and stability changes with richness.  
While an ideal metric of response diversity should be independent of richness.

```

We repeat the same model using imbalance instead of divergence.
```{r model_check7, fig.align="center", fig.height=9, fig.width=12, include=TRUE, echo=TRUE}
lm_rich_balance <- lm(data=complete_aggr, log10(stability)~log10(balance_f)*scale(as.numeric(richness)))

# check model assumptions
# check_model(lm_rich_balance)
```

```

```

Table 6: Type III anova table of the model with imbalance and richness as predictors of stability.
```{r include = TRUE, echo=TRUE}
anova4 <- car::Anova(lm_rich_balance, type = "III")

anova_tidy4 <- broom::tidy(anova4)
# Display the tidy ANOVA table using gt with formatted p-values and adjusted size
anova_tidy4 %>%
  gt() %>%
  cols_label(
    term = "Term",
    sumsq = "Sum of Squares",
    df = "DF",
    statistic = "F Statistic",
    p.value = "p-value"
  ) %>%
  fmt_number(
    columns = vars(p.value),
    decimals = 3
  ) %>%
  tab_style(
    style = cell_text(weight = "bold"),
    locations = cells_body(
      columns = vars(p.value),
      rows = p.value < 0.05
    )
  ) %>%
  tab_options(
    table.width = px(600),          # Adjust table width (e.g., 400px)
    table.font.size = px(12),       # Adjust font size (e.g., 12px)
    data_row.padding = px(10)      # Adjust row padding (e.g., 4px for more compact rows)
  )
}

```

Imbalance does not significantly interact with richness, suggesting that the relationship between imbalance and stability is stable across richness levels.

Variable importance

Finally, we assess variable importance using the relative importance of predictors in the full model. We use the package vip (<https://cran.r-project.org/web/packages/vip/vignettes/vip.html>) to calculate the relative importance of predictors in the full model. The function vip::vip for multiple linear regression, or linear models (LMs), uses the absolute value of the t -statistic as a measure of VI. Motivation for the use of the associated t -statistic is given in Bring (1994) [<https://www.tandfonline.com/doi/abs/10.1080/00031305.1994.10476059>].

```

```{r vip1, fig.align="center", fig.height=4, fig.width=8, include=TRUE, echo=TRUE}
vip::vip(lm_div_balance)
```

```

Figure 5: Variable importance in the model including both imbalance and divergence as predictors of stability.

We believe that the extensive evidence here provided justifies focusing the analysis around imbalance, and not divergence, as a metric of response diversity.
We will thus only look at imbalance for the rest of the analysis.

Effect RD

We are now going to look at how imbalance affected temporal stability of total community biomass. We are going to look at the relationship between fundamental imbalance (so based only on species response surfaces measured in monoculture), an realised imbalance (measured accounting for species contribution to balance).

This is fundamentally testing our most important hypothesis.

```

```{r effect_RD, results='hide', echo=FALSE, warning=FALSE, fig.align="center", fig.height=10, fig.width=20}

```

```

balance_dd<-complete_aggr%>%dplyr::select(balance_f,richness,CV)%>%
 cbind(type="fundamental")%>%mutate(balance_f=balance_f/as.numeric(richness))
names(balance_dd)[names(balance_dd)=="balance_f"]<-"balance"

```

```

weighted_balance_dd<-complete_aggr%>%dplyr::select(balance_r,richness,CV)%>%
 cbind(type="realised")
names(weighted_balance_dd)[names(weighted_balance_dd)=="balance_r"]<-"balance"

```

```

main_r_dd<-rbind(balance_dd,weighted_balance_dd)

```



```

plot_main <- main_r_dd %>%
 ggplot(aes(x = (log10(balance)), y = log10(1/CV), color = type)) +
 geom_point(aes(x = log10(balance), y = log10(1/CV)), size = 5, alpha = 0.65) +
 geom_smooth(method = "lm", aes(x = log10(balance), y = log10(1/CV)), linewidth = 3, show.legend=F) +
 theme_classic() +
 scale_color_viridis_d(option = "inferno", begin = 0.3, end = 0.6) +
 ylab("Temporal stability\nlog(mean/s.d.)")+
 xlab("log10(Imbalance)") +
 theme(
 axis.title.x = element_text(size = 25),
 axis.title.y = element_text(size = 25),
 axis.text = element_text(size = 20), # Axis text size
 legend.title = element_text(size = 25), # Legend title size
 legend.text = element_text(size = 25), # Legend text size
 strip.background = element_rect(fill = "grey80", color = NA),
 strip.text = element_text(size = 25)) +
 labs(color = "Imbalance") +
 facet_wrap(~richness, labeller = as_labeller(function(x) paste0("richness = ", x)))

plot_main

#gggsave("figures_ms/fig.2.png", plot = plot_main, width = 20, height = 10, dpi = 600)

Figure 6: Effects of fundamental and realised imbalance on total community biomass temporal stability.

```

We can see that imbalance is always negatively related to temporal stability, which means that balance in species responses promotes stability across richness levels. Interestingly, we see that there is little difference between fundamental and realised imbalance. Yet, as the richness increases, the relationship between realised imbalance and stability becomes steeper compared to fundamental balance.

But is the difference between fundamental and realised imbalance significant? We can test this using a linear model with both fundamental and realised imbalance as predictors of stability, and one with only fundamental imbalance as predictor of stability, and compare whether the models are significantly different.

```

Imbalance: realised vs fundamental
```{r include=TRUE, echo=TRUE}
# compare if the slope of fundamental and realised balance is significantly different for each richness level
# Fit the linear model with interaction
complete_aggr_2<- complete_aggr %>%
  # Remove the units from the 'nutrients' and 'temperature' columns
  mutate(
    nutrients = as.numeric(gsub(" g/L", "", nutrients)), # Convert nutrients to numeric
    temperature = gsub(" °C", "", temperature) # Remove the unit but keep as character
  ) %>%
  # Convert temperature ranges to numeric codes using case_when
  mutate(
    temperature = case_when(
      temperature == "18-21" ~ 1,
      temperature == "22-25" ~ 2,
      temperature == "25-28" ~ 3,
      TRUE ~ NA_real_ # Handle unexpected values with NA
    )
  )

# Fit the linear model with interaction
lm_full_int1<-lm(data=complete_aggr_2,log10(stability)~log10(balance_f)+scale(nutrients)*scale(temperature)+richness)
lm_full_int2<-lm(data=complete_aggr_2,log10(stability)~log10(balance_f)+
log10(balance_r)+scale(nutrients)*scale(temperature)+richness)
...

**Table 7**: Anova table of the model with only realised balance vs one with both realised and fundamental balance as predictors of stability.
```{r include = TRUE, echo=TRUE}
anova5 <- anova(lm_full_int1, lm_full_int2, test = "Chisq")

```

```

anova_tidy5 <- broom::tidy(anova5)
Display the tidy ANOVA table using gt with formatted p-values and adjusted size
anova_tidy5 %>%
 gt() %>%
 cols_label(
 term = "Term",
 df.residual = "DF",
 rss = "RSS",
 sumsq = "sumsq",
 p.value = "p-value"
) %>%
 fmt_number(
 columns = vars(p.value),
 decimals = 3
) %>%
 tab_style(
 style = cell_text(weight = "bold"),
 locations = cells_body(
 columns = vars(p.value),
 rows = p.value < 0.05
)
) %>%
 tab_options(
 table.width = px(600), # Adjust table width (e.g., 400px)
 table.font.size = px(12), # Adjust font size (e.g., 12px)
 data_row.padding = px(10) # Adjust row padding (e.g., 4px for more compact rows)
)

```

...

A model with both fundamental and realised imbalance as predictors improved very little the variance explained by the model. The two models are not significantly different, suggesting that fundamental imbalance captures well the effect of imbalance on stability, and adding the species contribution to total biomass (realised imbalance) does not improve the model.

We now compare also the AIC of the two models

```
```{r include=TRUE, echo=TRUE}
AIC(lm_full_int1, lm_full_int2)
```
```

The AIC of the model with only fundamental imbalance is lower than the AIC of the model with both fundamental and realised imbalance, suggesting that the model with only fundamental imbalance is a better model. However, the difference is very small.

```
```{r}

# Model with balance_f only
lm_balance_f <- lm(data = complete_aggr_2,
  log10(stability) ~ log10(balance_f) + scale(nutrients)*scale(temperature) + richness)

# Model with balance_r only
lm_balance_r <- lm(data = complete_aggr_2,
  log10(stability) ~ log10(balance_r) + scale(nutrients)*scale(temperature) + richness)

...

<!-- # ```{r} -->
<!-- # # Compare AIC -->
<!-- # AIC(lm_balance_f, lm_balance_r, lm_full_int1, lm_full_int2) -->
<!-- # -->
<!-- # # Compare adjusted R² -->
<!-- # summary(lm_balance_f)$adj.r.squared -->
<!-- # -->
<!-- # summary(lm_balance_r)$adj.r.squared -->
<!-- # -->
<!-- # # ANOVA comparison (nested models) -->
<!-- # anova(lm_balance_f, lm_full_int2) -->
<!-- # anova(lm_balance_r, lm_full_int2) -->
<!-- # ``` -->

<!-- # ```{r} -->
<!-- # # Full model with both variables -->
<!-- # lm_both <- lm(data = complete_aggr_2, -->
<!-- #       log10(stability) ~ log10(balance_f) + log10(balance_r) + scale(nutrients)*scale(temperature) + richness) -->
<!-- # -->
<!-- # # Get R² values -->
<!-- # r2_f <- summary(lm_balance_f)$r.squared -->
<!-- # r2_r <- summary(lm_balance_r)$r.squared -->
<!-- # r2_both <- summary(lm_both)$r.squared -->
<!-- # -->
<!-- # # Shared variance -->
<!-- # shared_variance <- r2_both - (r2_f + r2_r) -->
<!-- # shared_variance -->
<!-- # -->
<!-- # ``` -->

<!-- # -->

<!-- # -->

<!-- # Example: Using the R-squared values from the models -->
<!-- r2_f <- summary(lm_balance_f)$r.squared # R² for balance_f model -->
<!-- r2_r <- summary(lm_balance_r)$r.squared # R² for balance_r model -->
<!-- r2_both <- summary(lm_both)$r.squared # R² for model with both variables -->

<!-- # Calculate shared variance -->
<!-- shared_variance <- r2_both - (r2_f + r2_r) -->
<!-- total_variance <- r2_both -->

<!-- # Percentage of shared variance -->
<!-- shared_percentage <- (shared_variance / total_variance) * 100 -->
<!-- shared_percentage -->

<!-- # -->

# Linear models

## Dependence of Imbalance on Temperature

Since imbalance depends on the slopes of species' responses, and the slopes of species responses between two environmental conditions depend on the shape of species responses and the position of the optimum of the species response curve, we may expect that imbalance depends on temperature. For example, when the environment (i.e. temperature) fluctuates at higher mean temperatures, the species responses may be more similar (i.e. all species have the same sign of response), leading to less symmetrical response distributions, and thus higher imbalance.

We look now at how imbalance changes with temperature, in all possible combinations of species responses for each richness level and each environmental condition. We calculate the sum of slopes of all possible combinations of species responses, and then calculate the imbalance for each combination. We then plot the relationship between imbalance and temperature for each combination of species responses.

```{r}
```

```
Function to calculate combinations and their sum of slopes
calculate_compositions <- function(data) {
 # Generate combinations for n = 2, n = 3, n = 4
 combinations <- lapply(2:4, function(n) combn(data$predict_spec, n, simplify = FALSE))
 combinations <- unlist(combinations, recursive = FALSE)
}
```

```

Create a dataframe with combinations and their summed slopes
result <- data.frame(
 composition = sapply(combinations, paste, collapse = ""),
 sum_slopes = sapply(combinations, function(comb) {
 sum(data$slope[data$predict_spec %in% comb])
 }),
 magnitude=sapply(combinations, function(comb) {
 sum(abs(data$slope[data$predict_spec %in% comb]))
 })
)
return(result)
}

Apply the function for each group of `nutrients` and `temperature`
all_combi_df <- df_slopes %>%
 group_by(nutrients, temperature) %>%
 group_split() %>%
 lapply(function(group_data) {
 compositions <- calculate_compositions(group_data)
 # Add nutrients and temperature columns to each result
 compositions$nutrients <- unique(group_data$nutrients)
 compositions$temperature <- unique(group_data$temperature)
 return(compositions)
 }) %>%
 bind_rows()

all_combi_df<-all_combi_df%>%mutate(balance=abs(sum_slopes),
 richness=as.factor(nchar(composition)))

all_combi_df<-all_combi_df%>%mutate(is_selected=all_combi_df$balance%in%complete_aggr$balance_f)

test_representative<-
t.test(log10(dplyr::filter(all_combi_df,is_selected==F)$balance),log10(dplyr::filter(all_combi_df,is_selected==T)$balance),var.equal
=F)

lm_representative<-lm(data=all_combi_df,log10(balance)~temperature*is_selected)
#check_model(lm_representative)
```



```

```{r effect_T, results='hide', echo=FALSE, warning=FALSE, fig.align="center", fig.height=10, fig.width=20}
plot_all_combi<-ggplot(data=all_combi_df)+theme_bw(base_size = 25)+
  geom_quasirandom(data= all_combi_df,aes(y=log10(balance), x=temperature, group=temperature, colour=as.factor(temperature)),
    dodge.width=2, size= 6) +

  xlab("Temperature regime")+
  ylab("log10(balance)")+
  labs(color="Temperature regime", tag = "(a)")+
  scale_color_viridis_d(option = "inferno", begin = 0.2, end = 0.8)+
  facet_wrap(~is_selected,labeller = as_labeller(c("FALSE"="Not Selected","TRUE"="Selected")))+
  theme(legend.position = "none", strip.text = element_text(size = 25),
    strip.background = element_rect(fill = "grey80", color = NA))

plot_all_combi
```

Figure 7: Relationship between imbalance and temperature for all possible combinations of species responses. The points
represent the imbalance of each combination of species responses. The points are colored by temperature regime, and the facets
represent whether the combination of species responses was selected for the experiment.

We can see that imbalance increases with temperature, and that the relationship is the same in selected vs non selected community
compositions. This suggests that imbalance is intrinsically dependent on temperatures, and that the species responses are more
similar when the environment fluctuates at higher temperatures.
Inevitably, this will affect the stability of the community, as we have seen in the previous analysis. Yet, it also means that the
effect of imbalance on stability is not independent of temperature, or more generally, of the environmental conditions. Indeed,
temperature and nutrients have been used to estimate species' responses, and thus imbalance inherently depends on the environment.

Let's look at the variance explained by imbalance and the environment on stability.
```{r Rsquared, fig.align="center", fig.height=6, fig.width=12}
lm_balance_env<-lm(data=complete_aggr_2,log10(stability)~log10(balance_f)+scale(nutrients)*scale(temperature))
lm_balance<-lm(data=complete_aggr_2,log10(stability)~log10(balance_f))
lm_env<-lm(data=complete_aggr_2,log10(stability)~scale(nutrients)*scale(temperature))

sum_lm_balance<-summary(lm_balance)
sum_lm_env<-summary(lm_env)
sum_lm_balance_env<-summary(lm_balance_env)

R2_env_bal<-data.frame(R2=c(sum_lm_balance$r.squared,sum_lm_env$r.squared,sum_lm_balance_env$r.squared,
  sum_lm_balance_env$r.squared-sum_lm_env$r.squared,
  sum_lm_balance_env$r.squared-sum_lm_balance$r.squared,
  0),
  model=factor(c("log10(balance)","nut * temp","log10(balance) + nut * temp","log10(balance)","nut *
temp","log10(balance) + nut * temp")),
  type=c("total","total","total","unique","unique","unique"))

plot_R2_bal_env<-ggplot(data=R2_env_bal%>%dplyr::filter(type=="total"),aes(y=R2,x=factor(model,levels=c("log10(balance)","nut *
temp","log10(balance) + nut * temp")),fill=type))+
  geom_col(width=0.7)+
  geom_col(data=R2_env_bal%>%dplyr::filter(type=="unique"),width=0.7)+
  theme_classic()+
  theme(axis.title.x = element_text(size = 18),
    axis.title.y = element_text(size = 18),
    axis.text = element_text(size = 15),
    strip.background = element_rect(fill = "grey80", color = NA),
    strip.text = element_text(size = 18),plot.tag = element_text(size = 18),
    legend.text = element_text(size = 15), # Adjust legend text size
    legend.title = element_blank() ) +
  xlab("Model terms") +

```


```

```
ylab(expression(R^2))+
scale_fill_manual(values = c("total" = "grey80", "unique" = "grey20"))+labs(tag="(a)")
plot_R2_bal_env
```

**\*\*Figure 8\*\*:** Variance explained by imbalance and the environment on stability. The total variance explained by imbalance and the environment is shown in grey, and the unique variance explained by imbalance and the environment is shown in light grey.

95% of the variance explained by balance is shared with nutrients and temperature, which is not surprising considering how imbalance is calculated. However, there is additional variance explained by the environment that is not shared with balance. This suggests that the environment has an independent effect on stability, which is not mediated by imbalance. This may be related to the fact that imbalance was calculated using monoculture data collected at constant temperature, whereas in the community experiment, temperature was fluctuating, and species were interacting.

To disentangle the shared variance between imbalance, temperature, and nutrients, we first removed the variance in temperature and nutrients that could be explained by imbalance. This can be done by regressing temperature and nutrients on imbalance and extracting the residuals, a method previously used to isolate independent effects among collinear predictors (e.g., [Dormann et al. 2013] (<https://nsojournals.onlinelibrary.wiley.com/doi/full/10.1111/j.1600-0587.2012.07348.x>); [Graham 2003] ([https://esajournals.onlinelibrary.wiley.com/doi/full/10.1890/02-3114?casa\\_token=tjLz0BPcLf4AAAAA%3AYjunSb9x1IaHns00YLwGDjWzZswdTkV0WfcsBfIC3WjrA-n5B\\_94c0P2LFP5WIXMhunDwZOPFXCQgg](https://esajournals.onlinelibrary.wiley.com/doi/full/10.1890/02-3114?casa_token=tjLz0BPcLf4AAAAA%3AYjunSb9x1IaHns00YLwGDjWzZswdTkV0WfcsBfIC3WjrA-n5B_94c0P2LFP5WIXMhunDwZOPFXCQgg))). The residual temperature and nutrient values thus represented variation independent of imbalance, allowing us to test their effects on stability without confounding influences of shared variance with imbalance. This approach enabled us to evaluate the relative importance of species richness and imbalance while accounting for environmental variability. By including residual temperature and nutrient values in the model, we ensured that our analysis focused on their independent contributions to stability.

```
Control effect of environment on balance
```

```
`r`
lm_temp_resid<-lm(data=complete_aggr_2, temperature ~ log10(balance_f))

lm_nut_resid<-lm(data=complete_aggr_2, nutrients ~ log10(balance_f))

resid_temp<-lm_temp_resid$residuals

resid_nut<-lm_nut_resid$residuals

complete_aggr_2<-complete_aggr_2%>%mutate(resid_temp=resid_temp,resid_nut=resid_nut)

`r`
```

Now, we can build a linear model with imbalance, richness, the residuals of temperature and nutrients and their interaction as predictors of stability. We also use composition as random effect to account for some compositions being used in more than one treatment combination.

We also want to test whether the inclusion of random slopes for  $\log_{10}(\text{balance}_f)$  improves the model's ability to explain variability in  $\log_{10}(\text{stability})$  among compositions. We will compare the model with random slopes for  $\log_{10}(\text{balance}_f)$  to the model without random slopes for  $\log_{10}(\text{balance}_f)$  using a Likelihood Ratio Test.

```
`r`

Fit the mixed-effects model without random effects
model_no_random <- lmer(log10(stability) ~ log10(balance_f) + richness + resid_temp * resid_nut+
 (1 | composition), data = complete_aggr_2, REML = FALSE)

Fit the mixed-effects model with random effects
model_with_random <- lmer(log10(stability) ~ log10(balance_f) + resid_temp*resid_nut+ richness +
 (1 + log10(balance_f) | composition), data = complete_aggr_2, REML = FALSE)

Compare the models
anova(model_no_random, model_with_random)

`r`
```

A model with random slopes for  $\log_{10}(\text{balance}_f)$  is not better than one without. Thus, we will use the model without random slopes for  $\log_{10}(\text{balance}_f)$  for the rest of the analysis.

```
`r` model_check_int, fig.cap='model check 1.', fig.align="center", fig.height=9, fig.width=12, include=TRUE, echo=TRUE)
check model's assumptions
check_model(model_no_random)

`r`
```

**\*\*Figure 9\*\*:** Model check for the linear mixed-effects model with balance, richness, and the residuals of temperature and nutrients as predictors of stability. The model fits well the data.

**\*\*Table 10\*\*:** Linear mixed-effects model results for the effects of balance, richness, and the residuals of temperature and nutrients on community stability. Estimates are presented with 95% confidence intervals and p-values.

```
`r`
Create a gtsummary table

tbl_summary <- gtsummary::tbl_regression(model_no_random, intercept = TRUE,
 label = list("(Intercept)" = "Intercept",
 log_balance = "log10(balance)",
 nut_scaled = "nutrients (scaled)",
 temp_scaled = "temperature (scaled)",
 "nut_scaled*temp_scaled" = "nutrients x temperature (scaled)"),
 add_estimate_to_reference_row = TRUE, # Adds reference values for categorical predictors
 add_pairwise_contrasts = TRUE,
 add_significance_stars = TRUE) %>%

gtsummary::modify_table_body(
 ~ .x %>%
 dplyr::mutate(
 t_value = ifelse(!is.na(estimate), round(statistic, 2), NA), # Use 'statistic' for t-value
 # Remove std_error, as it is no longer needed
```

```

)
 std_error = NULL
)
) %>%
Reorder columns: estimate, 95% CI, t-value, p-value
gtsummary::modify_header(label = "***Predictor***",
 estimate = "***Estimate***",
 conf.int = "***95% CI***",
 t_value = "***t-value***",
 p.value = "***p-value***") %>%
gtsummary::modify_footnote(everything() ~ NA) %>% # Remove default footnotes for clean look
gtsummary::bold_labels() %>%
gtsummary::as_gt() %>%
tab_header(title = "Linear Regression Results") %>%
Ensure p-value is the last column
gt::cols_move(p.value, after = last_col()) %>%
Bold significant p-values
tab_style(
 style = cell_text(weight = "bold"),
 locations = cells_body(
 columns = vars(p.value),
 rows = p.value < 0.05
)
)

```

```
tbl_summary
```

```
...
```

The linear mixed-effects model was fitted by maximum likelihood (ML) with log-transformed stability as the response variable. The fixed effects included  $\log_{10}(\text{balance\_f})$ , richness, the residuals of temperature ( $\text{resid\_temp}$ ), and nutrients ( $\text{resid\_nut}$ ), as well as their interaction term. The random effect was specified for the composition group, which accounted for potential variability between different composition categories.

The model's results indicated that  $\log_{10}(\text{balance\_f})$  (Estimate =  $-0.078$ , SE =  $0.0079$ ,  $p < 2e-16$ ),  $\text{resid\_temp}$  (Estimate =  $-0.099$ , SE =  $0.0093$ ,  $p < 2e-16$ ), and  $\text{resid\_nut}$  (Estimate =  $0.248$ , SE =  $0.018$ ,  $p < 2e-16$ ) all had significant effects on stability, with both  $\text{resid\_temp}$  and  $\text{resid\_nut}$  showing large, statistically significant relationships with stability. Additionally, the interaction term  $\text{resid\_temp}:\text{resid\_nut}$  (Estimate =  $-0.091$ , SE =  $0.031$ ,  $p = 0.0037$ ) was also significant, indicating that the combined effects of temperature and nutrients on stability are not independent, but interact in shaping stability.

Richness levels ( $\text{richness3}$  and  $\text{richness4}$ ) did not show significant effects on stability ( $p > 0.77$ ), suggesting that, within the context of this model, variations in richness do not explain substantial variation in community stability.

**\*\*Table 11\*\*:** ANOVA table of the linear mixed-effects model with balance, richness, and the residuals of temperature and nutrients as predictors of stability.

```

```{r}
anova_F <- car::Anova(model_no_random, type = "III")
# Convert to tidy format
anova_tidyF <- broom::tidy(anova_F)
# Display the tidy ANOVA table using gt with formatted p-values and adjusted size
anova_tidyF %>%
  gt() %>%
  cols_label(
    term = "Term",
    df = "DF",
    statistic = "F Statistic",
    p.value = "p-value"
  ) %>%
  fmt_number(
    columns = vars(p.value),
    decimals = 3
  ) %>%
  tab_style(
    style = cell_text(weight = "bold"),
    locations = cells_body(
      columns = vars(p.value),
      rows = p.value < 0.05
    )
  ) %>%
  tab_options(
    table.width = px(600), # Adjust table width (e.g., 400px)
    table.font.size = px(20), # Adjust font size (e.g., 12px)
    data_row.padding = px(10) # Adjust row padding (e.g., 4px for more compact rows)
  )
```

```

# Asynchrony

Response diversity (one of the stabilising effects captured by imbalance) has been suggested as a mechanism that promotes temporal stability of community biomass by promoting species asynchrony.

We thus calculated the asynchrony index suggested by [Gross et al. 2014](<https://www.journals.uchicago.edu/doi/epdf/10.1086/673915>) to calculate the effect of asynchrony on temporal stability and to see how response diversity relate to asynchrony. The index ranges between  $-1$  and  $1$ , with  $-1$  indicating perfect asynchrony and  $1$  being perfectly synchronous, and  $0$  indicating random variation.

```
```{r echo=F, warning=FALSE, results='hide'}
```

```
#filter out completely extinct microcosms
```

```
async_df<-complete_ts%>%dplyr::mutate(rep_var=sub("_[^_]+$", "", sample_ID))
```

```
async_Gross <- async_df %>% group_by(composition,sample_ID) %>%
  do(synchrony_Gross = synchrony(., "day", "predict_spec",
    "biomass", metric = "Gross"))
```

```
async_Gross<-dplyr::mutate(async_Gross,synchrony_Gross=synchrony_Gross%>%unlist())
```

```

async_Loroeau <- async_df %>% group_by(composition,sample_ID) %>%
  do(synchrony_L = synchrony(., "day", "predict_spec",
    "biomass", metric = "Loroeau"))
async_Loroeau<-dplyr::mutate(async_Loroeau,synchrony_Loroeau=synchrony_L%>%unlist())

async_aggr<-full_join(async_Gross,complete_aggr,join_by(sample_ID,composition))

async_aggr<-full_join(async_aggr,async_Loroeau,join_by(sample_ID,composition))

async_aggr_f<-async_aggr%>%dplyr::filter(nutrients!="0.01 g/L")
async_aggr<-async_aggr%>%dplyr::mutate(rep_var=sub("_[^_]+$", "", sample_ID))

...

### Plot stability vs. Asynchrony Gross
```{r async_plots, fig.align="center", fig.height=6, fig.width=12}
plot_async_CV_G<-ggplot(data=async_aggr,aes(x=-synchrony_Gross,y=log10(1/CV),color=nutrients))+
 geom_point(size = 3.5, alpha = 0.65)+
 geom_smooth(method="lm",show.legend=F, linewidth = 2)+
 theme_classic()+
 scale_y_continuous(
 breaks = log10(c(0.1,0.2, 0.4, 0.8, 1.6, 3.2)),
 labels = c("0.1","0.2"," 0.4", "0.8", "1.6", "3.2")
) +
 theme(axis.title.x = element_text(size = 25),
 axis.title.y = element_text(size = 25),
 axis.text = element_text(size = 20),
 strip.background = element_rect(fill = "grey80", color = NA),
 strip.text = element_text(size = 25)) +
 ylab("Temporal stability\\nlog(mean/s.d.)")+
 xlab("Asynchrony")+
 scale_color_viridis_d(option = "inferno", begin = 0.2, end = 0.8)

plot_async_CV_G
...

Figure 10: Relationship between temporal stability and asynchrony (Gross) divided by nutrient level.

Plot Asynchrony Gross vs fundamental imbalance
```{r async, fig.align="center", fig.height=6, fig.width=12}
plot_async_B_G<-ggplot(data=async_aggr,aes(x=log10(balance_f),y=-synchrony_Gross,color=nutrients))+
  geom_point(size = 3.5, alpha = 0.65)+
  geom_smooth(method="lm",show.legend=F, linewidth = 2)+
  theme_classic()+
  theme(axis.title.x = element_text(size = 25),
    axis.title.y = element_text(size = 25),
    axis.text = element_text(size = 20),
    strip.background = element_rect(fill = "grey80", color = NA),
    strip.text = element_text(size = 25)) +
  xlab(expression(log10(Imbalance)["fund"] )) +
  ylab("Asynchrony")+
  scale_color_viridis_d(option = "inferno", begin = 0.2, end = 0.8)

plot_async_B_G
...

**Figure 11**: Relationship between asynchrony (Gross) and fundamental imbalance divided by nutrient level.

<!-- ```{r fig.align="center", fig.height=12, fig.width=16} -->

<!-- plot_async_CV_G <- plot_async_CV_G + -->
<!-- labs(tag = "(a)") + -->
<!-- theme(plot.tag = element_text(size = 20))+ theme(legend.position = "none") -->

<!-- plot_async_B_G <- plot_async_B_G + -->
<!-- labs(tag = "(b)") + -->
<!-- theme(plot.tag = element_text(size = 20)) + theme(legend.position = "none") -->

<!-- plot_pop_aggr_balance <- plot_pop_aggr_balance + -->
<!-- labs(tag = "(c)") + -->
<!-- theme(plot.tag = element_text(size = 20)) -->

<!-- plot_pop_balance <- plot_pop_balance + -->
<!-- labs(tag = "(d)") + -->
<!-- theme(plot.tag = element_text(size = 20))+ theme(legend.position = "none") -->

<!-- # Combine plots and share the legend -->
<!-- fig3 <- (plot_async_CV_G + plot_async_B_G) / -->
<!-- (plot_pop_aggr_balance + plot_pop_balance) + -->

```

```

<!--      plot_layout(guides = "collect") -->
<!-- fig3 -->

<!-- # ggsave("figures_ms/fig.3.png", plot = fig3, width = 18, height = 12, dpi = 600) -->
<!-- `` -->

## Evenness
Evenness in species biomass has been identified as an important factor potentially influencing ecosystem stability [Thibaut & Connolly 2013](https://onlinelibrary.wiley.com/doi/full/10.1111/ele.12019). In the context of our experiment, evenness in species biomass could help explaining why there is little difference between fundamental and realized imbalance. If evenness is high, then all species contribute similarly to total biomass. In this case, weighting for species-biomass contribution to total biomass (realized), should not fundamentally change the result, compared to an un-weighted (fundamental) measurement.
```{r evenness, fig.align="center", fig.height=6, fig.width=8}

even_aggr<-
complete_ts%>%group_by(sample_ID,predict_spec,richness)%>%summarize(mean_biom_contribution=mean(biom_contribution,na.rm=T))%>%ungroup()

exclude microcosms where all but one species
even_aggr<-even_aggr%>%dplyr::filter(!(mean_biom_contribution%in%c(0,1)))

even_aggr<-even_aggr%>%group_by(sample_ID,richness)%>%reframe(H= -sum(mean_biom_contribution*log(mean_biom_contribution)))
even_aggr<-even_aggr%>%mutate(evenness=H/log(as.numeric(richness)))

even_aggr<-full_join(even_aggr,async_aggr,by=c("sample_ID"))

mean_evenness <- mean(even_aggr$evenness, na.rm = TRUE)

Create the histogram with the mean indicated
dist_even <- ggplot(data = even_aggr, aes(x = evenness)) +
 geom_histogram(binwidth = 0.05, fill = "steelblue", color = "black", alpha = 0.7) + # Adjust binwidth for better visualization
 geom_vline(aes(xintercept = mean_evenness), color = "red", linetype = "dashed", size = 1.2) + # Add the mean line
 theme_classic() +
 labs(
 x = "Evenness",
 y = "Frequency",
 title = "Distribution of Evenness with Mean"
) +
 annotate("text", x = mean_evenness, y = 5, label = paste0("Mean = ", round(mean_evenness, 2)),
 color = "red", hjust = -0.1, size = 5)

Print the plot
dist_even
```

**Figure 12**:: Distribution of species evenness across experimental communities. The histogram represents the frequency of observed evenness values, while the red dashed line indicates the mean evenness (0.7). This highlights the central tendency of evenness across the dataset and its variation among communities.

Evenness was indeed generally high in our experimental communities, suggesting another potential factor reducing the potential difference between fundamental and realized balance.

# Population stability

The relationship between community stability and the stability of the individual populations that make up the community is a key question in ecology. Importantly, ecosystem stability can result from low population stability, if populations fluctuate asynchronously, or from high population stability, if populations do not fluctuate much. Synthesis of the literature suggests diversity can have a positive or negative effect on population stability [Campbell et al 2010] (https://nsojournals.onlinelibrary.wiley.com/doi/full/10.1111/j.1600-0706.2010.18768.x) and (Xu et al 2021) [https://onlinelibrary.wiley.com/doi/full/10.1111/ele.13777].

Theoretical work has suggested that community stability is a product of two quantities: the (a) synchrony of population fluctuations, and an average species-level population stability that is weighted by relative abundance [Thibaut & Connolly 2013] (https://onlinelibrary.wiley.com/doi/full/10.1111/ele.12019).

Critically, a imbalance value close to zero can result from high response diversity, but also from high population stability (population biomass does not change largely over time).
We want to look now at whether our new metric of imbalance can capture these two stabilising mechanisms.

Thus, we calculate species-level population stability weighted by relative abundance and look at how it relates to ecosystem stability.

```{r echo=FALSE, warning=FALSE, results='hide'}

pop_aggr<-
complete_ts%>%group_by(sample_ID,predict_spec)%>%summarize(pop_CV=sd(biomass)/mean(biomass),mean_biom_contribution=mean(biom_contribution,na.rm=T))

Lehman & Tilman (2000), THIBAUT & Connolly (2013), Gross et al. (2014)
pop_aggr<-pop_aggr%>%group_by(sample_ID)%>%summarize(pop_V=sum(pop_V),
 w_pop_CV=sum(pop_CV*mean_biom_contribution, na.rm=T))

pop_aggr<-pop_aggr%>%group_by(sample_ID)%>%full_join(async_aggr,pop_aggr,by="sample_ID")

plot_pop_aggr_balance <- ggplot(data = pop_aggr, aes(y = log10(stability), x = log10(1 / w_pop_CV), color = nutrients)) +
 geom_point(size = 3.5, alpha = 0.65) +
 geom_smooth(method = "lm", linewidth = 2, show.legend = F) +
 theme_classic(base_size = 25) +
 scale_color_viridis_d(option = "plasma", begin= 0.15, end= 0.85) + # Use plasma palette
 labs(x = "log10(population stability)", y = "Temporal stability\nlog(mean/s.d.)")

plot_pop_aggr_balance

#ggsave("figures_ms/fig.pop_stab.png", plot = plot_pop_aggr_balance, width = 9, height = 6, dpi = 600)

```

```
cor.test(pop_aggrw_pop_CV,pop_aggrstability)
```

```
...
```

**\*\*Figure 13\*\*:** Relationship between log10 of population stability and log 10 of ecosystem stability.

```
```{r pop_balance, fig.align="center", fig.height=6, fig.width=12}
plot_pop_balance<-ggplot(data=pop_aggr,aes(x=log10(balance_f), y = log10(1 / w_pop_CV),color=nutrients))+
  geom_point(size = 3.5, alpha = 0.65)+
  geom_smooth(method="lm",show.legend=F, linewidth = 2)+
  theme_classic()+
  theme(axis.title.x = element_text(size = 25),
        axis.title.y = element_text(size = 25),
        axis.text = element_text(size = 20),
        strip.background = element_rect(fill = "grey80", color = NA),
        strip.text = element_text(size = 25)) +
  xlab(expression(log10(Imbalance)["fund"] )) +
  ylab("Population stability")+
  scale_color_viridis_d(option = "inferno", begin = 0.2, end = 0.8)
```

```
plot_pop_balance
```

****Figure 14**:** Relationship between fundamental imbalance and population stability divided by nutrient level.

```
# SEM
```

Finally, we use a structural equation model (SEM) to explore how stability is influenced by asynchrony, population stability, imbalance and, nutrient levels. In order to develop a hypothesis regarding the influence of stability, we have drawn on existing literature. This has enabled us to posit that stability is influenced by two key factors: asynchrony and population stability. In turn, these are influenced by balance and, in our particular case, by nutrient levels.

```
```{r}
```

```
sem_aggr3 <- pop_aggr %>%
 ungroup() %>% # Ensure there is no grouping
 mutate(
 log_balance_f = log10(balance_f),
 stability = log10(1 / CV),
 richness = as.numeric(richness),
 temperature=temperature,
 asynchrony_Gross= (-synchrony_Gross),
 pop_stability= log10(1/w_pop_CV),
 pop_var = log10(pop_V)
 #Keep it as an ordered factor
)
```

```
model1C <- '
 stability ~ asynchrony_Gross
 +pop_stability

 asynchrony_Gross ~ log_balance_f + nutrients
 pop_stability~log_balance_f + nutrients
```

```
,
```

```
Fit the model
```

```
fit1C <- sem(model1C,estimator="MLM",meanstructure = TRUE,data = sem_aggr3)
#modificationindices(fit1C)
```

```
Summarize the results
summary(fit1C, standardized = TRUE,rsquare=T, fit.measures = TRUE)
```

```
...
```

```
```{r SEM, fig.cap = 'SEM.', fig.align="center", fig.height=6, fig.width=8}
knitr::include_graphics(("figures_ms/SEM_3.png"))
```

```
...
```

****Figure 15**:** Structural equation model (SEM) showing the relationships between stability, asynchrony, population stability, imbalance, and nutrient levels. The model includes standardized regression coefficients and fit indices.

****Model Fit Indices****

The model fit indices suggest that the model fits the data well.

***Chi-Square Test (User Model)*:** The chi-square test statistic for the user model is $\chi^2 = 1.626$ (scaled = 1.465) with 3 degrees of freedom and a p-value of 0.475 (scaled = 0.501). This indicates a good fit, as the test is non-significant, suggesting no significant difference between the observed and model-implied covariance matrices.

***Comparative Fit Index (CFI)* and *Tucker-Lewis Index (TLI)*:** Both CFI and TLI values are 1.000, indicating an excellent model fit. Values close to or above 0.95 are generally considered good.

***Root Mean Square Error of Approximation (RMSEA)*:** The RMSEA is 0.000, with a 90% confidence interval ranging from 0 to 0.090 (scaled = 0.080). This indicates a very good fit, as RMSEA values below 0.05 are ideal, and values below 0.08 are acceptable. The p-values for the RMSEA hypothesis tests suggest strong support for a close fit (RMSEA <= 0.05) and little evidence for a poor fit (RMSEA >= 0.08).

Standardized Root Mean Square Residual (SRMR): The SRMR value is 0.017, which is also within the acceptable range (values below 0.08 are generally considered good).
Overall, the fit indices suggest that the model is an excellent fit for the data.

Regression Paths and Interpretation

Stability Regressions

Stability ~ Asynchrony_Gross (asynchrony_Grass): The standardized estimate for the effect of asynchrony on stability is 0.340 ($p < 0.001$), indicating a significant positive association. Higher asynchrony in species dynamics is associated with increased community stability.

Stability ~ Population Stability (pop_stability): The standardized estimate is 0.977 ($p < 0.001$), showing a strong positive relationship. This suggests that community stability is highly dependent on the stability of individual populations within the community.

Asynchrony_Gross Regressions

Asynchrony_Gross ~ Log10(Balance): The standardized estimate is -0.176 ($p = 0.013$), indicating a significant negative effect. Higher imbalance leads to lower asynchrony, suggesting that as imbalance increases, species within the community fluctuate more synchronously.

Asynchrony_Gross ~ Nutrients: The standardized estimate is -0.469 ($p < 0.001$), showing a strong negative relationship. Higher nutrient levels appear to reduce asynchrony, possibly by causing similar responses across species.

Population Stability Regressions

Population Stability ~ Log10(Balance): The standardized estimate is -0.296 ($p < 0.001$), indicating that higher imbalance is associated with lower population stability.

Population Stability ~ Nutrients: The standardized estimate is 0.635 ($p < 0.001$), showing that higher nutrient levels are associated with increased population stability, likely because nutrients enhance conditions that support stable population dynamics.

Variances and R-Squared Values

R-Squared for Stability: The model explains 90.4% of the variance in community stability, indicating strong predictive power.

R-Squared for Asynchrony_Gross: The model explains 21.9% of the variance in asynchrony, which is moderate.

R-Squared for Population Stability: The model explains 56.2% of the variance in population stability, showing that nutrients and balance are important but not the only factors influencing it.

Summary Interpretation

Model Fit: The model has an excellent fit, as indicated by the fit indices.

Stability: Community stability is strongly influenced by both population stability and asynchrony among species, with population stability being the stronger predictor.

Asynchrony and Imbalance: Asynchrony decreases with increasing imbalance and nutrients, suggesting that these factors promote more synchronized fluctuations among species.

Population Stability and Nutrients: Higher nutrient levels are associated with increased population stability, suggesting that nutrient availability supports stable population dynamics. Conversely, higher imbalance is associated with decreased population stability.