

# What is a successful reintroduction?

2023-12-01

Workflow to characterise the factors driving reintroduction success

```
library(dplyr)
library(plyr)
library(tidyverse)
library(magrittr)
library(here)
library(arrow)
library(brms)
library(ggeffects)
library(ggpubr)
library(brms)
library(scales)
library(mgcv)

palette_3 = c("#604087", "#4f67a1", "#659c75")
palette_7 = c("#2b4570", "#cfc0ba", "#8e2e29", "#e8871e", "#8ba29c", "#a8d0db", "#a37a74")

load("sim.Rdata")
sims = sim_tom_ok
sims$z_1 = sims$Z
sims$delta_sp_32 = as.numeric(sims$delta_sp_32)
sims$delta_stabp_31 = sims$delta_stabp_31
sims$delta_stabc_21 = sims$delta_stabc_21
sims$delta_stabc_32 = sims$delta_stabc_32
sims$stab_com_1 = log(sims$stab_com_1)
sims$stab_com_2 = log(sims$stab_com_2)
summary(sims) #In analyses, need to log all deltas that arent comparing to a maximum.
```

```
##      sim_id      Z      nb_link      rep
## Min.    : 1  Min.    : 10.0  Min.    :1  Min.    : 1.0
```

## 1st Qu.:17474	1st Qu.: 20.0	1st Qu.:3	1st Qu.: 26.0
## Median :35065	Median : 100.0	Median :5	Median : 51.0
## Mean :35564	Mean : 268.5	Mean :5	Mean : 50.5
## 3rd Qu.:53914	3rd Qu.: 500.0	3rd Qu.:7	3rd Qu.: 75.0
## Max. :71400	Max. :1000.0	Max. :9	Max. :100.0
##		NA's :11614	
## realised_C_1	realised_C_2	realised_C_3	realised_S_1
## Min. :0.0274	Min. :0.0123	Min. :0.0254	Min. : 2.00
## 1st Qu.:0.0878	1st Qu.:0.0823	1st Qu.:0.0861	1st Qu.: 9.00
## Median :0.1276	Median :0.1228	Median :0.1280	Median :19.00
## Mean :0.1399	Mean :0.1350	Mean :0.1399	Mean :22.33
## 3rd Qu.:0.1800	3rd Qu.:0.1736	3rd Qu.:0.1821	3rd Qu.:30.00
## Max. :0.4167	Max. :0.4167	Max. :0.4800	Max. :59.00
## NA's :2837	NA's :2837	NA's :2837	NA's :2837
## realised_S_2	realised_S_3	g_1	g_3
## Min. : 2.0	Min. : 2.00	Min. : 1.0	Min. : 1.000
## 1st Qu.: 9.0	1st Qu.: 9.00	1st Qu.: 2.0	1st Qu.: 2.000
## Median :18.0	Median :18.00	Median : 4.0	Median : 4.000
## Mean :21.1	Mean :20.67	Mean : 5.7	Mean : 4.254
## 3rd Qu.:29.0	3rd Qu.:28.00	3rd Qu.: 8.0	3rd Qu.: 6.000
## Max. :59.0	Max. :59.00	Max. :48.0	Max. :29.000
## NA's :2837	NA's :2837	NA's :2837	NA's :2837
## stab_com_1	stab_com_2	tlvl_mean_1	tlvl_mean_2
## Min. : 1.694	Min. : 1.593	Min. :1.414	Min. :1.052
## 1st Qu.: 9.067	1st Qu.:10.243	1st Qu.:2.273	1st Qu.:2.085
## Median :12.315	Median :14.534	Median :2.580	Median :2.400
## Mean :13.002	Mean :14.670	Mean :2.679	Mean :2.478
## 3rd Qu.:16.558	3rd Qu.:18.387	3rd Qu.:2.981	3rd Qu.:2.791
## Max. :37.880	Max. :36.509	Max. :6.338	Max. :6.506
## NA's :2837	NA's :2837	NA's :2837	NA's :2837
## evenness_1	evenness_2	evenness_3	delta_ev_21
## Min. :0.0018	Min. :0.0016	Min. :0.0018	Min. : 0.110
## 1st Qu.:0.0444	1st Qu.:0.0528	1st Qu.:0.0540	1st Qu.: 1.037
## Median :0.0645	Median :0.0793	Median :0.0804	Median : 1.159
## Mean :0.0761	Mean :0.0955	Mean :0.0944	Mean : 1.296
## 3rd Qu.:0.0967	3rd Qu.:0.1249	3rd Qu.:0.1228	3rd Qu.: 1.372
## Max. :0.2282	Max. :0.3662	Max. :0.3661	Max. :16.462
## NA's :2837	NA's :2837	NA's :2837	NA's :2837
## delta_ev_31	delta_ev_32	delta_s_21	delta_s_32
## Min. : 0.0475	Min. : 0.0420	Min. :0.3333	Min. :0.4286
## 1st Qu.: 1.0358	1st Qu.: 0.9267	1st Qu.:0.9167	1st Qu.:1.0000

```
## Median : 1.1722 Median : 0.9957 Median :1.0000 Median :1.0000
## Mean : 1.2916 Mean : 1.0465 Mean :0.9470 Mean :0.9823
## 3rd Qu.: 1.3903 3rd Qu.: 1.0957 3rd Qu.:1.0000 3rd Qu.:1.0000
## Max. :17.5463 Max. :11.7232 Max. :1.0000 Max. :1.0000
## NA's :2837 NA's :2837 NA's :2837 NA's :2837
## delta_bmc_21 delta_bmc_32 delta_stabc_21 delta_stabc_32
## Min. :0.0672 Min. : 0.1895 Min. :0.000e+00 Min. : 0
## 1st Qu.:0.8128 1st Qu.: 1.0000 1st Qu.:1.000e+00 1st Qu.: 0
## Median :0.9487 Median : 1.0188 Median :6.000e+00 Median : 1
## Mean :0.8934 Mean : 1.1660 Mean :2.982e+06 Mean : 61514
## 3rd Qu.:0.9955 3rd Qu.: 1.1658 3rd Qu.:6.600e+01 3rd Qu.: 10
## Max. :5.8560 Max. :21.5461 Max. :7.675e+10 Max. :593909137
## NA's :2837 NA's :2837 NA's :2837 NA's :2837
## delta_g_31 delta_sp_32 delta_bmp_31 delta_stabp_31
## Min. :0.0270 Min. :0.0000 Min. : -Inf Min. :0.000e+00
## 1st Qu.:0.5000 1st Qu.:1.0000 1st Qu.:0.400 1st Qu.:1.000e+00
## Median :0.8333 Median :1.0000 Median :0.968 Median :3.000e+00
## Mean :1.3430 Mean :0.9092 Mean : NaN Mean :1.044e+08
## 3rd Qu.:1.5000 3rd Qu.:1.0000 3rd Qu.:2.510 3rd Qu.:6.000e+01
## Max. :9.0000 Max. :1.0000 Max. : Inf Max. :2.912e+12
## NA's :2837 NA's :2837 NA's :2878 NA's :8947
## z_1
## Min. : 10.0
## 1st Qu.: 20.0
## Median :100.0
## Mean : 268.5
## 3rd Qu.: 500.0
## Max. :1000.0
##
```

```
sims = subset(sims, is.finite(sims$delta_bmp_31))
sims = subset(sims, is.na(sims$nb_link))
```

Model 1 - Probability the predator can establish (survive) reintroduction

```
m1 = glm(delta_sp_32 ~ realised_C_1*realised_S_1 + g_1*delta_g_31 + delta_s_21 + z_1, data = sims, family=binomial(link='logit'))
summary(m1)
```

```
##
```

```
## Call:
## glm(formula = delta_sp_32 ~ realised_C_1 * realised_S_1 + g_1 *
##      delta_g_31 + delta_s_21 + z_1, family = binomial(link = "logit"),
##      data = sims)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -3.3947    0.1720    0.2369    0.2999    0.6097
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      1.3263038   0.7592107   1.747 0.080646 .
## realised_C_1     -0.1477199   1.5178120  -0.097 0.922469
## realised_S_1      0.0137905   0.0094220   1.464 0.143289
## g_1              0.0516643   0.0485779   1.064 0.287539
## delta_g_31       0.2239613   0.3624661   0.618 0.536653
## delta_s_21       1.2951460   0.6750597   1.919 0.055039 .
## z_1              0.0014682   0.0002306   6.366 1.94e-10 ***
## realised_C_1:realised_S_1 -0.2923196   0.0786419  -3.717 0.000202 ***
## g_1:delta_g_31     0.1588710   0.0680804   2.334 0.019618 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 3174.9  on 10968  degrees of freedom
## Residual deviance: 3019.2  on 10960  degrees of freedom
## AIC: 3037.2
##
## Number of Fisher Scoring iterations: 7

m1_z = ggpredict(m1, "z_1 [1:1000, by = 5]")
plt_m1_z = ggplot(m1_z) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Mass ratio (Predator:Prey)", y = "Probability population reestablishes") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic()

m1_c_s = ggpredict(m1, terms = c("realised_C_1 [0:0.3, by = 0.01]", "realised_S_1 [5, 15, 50]"))
```

```

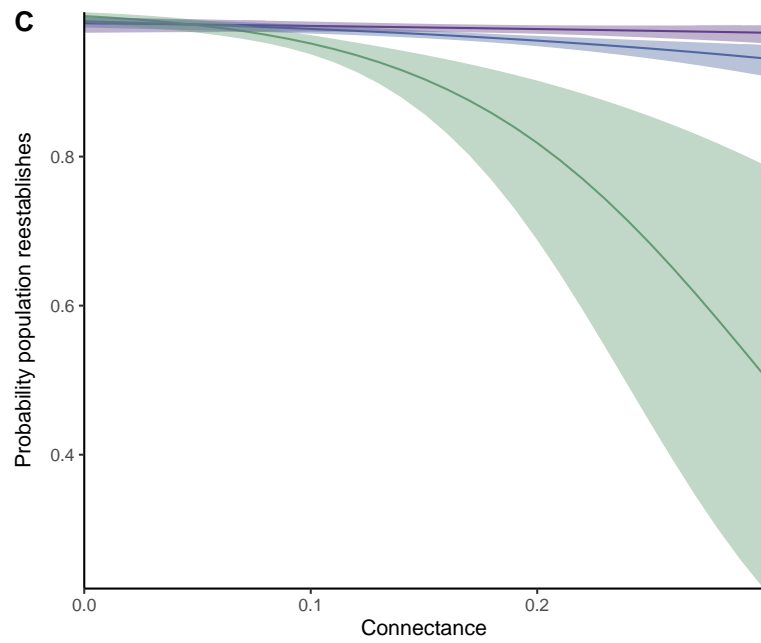
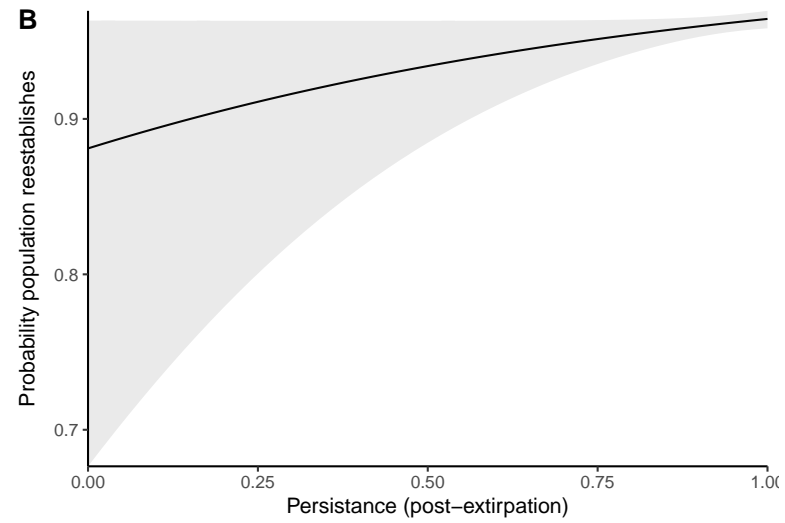
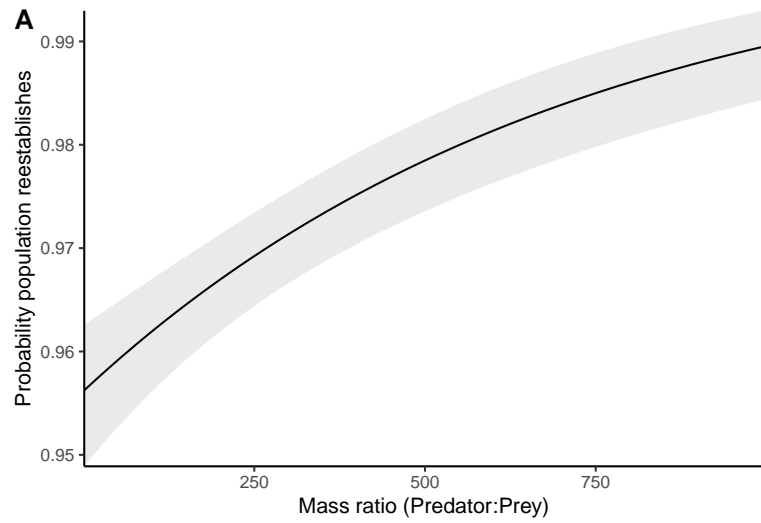
plt_m1_c_s = ggplot(m1_c_s) +
  geom_line(aes(x = x, y = predicted, colour = group)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.4) +
  labs(x = "Connectance", y = "Probability population reestablishes") +
  scale_fill_manual(values = palette_3, name = "Richness") +
  scale_colour_manual(values = palette_3, name = "Richness") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(legend.position = "bottom")

m1_g_deltag = ggpredict(m1, terms = c("delta_g_31 [0.5:1, by = 0.02]", "g_1 [3, 7, 10]"))
plt_m1_g_deltag = ggplot(m1_g_deltag) +
  geom_line(aes(x = x, y = predicted, colour = group)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.4) +
  labs(x = "Predator interaction intactness", y = "Probability population reestablishes") +
  scale_fill_manual(values = palette_3, name = "Interactions") +
  scale_colour_manual(values = palette_3, name = "Interactions") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(legend.position = "bottom")

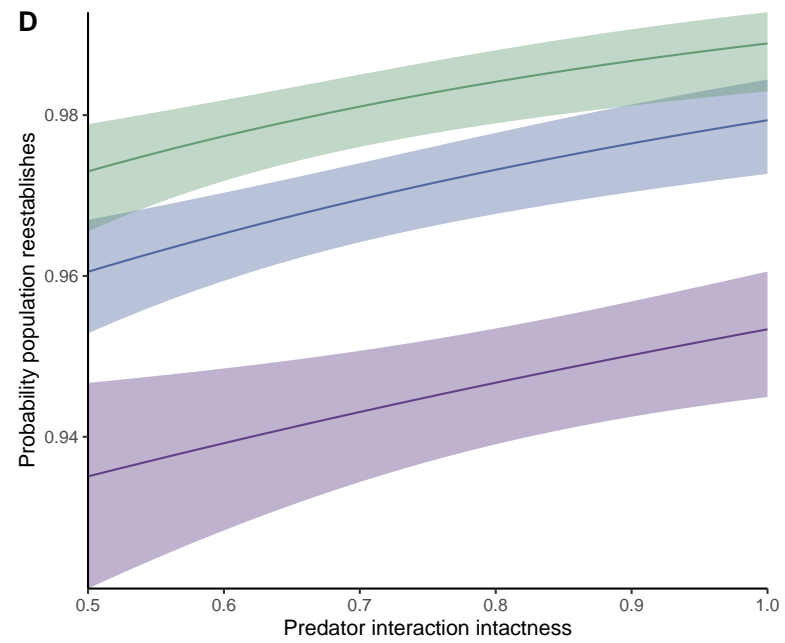
m1_delta_s = ggpredict(m1, "delta_s_21 [0:1, by = 0.02]")
plt_m1_delta_s = ggplot(m1_delta_s) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Persistence (post-extirpation)", y = "Probability population reestablishes") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic()

ggarrange(plt_m1_z, plt_m1_delta_s, plt_m1_c_s, plt_m1_g_deltag, ncol = 2, nrow = 2, align = "hv", labels = c("A", "B", "C", "D"), heights = c(1,

```



Richness 5 15 50



Interactions 3 7 10

## Model 2 - Change in predator population stability post-reintroduction

```
#Population re-establish stability?
m2 = lm(log(delta_stabp_31) ~ realised_C_1*realised_S_1 + g_1*delta_g_31 + delta_s_21 + z_1, data = sims)
summary(m2)

##
## Call:
## lm(formula = log(delta_stabp_31) ~ realised_C_1 * realised_S_1 +
##     g_1 * delta_g_31 + delta_s_21 + z_1, data = sims)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.5156  -1.4809  -0.3306   1.4118  23.3318
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      6.2239239   0.5381725   11.565 < 2e-16 ***
## realised_C_1      1.6978738   0.9150960    1.855 0.063566 .
## realised_S_1      0.0219007   0.0056689    3.863 0.000113 ***
## g_1              0.1200170   0.0266358    4.506 6.68e-06 ***
## delta_g_31      -0.8878407   0.2498271   -3.554 0.000381 ***
## delta_s_21      -4.5417159   0.4968707   -9.141 < 2e-16 ***
## z_1             -0.0009494   0.0001077   -8.815 < 2e-16 ***
## realised_C_1:realised_S_1 -0.2449197   0.0511673   -4.787 1.72e-06 ***
## g_1:delta_g_31    -0.1183862   0.0353658   -3.347 0.000818 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.543 on 10598 degrees of freedom
## (362 observations deleted due to missingness)
## Multiple R-squared:  0.04841, Adjusted R-squared:  0.04769
## F-statistic: 67.39 on 8 and 10598 DF, p-value: < 2.2e-16

m2_g = ggpredict(m2, "g_1 [0:40, by = 0.1]", back_transform = F)
plt_m2_g = ggplot(m2_g) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Interactions", y = "\u0394 Population stability") +
```

```

#scale_y_log10(expand = c(0,0),
#               breaks = c(0.1,1,10),
#               labels = c("1/10", "1/1", "10/1"),
#               limits =c(0.05, 50)) +
scale_x_continuous(expand = c(0,0),
                   limits = c(0, 20)) +
theme_classic()

m2_deltag = ggpredict(m2, terms = c("delta_g_31 [0.5:1, by = 0.02]"), back_transform = F)
plt_m2_deltag = ggplot(m2_deltag) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Predator interaction intactness", y = "\u0394 Population stability") +
  scale_x_continuous(expand = c(0,0),
                    limits = c(0.5, 1)) +
  theme_classic()

m2_delta_s = ggpredict(m2, "delta_s_21 [0:1, by = 0.02]", back_transform = F)
plt_m2_delta_s = ggplot(m2_delta_s) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Persistence (post-extirpation)", y = "\u0394 Population stability") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

m2_c = ggpredict(m2, "realised_C_1 [0:0.4, by = 0.01]", back_transform = F)
plt_m2_c = ggplot(m2_c) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Connectance", y = "\u0394 Population stability") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

m2_z = ggpredict(m2, "z_1 [0:1000, by = 10]", back_transform = F)
plt_m2_z = ggplot(m2_z) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +

```

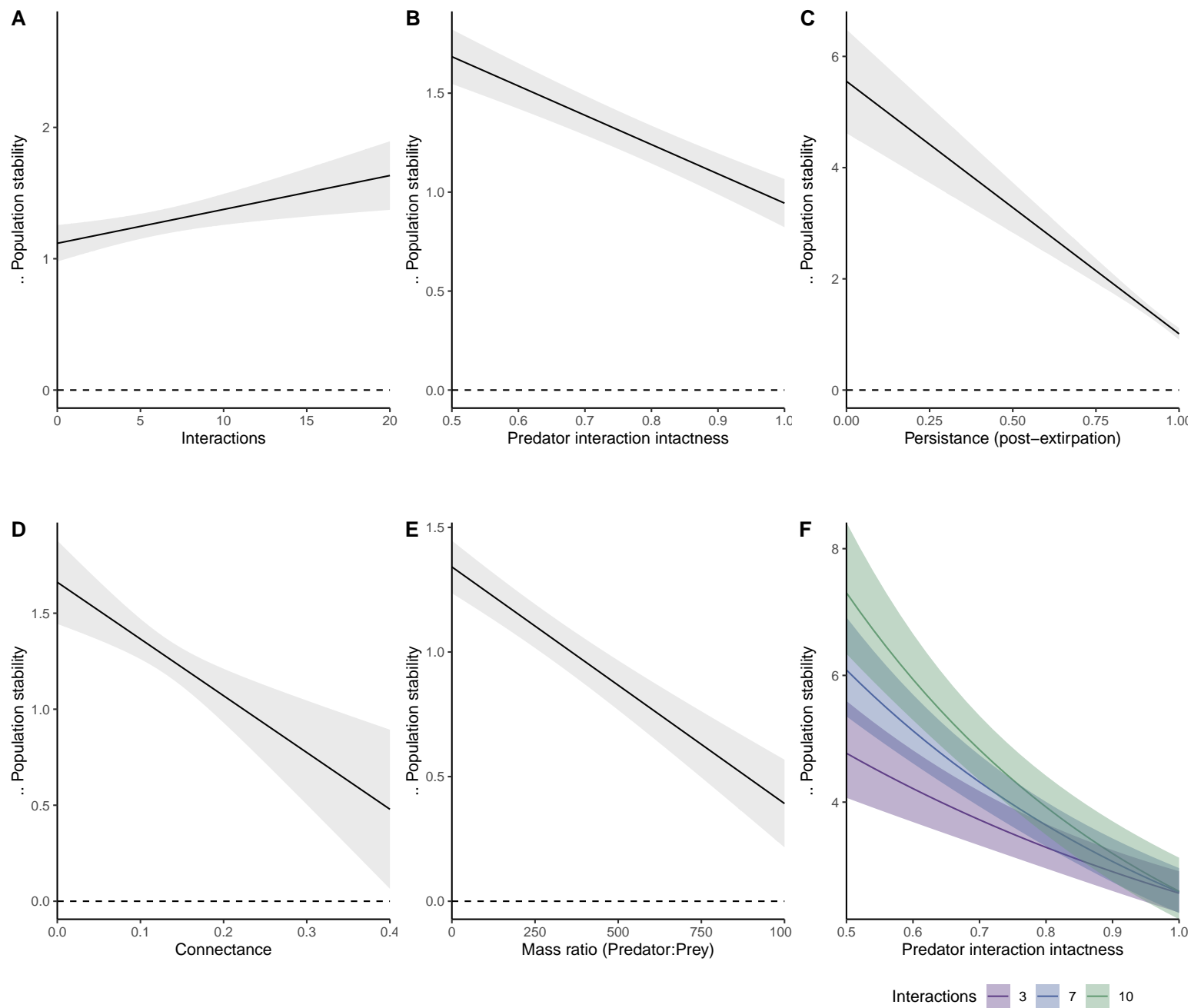
```

  labs(x = "Mass ratio (Predator:Prey)", y = "\u0394 Population stability") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

m2_g_deltag = ggpredict(m2, terms = c("delta_g_31 [0.5:1, by = 0.02]", "g_1 [3, 7, 10]"))
plt_m2_g_deltag = ggplot(m2_g_deltag) +
  geom_line(aes(x = x, y = predicted, colour = group)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.4) +
  labs(x = "Predator interaction intactness", y = "\u0394 Population stability") +
  scale_fill_manual(values = palette_3, name = "Interactions") +
  scale_colour_manual(values = palette_3, name = "Interactions") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(legend.position = "bottom")

ggarrange(plt_m2_g, plt_m2_deltag, plt_m2_delta_s, plt_m2_c, plt_m2_z, plt_m2_g_deltag, ncol = 3, nrow = 2, align = "hv", labels = c("A", "B", "C"))

```



### Model 3 - Change in predator biomass post-reintroduction

```
sims$delta_bmp_31 = ifelse(sims$delta_bmp_31 == 0, NA, sims$delta_bmp_31) #Remove cases where predator went extinct in 3

m3 = lm(log(delta_bmp_31) ~ realised_C_1*realised_S_1 + g_1*delta_g_31 + delta_s_21 + z_1, data = sims)
summary(m3)
```

```
##
## Call:
## lm(formula = log(delta_bmp_31) ~ realised_C_1 * realised_S_1 +
##      g_1 * delta_g_31 + delta_s_21 + z_1, data = sims)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.5618  -0.3794  -0.0700   0.1676  11.1245
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.924e-01  2.040e-01   0.943  0.345707
## realised_C_1      1.136e+00  3.463e-01   3.279  0.001045 **
## realised_S_1      9.452e-04  2.147e-03   0.440  0.659753
## g_1               2.771e-02  1.010e-02   2.743  0.006090 **
## delta_g_31       -3.628e-01  9.442e-02  -3.842  0.000123 ***
## delta_s_21        2.670e-01  1.882e-01   1.419  0.156006
## z_1              -9.172e-05  4.066e-05  -2.256  0.024099 *
## realised_C_1:realised_S_1 -6.726e-02  1.932e-02  -3.481  0.000501 ***
## g_1:delta_g_31     -5.597e-02  1.340e-02  -4.176  2.99e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.346 on 10710 degrees of freedom
## (250 observations deleted due to missingness)
## Multiple R-squared:  0.01667,    Adjusted R-squared:  0.01593
## F-statistic: 22.69 on 8 and 10710 DF,  p-value: < 2.2e-16
```

```
m3_z = ggpredict(m3, "z_1 [1:1000, by = 5]", back_transform = F)
plt_m3_z = ggplot(m3_z) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 1), linetype = "dashed") +
```

```

labs(x = "Mass ratio (Predator:Prey)", y = "\u0394 Predator biomass") +
scale_x_continuous(expand = c(0,0)) +
theme_classic()

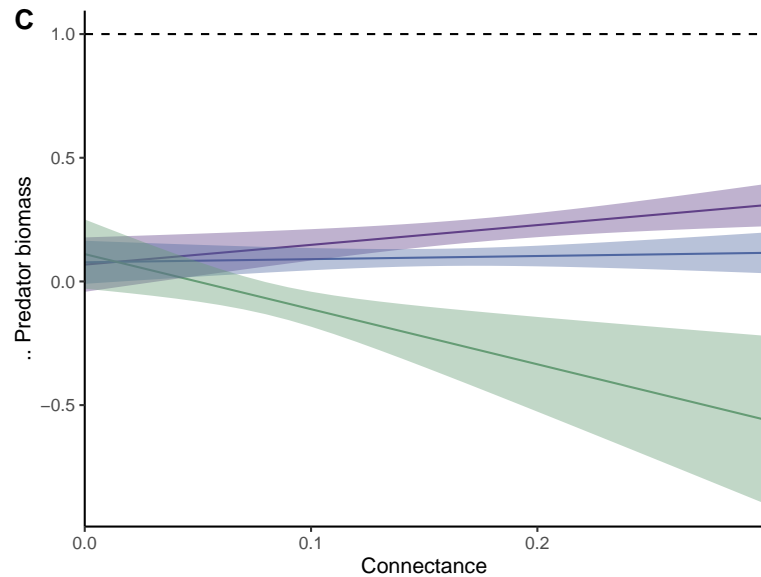
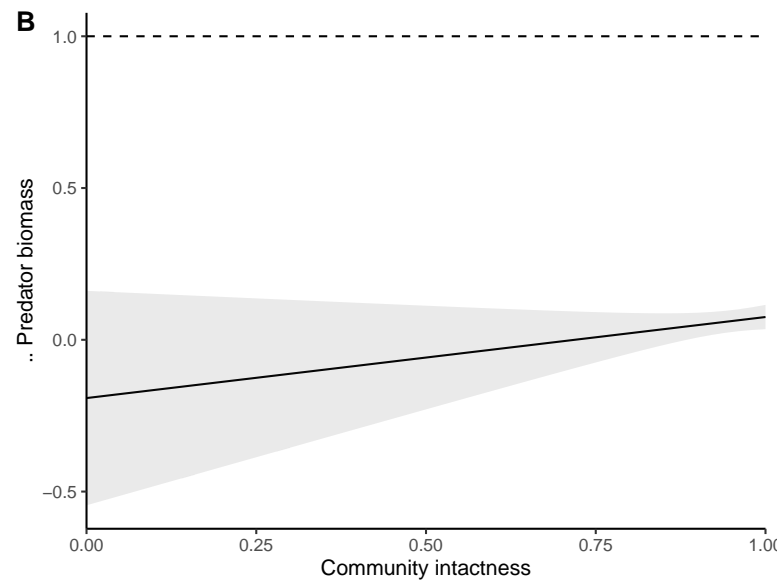
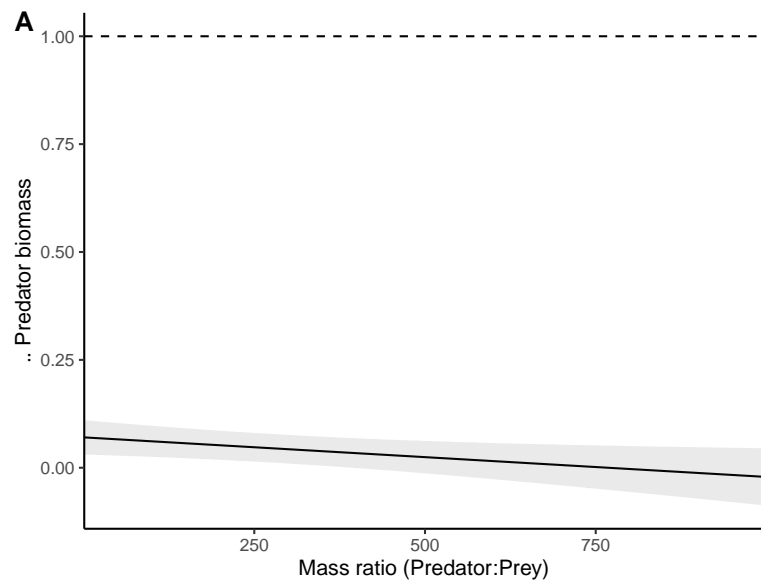
m3_c_s = ggpredict(m3, terms = c("realised_C_1 [0:0.3, by = 0.01]", "realised_S_1 [5, 15, 50]"), back_transform = F)
plt_m3_c_s = ggplot(m3_c_s) +
  geom_line(aes(x = x, y = predicted, colour = group)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.4) +
  geom_hline(aes(yintercept = 1), linetype = "dashed") +
  labs(x = "Connectance", y = "\u0394 Predator biomass") +
  scale_fill_manual(values = palette_3, name = "Richness") +
  scale_colour_manual(values = palette_3, name = "Richness") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(legend.position = "bottom")

m3_g_deltag = ggpredict(m3, terms = c("delta_g_31 [0.5:1, by = 0.02]", "g_1 [3, 7, 10]"), back_transform = F)
plt_m3_g_deltag = ggplot(m3_g_deltag) +
  geom_line(aes(x = x, y = predicted, colour = group)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.4) +
  labs(x = "Predator interaction intactness", y = "\u0394 Predator biomass") +
  scale_fill_manual(values = palette_3, name = "Interactions") +
  scale_colour_manual(values = palette_3, name = "Interactions") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(legend.position = "bottom")

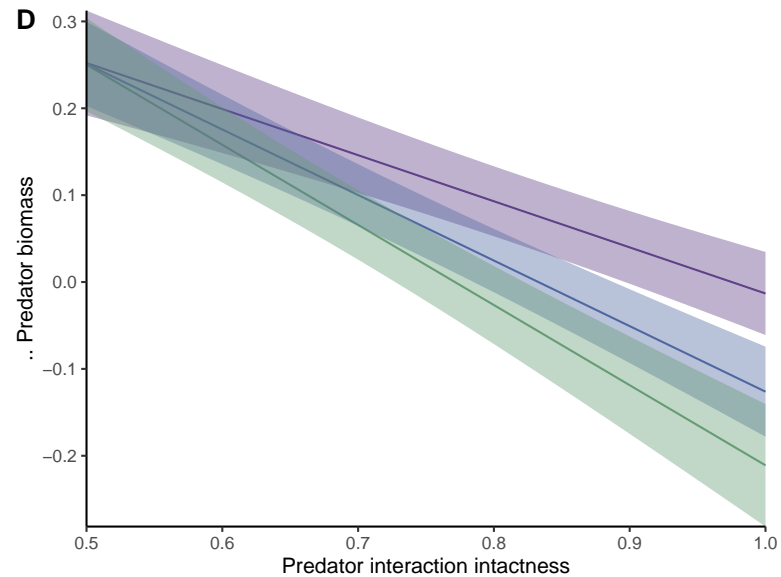
m3_delta_s = ggpredict(m3, "delta_s_21 [0:1, by = 0.02]", back_transform = F)
plt_m3_delta_s = ggplot(m3_delta_s) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 1), linetype = "dashed") +
  labs(x = "Community intactness", y = "\u0394 Predator biomass") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

ggarrange(plt_m3_z, plt_m3_delta_s, plt_m3_c_s, plt_m3_g_deltag, ncol = 2, nrow = 2, align = "h", labels = c("A", "B", "C", "D"), heights = c(1, 1

```



Richness 5 15 50



Interactions 3 7 10

#### Model 4 - Post-extirpation community persistence

```
sims$delta_s_21 = sims$delta_s_21 - 0.01

m4 = gam(delta_s_21 ~ realised_C_1*realised_S_1 + g_1 + z_1 + stab_com_1 + tlvl_mean_1, data = sims, family=betar(link="logit"))
summary(m4)

##
## Family: Beta regression(12.699)
## Link function: logit
##
## Formula:
## delta_s_21 ~ realised_C_1 * realised_S_1 + g_1 + z_1 + stab_com_1 +
##      tlvl_mean_1
##
## Parametric coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      3.046e+00  7.180e-02  42.430 < 2e-16 ***
## realised_C_1      9.888e-01  2.208e-01   4.477 7.55e-06 ***
## realised_S_1      4.084e-03  1.371e-03   2.980 0.002887 **
## g_1              -3.018e-02  1.817e-03 -16.610 < 2e-16 ***
## z_1               1.451e-04  2.912e-05   4.983 6.27e-07 ***
## stab_com_1        7.357e-03  2.092e-03   3.516 0.000438 ***
## tlvl_mean_1      -8.011e-02  2.040e-02  -3.927 8.61e-05 ***
## realised_C_1:realised_S_1 -1.202e-01  1.308e-02  -9.194 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## R-sq.(adj) =  0.0673   Deviance explained = 8.77%
## -REML = -19979   Scale est. = 1           n = 10969

m4_z = ggpredict(m4, "z_1 [1:1000, by = 5]")
plt_m4_z = ggplot(m4_z) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Mass ratio (Predator:Prey)", y = "Persistence (post-extirpation)") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic()
```

```

m4_tlvl = ggpredict(m4, "tlvl_mean_1 [1:6, by = 0.1]")
plt_m4_tlvl = ggplot(m4_tlvl) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Trophic level (mean)", y = "Persistence (post-extirpation)") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic()

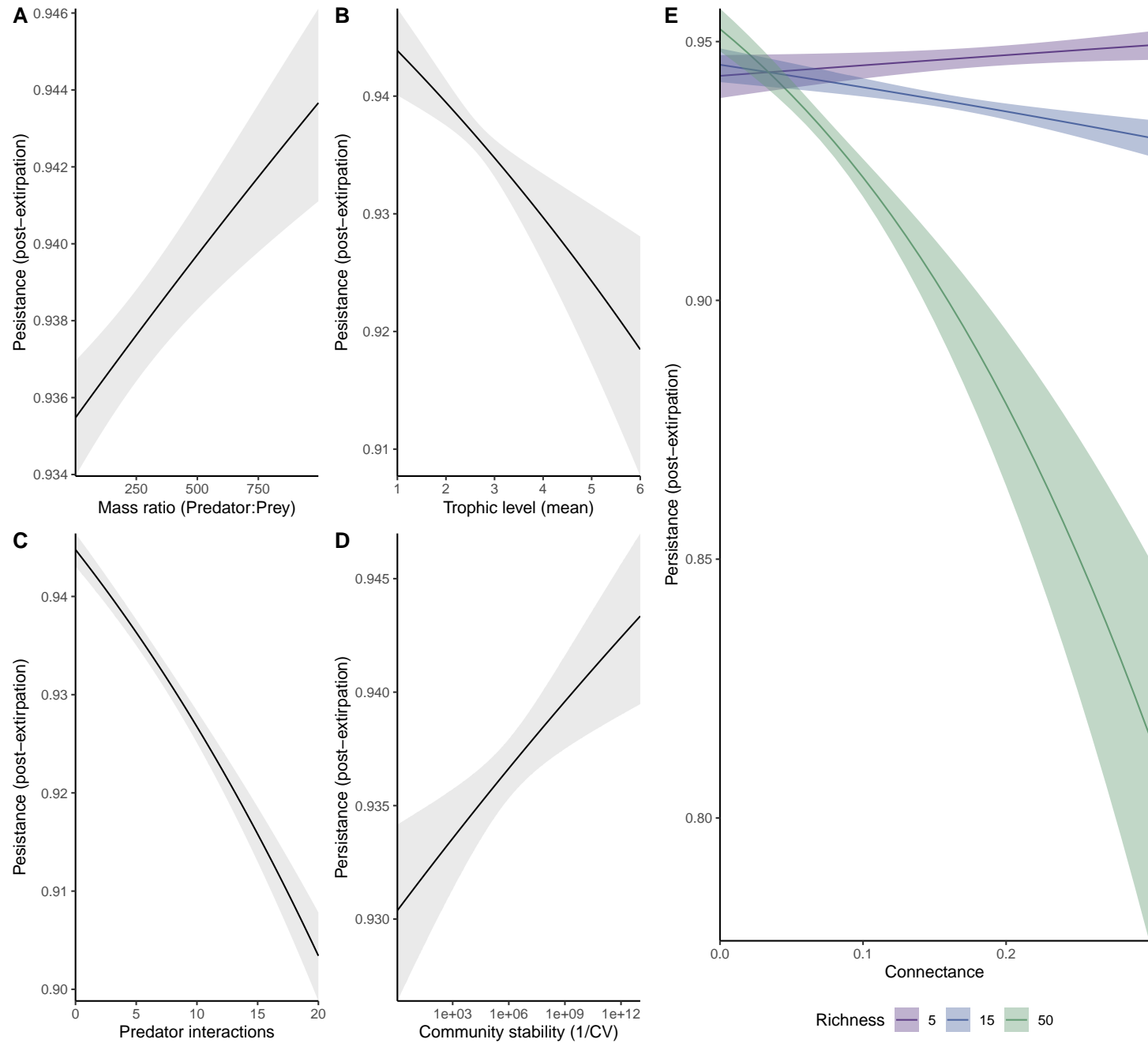
# Needs transformation
m4_cv_com = ggpredict(m4, "stab_com_1 [0.1:30, by = 0.1]")
plt_m4_cv_com = ggplot(m4_cv_com) +
  geom_line(aes(x = exp(x), y = predicted)) +
  geom_ribbon(aes(x = exp(x), ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Community stability (1/CV)", y = "Persistence (post-extirpation)") +
  scale_x_log10(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic()

m4_g = ggpredict(m4, "g_1 [0:20, by = 0.1]")
plt_m4_g = ggplot(m4_g) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Predator interactions", y = "Persistence (post-extirpation)") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic()

m4_c_s = ggpredict(m4, terms = c("realised_C_1 [0:0.3, by = 0.01]", "realised_S_1 [5, 15, 50]"))
plt_m4_c_s = ggplot(m4_c_s) +
  geom_line(aes(x = x, y = predicted, colour = group)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.4) +
  labs(x = "Connectance", y = "Persistence (post-extirpation)") +
  scale_fill_manual(values = palette_3, name = "Richness") +
  scale_colour_manual(values = palette_3, name = "Richness") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(legend.position = "bottom")

```

```
ggarrange(  
  ggarrange(plt_m4_z, plt_m4_tlvl, plt_m4_g, plt_m4_cv_com, ncol = 2, nrow = 2, align = "hv", labels = c("A", "B", "C", "D"), heights = c(1)),  
  plt_m4_c_s,  
  ncol = 2, nrow = 1, widths = c(1,0.7), labels = c("", "E"), align = "v")
```



## Model 5 - Post-reintroduction community persistence

```
sims$delta_s_32 = sims$delta_s_32 - 0.01

m5 = gam(delta_s_32 ~ realised_C_1*realised_S_1 + delta_s_21 + g_1 + z_1 + tlvl_mean_1 + delta_sp_32 + stab_com_2, data = sims, family=betar(link=summary(m5)

##
## Family: Beta regression(57.874)
## Link function: logit
##
## Formula:
## delta_s_32 ~ realised_C_1 * realised_S_1 + delta_s_21 + g_1 +
##      z_1 + tlvl_mean_1 + delta_sp_32 + stab_com_2
##
## Parametric coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      3.506e+00  1.120e-01  31.322 < 2e-16 ***
## realised_C_1      2.689e-01  1.884e-01   1.427 0.153541
## realised_S_1     -4.688e-03  1.127e-03  -4.158 3.20e-05 ***
## delta_s_21       3.406e-01  9.111e-02   3.738 0.000185 ***
## g_1              5.214e-03  1.637e-03   3.185 0.001448 **
## z_1             -1.220e-04  2.333e-05  -5.229 1.70e-07 ***
## tlvl_mean_1     -1.810e-02  1.724e-02  -1.050 0.293852
## delta_sp_32      3.526e-02  3.902e-02   0.904 0.366100
## stab_com_2       1.489e-02  1.685e-03   8.841 < 2e-16 ***
## realised_C_1:realised_S_1 -7.215e-02  1.089e-02  -6.628 3.39e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## R-sq.(adj) =  0.0611   Deviance explained = 8.21%
## -REML = -31063   Scale est. = 1           n = 10969

m5_z = ggpredict(m5, "z_1 [1:1000, by = 5]")
plt_m5_z = ggplot(m5_z) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Mass ratio (Predator:Prey)", y = "Persistence (post-reintroduction)") +
  scale_x_continuous(expand = c(0,0)) +
```

```

scale_y_continuous(expand = c(0,0)) +
theme_classic()

m5_tlvl1 = ggpredict(m5, "tlvl_mean_1 [1:6, by = 0.1]")
plt_m5_tlvl1 = ggplot(m5_tlvl1) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Trophic level (mean)", y = "Persistence (post-reintroduction)") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic()

# Needs transformation
m5_cv_com = ggpredict(m5, "stab_com_2 [0.1:30, by = 0.1]")
plt_m5_cv_com = ggplot(m5_cv_com) +
  geom_line(aes(x = exp(x), y = predicted)) +
  geom_ribbon(aes(x = exp(x), ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Community stability (1/CV)", y = "Persistence (post-reintroduction)") +
  scale_x_log10(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic()

m5_deltas_21 = ggpredict(m5, "delta_s_21 [0.1:1, by = 0.01]")
plt_m5_deltas_21 = ggplot(m5_deltas_21) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Persistence (post-extirpation)", y = "Persistence (post-reintroduction)") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic()

m5_deltasp_32 = ggpredict(m5, "delta_sp_32 [0.1:1, by = 0.01]")
plt_m5_deltasp_32 = ggplot(m5_deltasp_32) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Probability population reestablishes", y = "Persistence (post-reintroduction)") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic()

m5_g = ggpredict(m5, "g_1 [0:20, by = 0.1]")

```

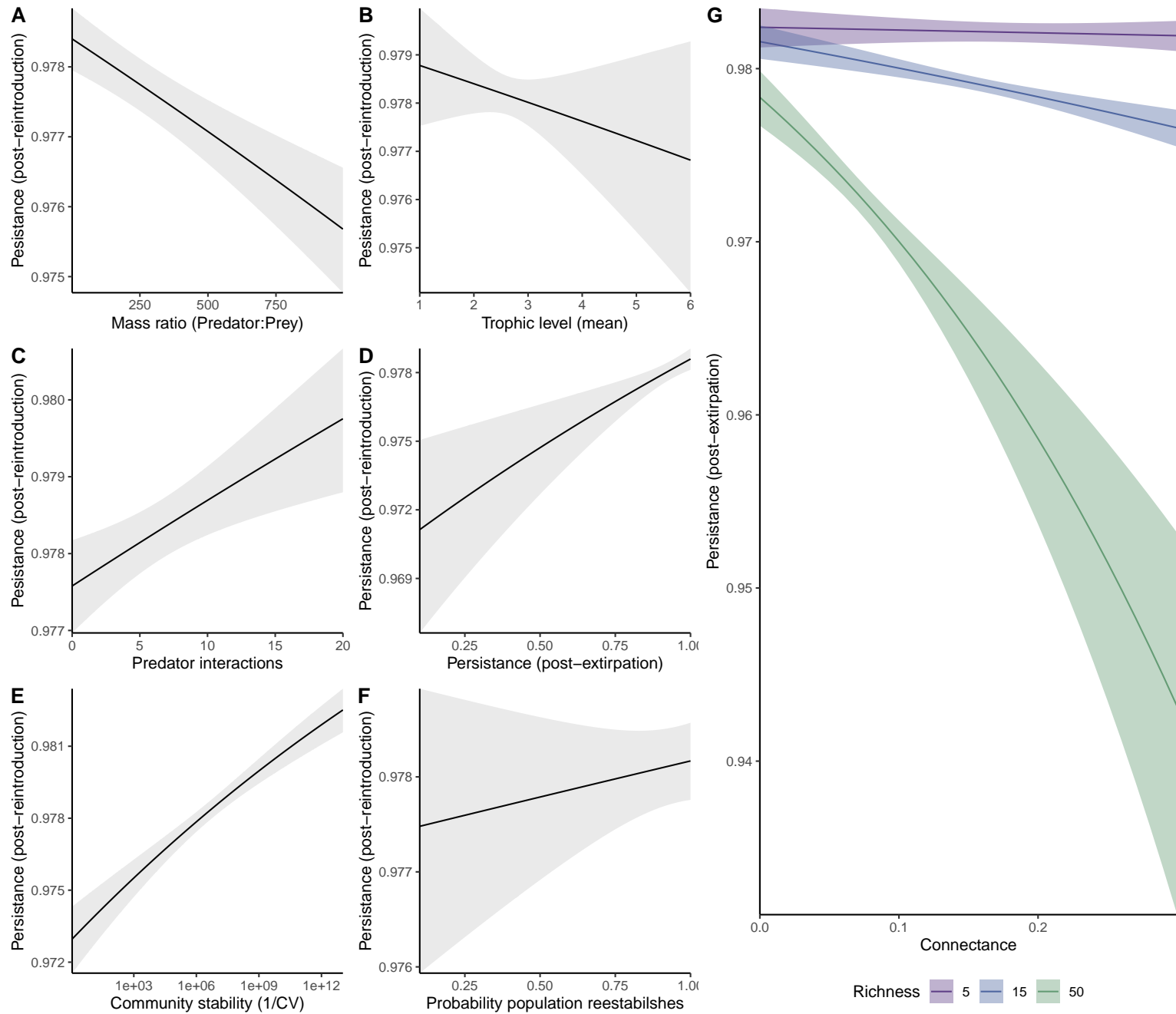
```

plt_m5_g = ggplot(m5_g) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  labs(x = "Predator interactions", y = "Persistence (post-reintroduction)") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic()

m5_c_s = ggpredict(m5, terms = c("realised_C_1 [0:0.3, by = 0.01]", "realised_S_1 [5, 15, 50]"))
plt_m5_c_s = ggplot(m5_c_s) +
  geom_line(aes(x = x, y = predicted, colour = group)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.4) +
  labs(x = "Connectance", y = "Persistence (post-extirpation)") +
  scale_fill_manual(values = palette_3, name = "Richness") +
  scale_colour_manual(values = palette_3, name = "Richness") +
  scale_x_continuous(expand = c(0,0)) +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(legend.position = "bottom")

ggarrange(
  ggarrange(plt_m5_z, plt_m5_tlvl, plt_m5_g, plt_m5_deltas_21, plt_m5_cv_com, plt_m5_deltasp_32, ncol = 2, nrow = 3, align = "h", labels = c("A",
plt_m5_c_s,
  ncol = 2, nrow = 1, widths = c(1,0.7), labels = c("", "G"))

```



## Model 6 - Change in community biomass post-extirpation

```
#Population re-establish stability?
m6 = lm(log(delta_bmc_21) ~ realised_C_1*realised_S_1 + g_1 + z_1 + stab_com_1 + tlvl_mean_1, data = sims)
summary(m6)
```

```
##
## Call:
## lm(formula = log(delta_bmc_21) ~ realised_C_1 * realised_S_1 +
##      g_1 + z_1 + stab_com_1 + tlvl_mean_1, data = sims)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.10440 -0.08519  0.03050  0.13175  1.47894
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      2.956e-01  1.948e-02  15.170 < 2e-16 ***
## realised_C_1     -1.863e-01  5.912e-02  -3.151  0.00163 **
## realised_S_1      1.166e-03  3.724e-04   3.132  0.00174 **
## g_1              -2.026e-02  5.294e-04 -38.265 < 2e-16 ***
## z_1              -1.138e-04  7.862e-06 -14.471 < 2e-16 ***
## stab_com_1       -1.457e-02  5.644e-04 -25.825 < 2e-16 ***
## tlvl_mean_1      -1.028e-01  5.636e-03 -18.243 < 2e-16 ***
## realised_C_1:realised_S_1  6.759e-02  3.630e-03  18.620 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2351 on 10961 degrees of freedom
## Multiple R-squared:  0.2454, Adjusted R-squared:  0.2449
## F-statistic: 509.1 on 7 and 10961 DF,  p-value: < 2.2e-16
```

```
m6_z = ggpredict(m6, "z_1 [1:1000, by = 5]", back_transform = F)
plt_m6_z = ggplot(m6_z) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Mass ratio (Predator:Prey)", y = "\u0394 Community biomass") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()
```

```

m6_tlvl = ggpredict(m6, "tlvl_mean_1 [1:6, by = 0.1]", back_transform = F)
plt_m6_tlvl = ggplot(m6_tlvl) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Trophic level (mean)", y = "\u0394 Community biomass") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

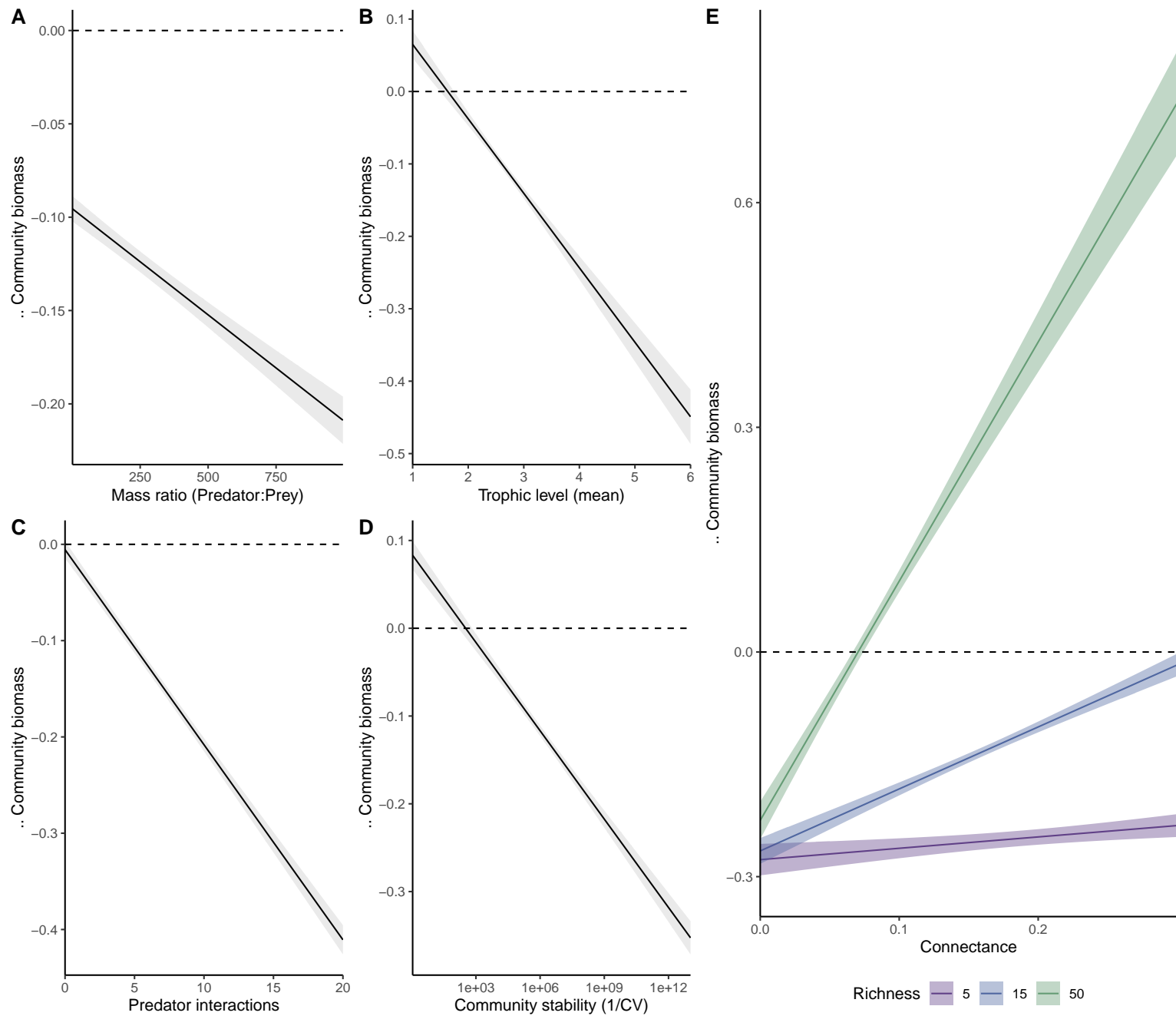
# Needs transformation
m6_cv_com = ggpredict(m6, "stab_com_1 [0.1:30, by = 0.1]", back_transform = F)
plt_m6_cv_com = ggplot(m6_cv_com) +
  geom_line(aes(x = exp(x), y = predicted)) +
  geom_ribbon(aes(x = exp(x), ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Community stability (1/CV)", y = "\u0394 Community biomass") +
  scale_x_log10(expand = c(0,0)) +
  theme_classic()

m6_g = ggpredict(m6, "g_1 [0:20, by = 0.1]", back_transform = F)
plt_m6_g = ggplot(m6_g) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Predator interactions", y = "\u0394 Community biomass") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

m6_c_s = ggpredict(m6, terms = c("realised_C_1 [0:0.3, by = 0.01]", "realised_S_1 [5, 15, 50]"), back_transform = F)
plt_m6_c_s = ggplot(m6_c_s) +
  geom_line(aes(x = x, y = predicted, colour = group)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.4) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Connectance", y = "\u0394 Community biomass") +
  scale_fill_manual(values = palette_3, name = "Richness") +
  scale_colour_manual(values = palette_3, name = "Richness") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(legend.position = "bottom")

```

```
ggarrange(  
  ggarrange(plt_m6_z, plt_m6_tlvl, plt_m6_g, plt_m6_cv_com, ncol = 2, nrow = 2, align = "h", labels = c("A", "B", "C", "D"), heights = c(1)),  
  plt_m6_c_s,  
  ncol = 2, nrow = 1, widths = c(1,0.7), labels = c("", "E"))
```



## Model 7 - Change in community biomass post-reintroduction

```
m7 = lm(log(delta_bmc_32) ~ realised_C_1*realised_S_1 + delta_s_21 + g_1 + z_1 + tlv1_mean_1 + delta_sp_32 + stab_com_2, data = sims)
summary(m7)
```

```
##
## Call:
## lm(formula = log(delta_bmc_32) ~ realised_C_1 * realised_S_1 +
##     delta_s_21 + g_1 + z_1 + tlv1_mean_1 + delta_sp_32 + stab_com_2,
##     data = sims)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.34284 -0.12480 -0.03076  0.07787  2.51077
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.204e-01  3.433e-02   9.333 < 2e-16 ***
## realised_C_1      3.806e-01  5.589e-02   6.811 1.02e-11 ***
## realised_S_1     -1.637e-03  3.508e-04  -4.667 3.09e-06 ***
## delta_s_21       -5.224e-01  2.791e-02 -18.718 < 2e-16 ***
## g_1              1.777e-02  5.080e-04  34.989 < 2e-16 ***
## z_1              9.188e-05  7.252e-06  12.669 < 2e-16 ***
## tlv1_mean_1      8.595e-02  5.365e-03  16.022 < 2e-16 ***
## delta_sp_32      8.905e-02  1.195e-02   7.452 9.91e-14 ***
## stab_com_2       1.689e-03  5.016e-04   3.368 0.00076 ***
## realised_C_1:realised_S_1 -7.511e-02  3.437e-03 -21.855 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2219 on 10959 degrees of freedom
## Multiple R-squared:  0.2361, Adjusted R-squared:  0.2355
## F-statistic: 376.4 on 9 and 10959 DF,  p-value: < 2.2e-16
```

```
m7_z = ggpredict(m7, "z_1 [1:1000, by = 5]", back_transform = F)
plt_m7_z = ggplot(m7_z) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Mass ratio (Predator:Prey)", y = "\u0394 Community biomass") +
```

```

scale_x_continuous(expand = c(0,0)) +
theme_classic()

m7_tlvl1 = ggpredict(m7, "tlvl_mean_1 [1:6, by = 0.1]", back_transform = F)
plt_m7_tlvl1 = ggplot(m7_tlvl1) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Trophic level (mean)", y = "\u0394 Community biomass") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

# Needs transformation
m7_cv_com = ggpredict(m7, "stab_com_2 [0.1:30, by = 0.1]", back_transform = F)
plt_m7_cv_com = ggplot(m7_cv_com) +
  geom_line(aes(x = exp(x), y = predicted)) +
  geom_ribbon(aes(x = exp(x), ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Community stability (1/CV)", y = "\u0394 Community biomass") +
  scale_x_log10(expand = c(0,0)) +
  theme_classic()

m7_deltas_21 = ggpredict(m7, "delta_s_21 [0.1:1, by = 0.01]", back_transform = F)
plt_m7_deltas_21 = ggplot(m7_deltas_21) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Persistence (post-extirpation)", y = "\u0394 Community biomass") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

m7_deltasp_32 = ggpredict(m7, "delta_sp_32 [0.1:1, by = 0.01]", back_transform = F)
plt_m7_deltasp_32 = ggplot(m7_deltasp_32) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Probability population reestablishes", y = "\u0394 Community biomass") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

m7_g = ggpredict(m7, "g_1 [0:20, by = 0.1]", back_transform = F)

```

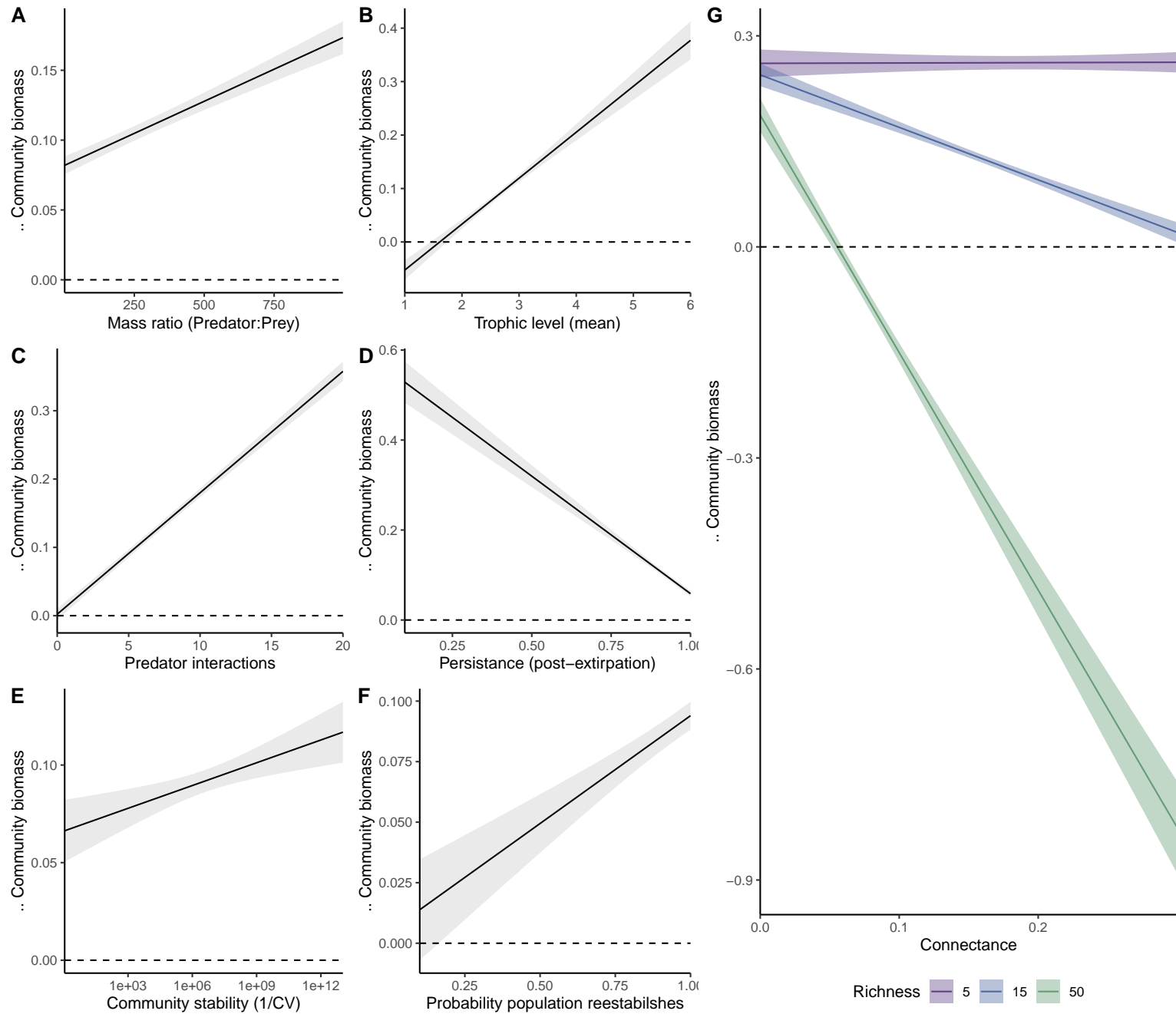
```

plt_m7_g = ggplot(m7_g) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Predator interactions", y = "\u0394 Community biomass") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

m7_c_s = ggpredict(m7, terms = c("realised_C_1 [0:0.3, by = 0.01]", "realised_S_1 [5, 15, 50]"), back_transform = F)
plt_m7_c_s = ggplot(m7_c_s) +
  geom_line(aes(x = x, y = predicted, colour = group)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.4) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Connectance", y = "\u0394 Community biomass") +
  scale_fill_manual(values = palette_3, name = "Richness") +
  scale_colour_manual(values = palette_3, name = "Richness") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(legend.position = "bottom")

ggarrange(
  ggarrange(plt_m7_z, plt_m7_tlvl, plt_m7_g, plt_m7_deltas_21, plt_m7_cv_com, plt_m7_deltasp_32, ncol = 2, nrow = 3, align = "h", labels = c("A",
plt_m7_c_s,
  ncol = 2, nrow = 1, widths = c(1,0.7), labels = c("", "G"))

```



## Model 8 - Change in community stability post-extirpation

```
#Population re-establish stability?
m8 = lm(log(delta_stabc_21) ~ realised_C_1*realised_S_1 + g_1 + z_1 + stab_com_1 + tlvl_mean_1, data = sims)
summary(m8)

##
## Call:
## lm(formula = log(delta_stabc_21) ~ realised_C_1 * realised_S_1 +
##     g_1 + z_1 + stab_com_1 + tlvl_mean_1, data = sims)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.9453  -2.4566  -0.0581   2.4704  14.1978
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    15.0696960   0.3210290   46.942 < 2e-16 ***
## realised_C_1     5.6412348   0.9742157    5.791 7.21e-09 ***
## realised_S_1    -0.0138951   0.0061355   -2.265  0.0235 *
## g_1            -0.0766990   0.0087234   -8.792 < 2e-16 ***
## z_1            -0.0010868   0.0001296   -8.389 < 2e-16 ***
## stab_com_1     -0.5672360   0.0092991  -60.999 < 2e-16 ***
## tlvl_mean_1    -1.5498226   0.0928679  -16.688 < 2e-16 ***
## realised_C_1:realised_S_1 -0.6563143   0.0598165  -10.972 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.874 on 10961 degrees of freedom
## Multiple R-squared:  0.2762, Adjusted R-squared:  0.2757
## F-statistic: 597.4 on 7 and 10961 DF,  p-value: < 2.2e-16

m8_z = ggpredict(m8, "z_1 [1:1000, by = 5]", back_transform = F)
plt_m8_z = ggplot(m8_z) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Mass ratio (Predator:Prey)", y = "\u0394 Community stability") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()
```

```

m8_tlvl = ggpredict(m8, "tlvl_mean_1 [1:6, by = 0.1]", back_transform = F)
plt_m8_tlvl = ggplot(m8_tlvl) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Trophic level (mean)", y = "\u0394 Community stability") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

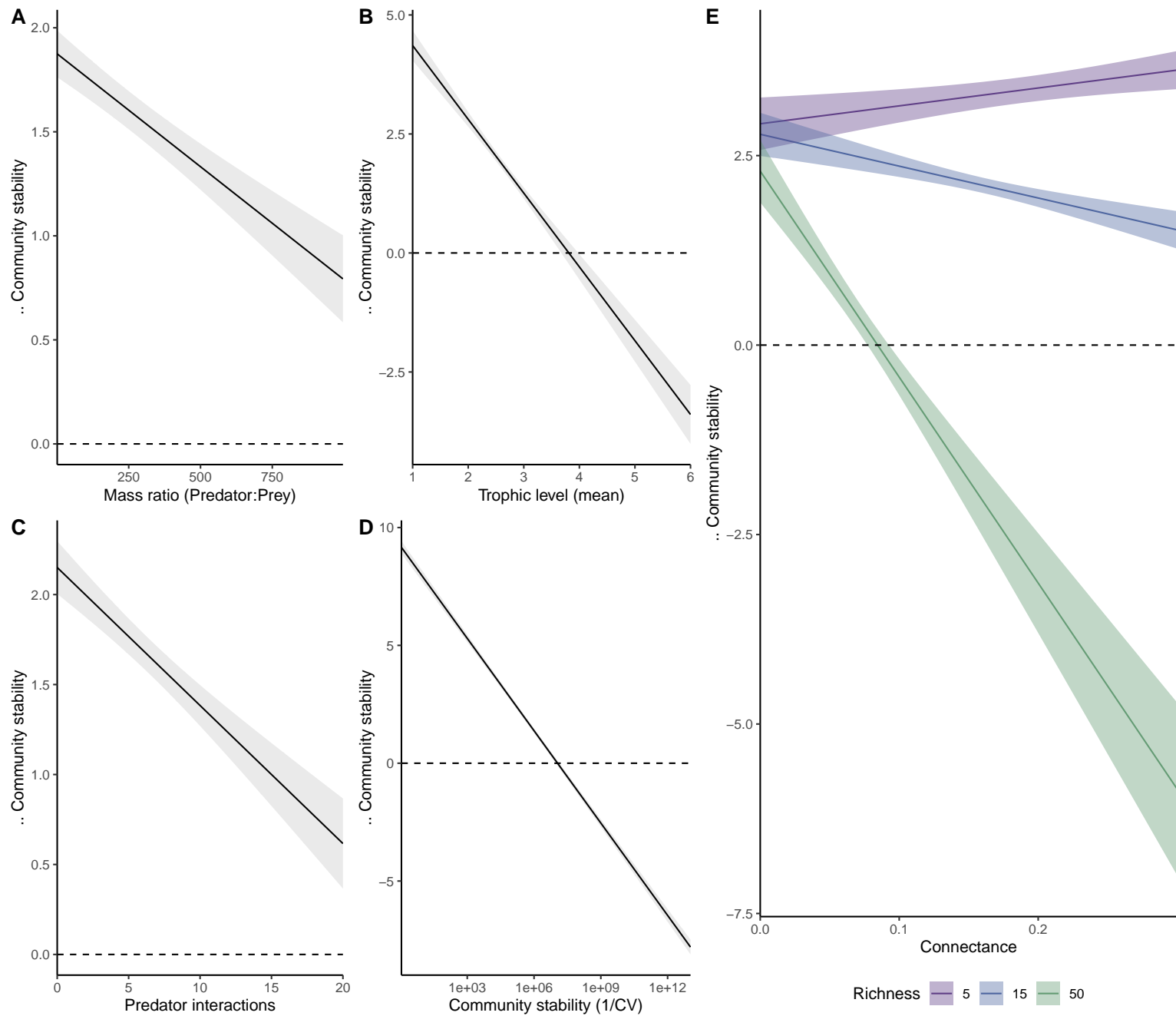
# Needs transformation
m8_cv_com = ggpredict(m8, "stab_com_1 [0.1:30, by = 0.1]", back_transform = F)
plt_m8_cv_com = ggplot(m8_cv_com) +
  geom_line(aes(x = exp(x), y = predicted)) +
  geom_ribbon(aes(x = exp(x), ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Community stability (1/CV)", y = "\u0394 Community stability") +
  scale_x_log10(expand = c(0,0)) +
  theme_classic()

m8_g = ggpredict(m8, "g_1 [0:20, by = 0.1]", back_transform = F)
plt_m8_g = ggplot(m8_g) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Predator interactions", y = "\u0394 Community stability") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

m8_c_s = ggpredict(m8, terms = c("realised_C_1 [0:0.3, by = 0.01]", "realised_S_1 [5, 15, 50]"), back_transform = F)
plt_m8_c_s = ggplot(m8_c_s) +
  geom_line(aes(x = x, y = predicted, colour = group)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.4) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Connectance", y = "\u0394 Community stability") +
  scale_fill_manual(values = palette_3, name = "Richness") +
  scale_colour_manual(values = palette_3, name = "Richness") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(legend.position = "bottom")

```

```
ggarrange(  
  ggarrange(plt_m8_z, plt_m8_tlvl, plt_m8_g, plt_m8_cv_com, ncol = 2, nrow = 2, align = "h", labels = c("A", "B", "C", "D"), heights = c(1)),  
  plt_m8_c_s,  
  ncol = 2, nrow = 1, widths = c(1,0.7), labels = c("", "E"))
```



## Model 9 - Change in community stability post-reintroduction

```
m9 = lm(log(delta_stabc_32) ~ realised_C_1*realised_S_1 + delta_s_21 + g_1 + z_1 + tlv1_mean_1 + delta_sp_32 + stab_com_2, data = sims)
summary(m9)
```

```
##
## Call:
## lm(formula = log(delta_stabc_32) ~ realised_C_1 * realised_S_1 +
##     delta_s_21 + g_1 + z_1 + tlv1_mean_1 + delta_sp_32 + stab_com_2,
##     data = sims)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.3696  -2.5764   0.0764   2.5542  15.3545
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      17.9475822   0.5940386  30.213  <2e-16 ***
## realised_C_1      -1.4900021   0.9670935  -1.541    0.123
## realised_S_1      -0.0064243   0.0060707  -1.058    0.290
## delta_s_21       -5.3125451   0.4830032 -10.999  <2e-16 ***
## g_1               0.1103430   0.0087904  12.553  <2e-16 ***
## z_1              -0.0033898   0.0001255 -27.010  <2e-16 ***
## tlv1_mean_1      -1.1711224   0.0928349 -12.615  <2e-16 ***
## delta_sp_32       0.0125349   0.2068073   0.061    0.952
## stab_com_2       -0.5340021   0.0086805 -61.518  <2e-16 ***
## realised_C_1:realised_S_1 -0.4996927  0.0594779  -8.401  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.84 on 10959 degrees of freedom
## Multiple R-squared:  0.3135, Adjusted R-squared:  0.3129
## F-statistic: 556 on 9 and 10959 DF, p-value: < 2.2e-16
```

```
m9_z = ggpredict(m9, "z_1 [1:1000, by = 5]", back_transform = F)
plt_m9_z = ggplot(m9_z) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Mass ratio (Predator:Prey)", y = "\u0394 Community stability") +
```

```

scale_x_continuous(expand = c(0,0)) +
theme_classic()

m9_tlvl1 = ggpredict(m9, "tlvl_mean_1 [1:6, by = 0.1]", back_transform = F)
plt_m9_tlvl1 = ggplot(m9_tlvl1) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Trophic level (mean)", y = "\u0394 Community stability") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

# Needs transformation
m9_cv_com = ggpredict(m9, "stab_com_2 [0.1:30, by = 0.1]", back_transform = F)
plt_m9_cv_com = ggplot(m9_cv_com) +
  geom_line(aes(x = exp(x), y = predicted)) +
  geom_ribbon(aes(x = exp(x), ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Community stability (1/CV)", y = "\u0394 Community stability") +
  scale_x_log10(expand = c(0,0)) +
  theme_classic()

m9_deltas_21 = ggpredict(m9, "delta_s_21 [0.1:1, by = 0.01]", back_transform = F)
plt_m9_deltas_21 = ggplot(m9_deltas_21) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Persistence (post-extirpation)", y = "\u0394 Community stability") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

m9_deltasp_32 = ggpredict(m9, "delta_sp_32 [0.1:1, by = 0.01]", back_transform = F)
plt_m9_deltasp_32 = ggplot(m9_deltasp_32) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Probability population reestablishes", y = "\u0394 Community stability") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

m9_g = ggpredict(m9, "g_1 [0:20, by = 0.1]", back_transform = F)

```

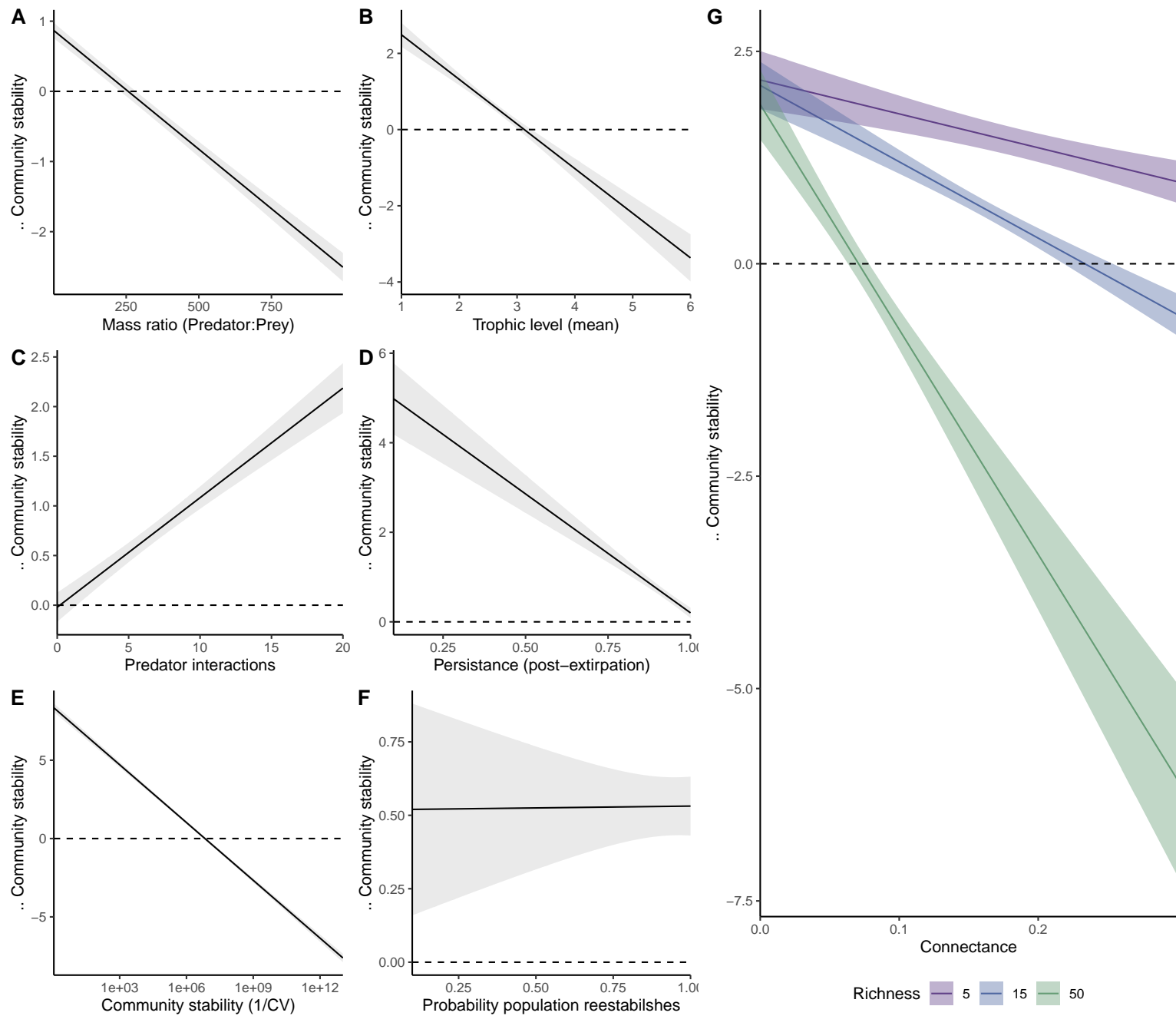
```

plt_m9_g = ggplot(m9_g) +
  geom_line(aes(x = x, y = predicted)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high), alpha = 0.1) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Predator interactions", y = "\u0394 Community stability") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic()

m9_c_s = ggpredict(m9, terms = c("realised_C_1 [0:0.3, by = 0.01]", "realised_S_1 [5, 15, 50]"), back_transform = F)
plt_m9_c_s = ggplot(m9_c_s) +
  geom_line(aes(x = x, y = predicted, colour = group)) +
  geom_ribbon(aes(x = x, ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.4) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  labs(x = "Connectance", y = "\u0394 Community stability") +
  scale_fill_manual(values = palette_3, name = "Richness") +
  scale_colour_manual(values = palette_3, name = "Richness") +
  scale_x_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(legend.position = "bottom")

ggarrange(
  ggarrange(plt_m9_z, plt_m9_tlvl, plt_m9_g, plt_m9_deltas_21, plt_m9_cv_com, plt_m9_deltasp_32, ncol = 2, nrow = 3, align = "h", labels = c("A",
    plt_m9_c_s,
    ncol = 2, nrow = 1, widths = c(1,0.7), labels = c("", "G"))

```



## Synthesise - Synthesis of effect size

```
synth = data.frame(  
  model = c(  
    rep("Probability population reestablishes", 8),  
    rep("\u0394 Population stability", 8),  
    rep("\u0394 Predator biomass", 8),  
    rep("Persistence (post-extirpation)", 7),  
    rep("Persistence (post-reintroduction)", 9),  
    rep("\u0394 Community biomass (post-extirpation)", 7),  
    rep("\u0394 Community biomass (post-reintroduction)", 9),  
    rep("\u0394 Community stability (post-extirpation)", 7),  
    rep("\u0394 Community stability (post-reintroduction)", 9)  
  ),  
  variable = c(  
    names(summary(m1)$coefficients[-1,3]),  
    names(summary(m2)$coefficients[-1,3]),  
    names(summary(m3)$coefficients[-1,3]),  
    names(summary(m4)$p.coef[-1]),  
    names(summary(m5)$p.coef[-1]),  
    names(summary(m6)$coefficients[-1,3]),  
    names(summary(m7)$coefficients[-1,3]),  
    names(summary(m8)$coefficients[-1,3]),  
    names(summary(m9)$coefficients[-1,3])  
  ),  
  coef = c(  
    summary(m1)$coefficients[-1,1],  
    summary(m2)$coefficients[-1,1],  
    summary(m3)$coefficients[-1,1],  
    summary(m4)$p.coef[-1],  
    summary(m5)$p.coef[-1],  
    summary(m6)$coefficients[-1,1],  
    summary(m7)$coefficients[-1,1],  
    summary(m8)$coefficients[-1,1],  
    summary(m9)$coefficients[-1,1]  
  ),  
  error = c(  
    summary(m1)$coefficients[-1,2],  
    summary(m2)$coefficients[-1,2],  
    summary(m3)$coefficients[-1,2],  
    summary(m4)$p.coef[-1],  
    summary(m5)$p.coef[-1],  
    summary(m6)$coefficients[-1,2],  
    summary(m7)$coefficients[-1,2],  
    summary(m8)$coefficients[-1,2],  
    summary(m9)$coefficients[-1,2]
```

```

summary(m3)$coefficients[-1,2],
summary(m4)$se[-1],
summary(m5)$se[-1],
summary(m6)$coefficients[-1,2],
summary(m7)$coefficients[-1,2],
summary(m8)$coefficients[-1,2],
summary(m9)$coefficients[-1,2]
)

)
synth$test_stat = synth$coef/synth$error
synth$test_stat2 = synth$test_stat
synth$test_stat2 = ifelse(synth$test_stat2 > 10, 10, synth$test_stat2)
synth$test_stat2 = ifelse(synth$test_stat2 < -10, -10, synth$test_stat2)

synth2 = expand.grid(model = unique(synth$model), variable = unique(synth$variable))
synth2 = left_join(synth2, synth)

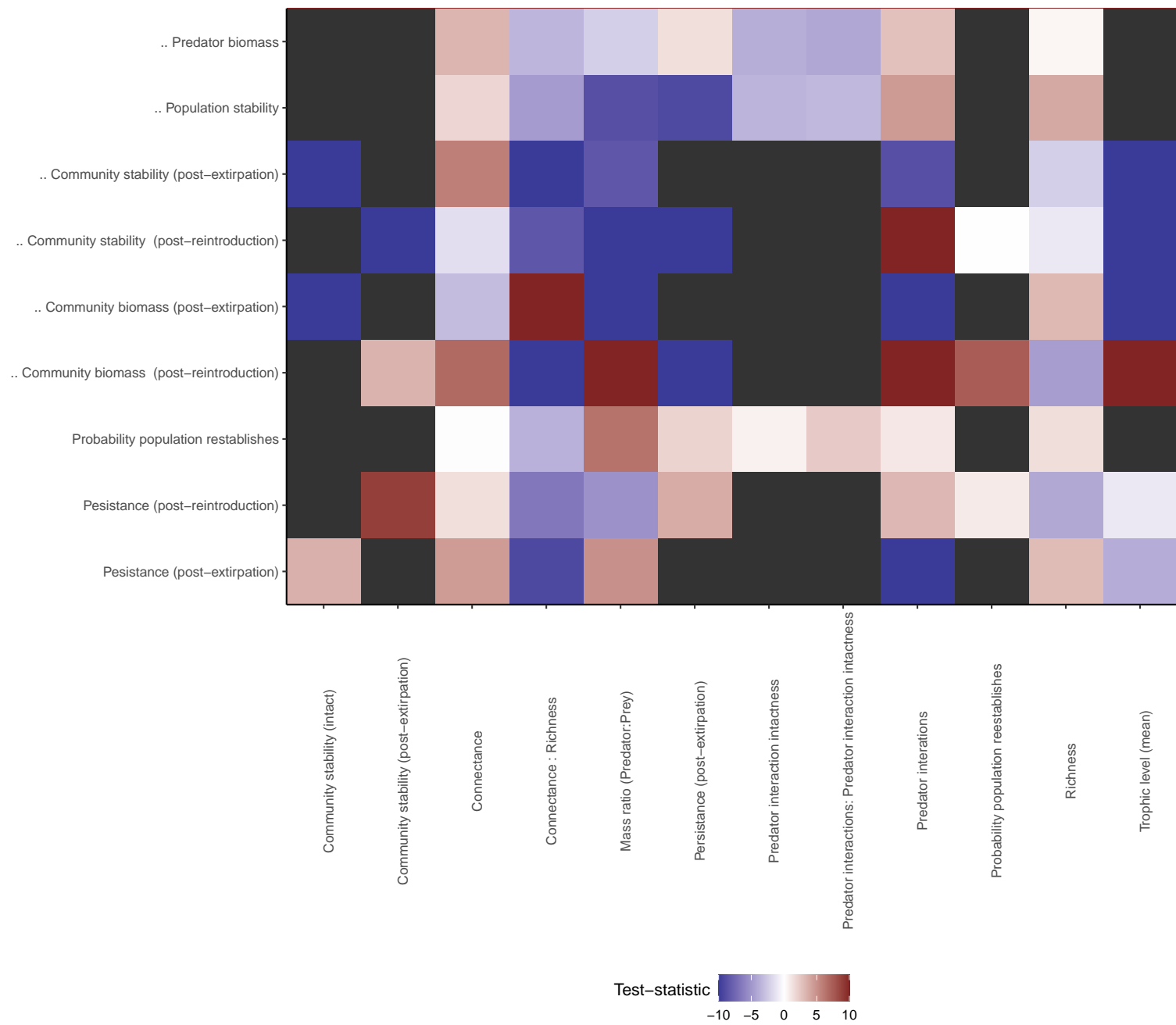
variable_rename = data.frame(
  variable = unique(synth2$variable),
  variable2 = c("Connectance",
    "Richness",
    "Predator interactions",
    "Predator interaction intactness",
    "Persistence (post-extirpation)",
    "Mass ratio (Predator:Prey)",
    "Connectance : Richness",
    "Predator interactions: Predator interaction intactness",
    "Community stability (intact)",
    "Trophic level (mean)",
    "Probability population reestablishes",
    "Community stability (post-extirpation)"
  )
)

synth2 = left_join(synth2, variable_rename)

ggplot(synth2) +
  geom_tile(aes(x = variable2, y = model, fill = test_stat2)) +
  scale_fill_gradient2(low = scales::muted("darkblue"), high = scales::muted("darkred"), na.value = "grey20", mid = "white", midpoint = 0, breaks
  scale_x_discrete(expand = c(0,0)) +
  scale_y_discrete(expand = c(0,0)) +

```

```
labs(x = "", y = "") +  
theme_classic() +  
theme(axis.text.x = element_text(angle = 90),  
      panel.background = element_rect(fill = scales::muted("darkred")),  
      legend.position = "bottom")
```



```

scen_df = data.frame(
  scen = c("Tall", "Short", "Large", "Small", "Large", "Small", "Intact", "Depleted", "Strong", "Weak", "High", "Low", "High", "Low"),
  fac = c("Food-web\ntrophic-height", "Food-web\ntrophic-height", "Mass ratio\nPredator:Prey", "Mass ratio\nPredator:Prey", "Predator\ndiet-breadth", "Predator\ndiet-breadth", "Predator\ndiet-breadth", "Predator\ndiet-breadth", "Predator\ndiet-breadth", "Predator\ndiet-breadth", "Predator\ndiet-breadth", "Predator\ndiet-breadth", "Predator\ndiet-breadth", "Predator\ndiet-breadth"),
  tlvl_mean_1 = c(quantile(sims$tlvl_mean_1, probs = 0.1), quantile(sims$tlvl_mean_1, probs = 0.9), rep(mean(sims$tlvl_mean_1), 12)),
  z_1 = c(rep(mean(sims$z_1), 2), quantile(sims$z_1, probs = 0.9), quantile(sims$z_1, probs = 0.1), rep(mean(sims$z_1), 10)),
  g_1 = c(rep(mean(sims$g_1), 4), quantile(sims$g_1, probs = 0.9), quantile(sims$g_1, probs = 0.1), rep(mean(sims$g_1), 8)),
  delta_s_21 = c(rep(mean(sims$delta_s_21), 6), 1, 0.5, rep(mean(sims$delta_s_21), 6)),
  realised_C_1 = c(rep(mean(sims$realised_C_1), 8), quantile(sims$realised_C_1, probs = 0.3), quantile(sims$realised_C_1, probs = 0.1), rep(mean(sims$realised_C_1), 10)),
  realised_S_1 = c(rep(50, 14)),
  delta_sp_32 = c(rep(mean(sims$delta_sp_32), 10), quantile(sims$delta_sp_32, probs = 0.1), quantile(sims$delta_sp_32, probs = 0.9), rep(mean(sims$delta_sp_32), 10)),
  delta_g_31 = c(rep(mean(sims$delta_g_31), 12), 1, 0.5),
  stab_com_1 = c(rep(mean(sims$stab_com_1), 14)),
  stab_com_2 = c(rep(mean(sims$stab_com_2), 14))
)

m1_pred = predict(m1, newdata = scen_df, se.fit = TRUE)
scen_df$m1_pred = predict(m1, newdata = scen_df, se.fit = F, type = "response")
scen_df$m1_low = with(m1_pred, plogis(fit + qnorm(0.025)*se.fit))
scen_df$m1_upr = with(m1_pred, plogis(fit + qnorm(0.975)*se.fit))

m2_pred = predict(m2, newdata = scen_df, se.fit=TRUE, interval="confidence", level=0.95)
scen_df$m2_pred = m2_pred$fit[,1]
scen_df$m2_low = m2_pred$fit[,2]
scen_df$m2_upr = m2_pred$fit[,3]

scen_df$var_mn = exp(scen_df$m2_pred) * 0.1 #Mean stab = 0.1 / CV = 10%
scen_df$var_low = exp(scen_df$m2_low) * 0.1
scen_df$var_upr = exp(scen_df$m2_upr) * 0.1

m3_pred = predict(m3, newdata = scen_df, se.fit=TRUE, interval="confidence", level=0.95)
scen_df$m3_pred = m3_pred$fit[,1]
scen_df$m3_low = m3_pred$fit[,2]
scen_df$m3_upr = m3_pred$fit[,3]

scen_df$ab_mn = exp(scen_df$m3_pred) * 10 #Mean abundance = 1
scen_df$ab_low = exp(scen_df$m3_low) * 10
scen_df$ab_upr = exp(scen_df$m3_upr) * 10

m4_pred = predict(m4, newdata = scen_df, se.fit = TRUE)
scen_df$m4_pred = as.vector(predict(m4, newdata = scen_df, se.fit = F, type = "response"))
scen_df$m4_low = as.vector(with(m4_pred, plogis(fit + qnorm(0.025)*se.fit)))

```

```

scen_df$m4_upr = as.vector(with(m4_pred, plogis(fit + qnorm(0.975)*se.fit)))

scen_df$rich_mn = scen_df$m4_pred * 50 #50 species
scen_df$rich_low = scen_df$m4_low * 50
scen_df$rich_upr = scen_df$m4_upr * 50

m5_pred = predict(m5, newdata = scen_df, se.fit = TRUE)
scen_df$m5_pred = predict(m5, newdata = scen_df, se.fit = F, type = "response")
scen_df$m5_low = with(m5_pred, plogis(fit + qnorm(0.025)*se.fit))
scen_df$m5_upr = with(m5_pred, plogis(fit + qnorm(0.975)*se.fit))

scen_df$rich2_mn = scen_df$m5_pred * scen_df$rich_mn
scen_df$rich2_low = scen_df$m5_low * scen_df$rich_low
scen_df$rich2_upr = scen_df$m5_upr * scen_df$rich_upr

m6_pred = predict(m6, newdata = scen_df, se.fit=TRUE, interval="confidence", level=0.95)
scen_df$m6_pred = m6_pred$fit[,1]
scen_df$m6_low = m6_pred$fit[,2]
scen_df$m6_upr = m6_pred$fit[,3]

scen_df$bio_mn = exp(scen_df$m6_pred) * 100 #100kg biomass
scen_df$bio_low = exp(scen_df$m6_low) * 100
scen_df$bio_upr = exp(scen_df$m6_upr) * 100

m7_pred = predict(m7, newdata = scen_df, se.fit=TRUE, interval="confidence", level=0.95)
scen_df$m7_pred = m7_pred$fit[,1]
scen_df$m7_low = m7_pred$fit[,2]
scen_df$m7_upr = m7_pred$fit[,3]

scen_df$bio2_mn = exp(scen_df$m7_pred) * scen_df$bio_mn
scen_df$bio2_low = exp(scen_df$m7_low) * scen_df$bio_low
scen_df$bio2_upr = exp(scen_df$m7_upr) * scen_df$bio_upr

m8_pred = predict(m8, newdata = scen_df, se.fit=TRUE, interval="confidence", level=0.95)
scen_df$m8_pred = m8_pred$fit[,1]
scen_df$m8_low = m8_pred$fit[,2]
scen_df$m8_upr = m8_pred$fit[,3]

scen_df$stab_mn = exp(scen_df$m8_pred) * 0.1 #0.1 stability is 10%cv average
scen_df$stab_low = exp(scen_df$m8_low) * 0.1
scen_df$stab_upr = exp(scen_df$m8_upr) * 0.1

```

```

m9_pred = predict(m9, newdata = scen_df, se.fit=TRUE, interval="confidence", level=0.95)
scen_df$m9_pred = m9_pred$fit[,1]
scen_df$m9_low = m9_pred$fit[,2]
scen_df$m9_upr = m9_pred$fit[,3]

scen_df$stab2_mn = exp(scen_df$m9_pred) * scen_df$stab_mn #Mean abundance = 1
scen_df$stab2_low = exp(scen_df$m9_low) * scen_df$stab_low
scen_df$stab2_upr = exp(scen_df$m9_upr) * scen_df$stab_upr

plt_a = ggplot() +
  geom_point(aes(x = "Average\ncommunity", y = 50), colour = "grey30", size = 2.5) +
  geom_hline(aes(yintercept = 50), linetype = "dashed") +
  labs(x = "", y = "Species richness", title = "Intact") +
  scale_y_continuous(expand = c(0,0), limits = c(40,51)) +
  facet_grid(. ~ "", scales = "free") +
  theme_classic() +
  theme(strip.background = element_rect(color="white", fill="white", size=1.5, linetype="solid"))

plt_b = ggplot(data = scen_df) +
  geom_pointrange(aes(x = scen, y = rich_mn, ymin = rich_low, ymax = rich_upr, colour = fac), size = 0.5) +
  geom_hline(aes(yintercept = 50), linetype = "dashed") +
  labs(x = "", y = "", title = "Post-extirpation") +
  scale_y_continuous(expand = c(0,0), limits = c(40,51)) +
  scale_colour_manual(values = palette_7, name = "", guide = F) +
  facet_grid(. ~ fac, drop = T, scales = "free") +
  theme_classic() +
  theme(
    legend.position = "bottom",
    axis.text.y = element_blank(),
    axis.line.y = element_blank(),
    axis.ticks.y = element_blank(),
    strip.background = element_rect(color="white",size=1.5, linetype="solid")
  )

plt_c = ggplot() +
  geom_point(aes(x = "Average\ncommunity", y = 100), colour = "grey30", size = 2.5) +
  geom_hline(aes(yintercept = 100), linetype = "dashed") +
  labs(x = "", y = "Biomass", title = "") +

```

```

scale_y_continuous(expand = c(0,0), limits = c(80,140)) +
facet_grid(. ~ "", scales = "free") +
theme_classic() +
theme(strip.background = element_rect(color="white", fill="white", size=1.5, linetype="solid"))

plt_d = ggplot(data = scen_df) +
  geom_pointrange(aes(x = scen, y = bio_mn, ymin = bio_low, ymax = bio_upr, colour = fac), size = 0.5) +
  geom_hline(aes(yintercept = 100), linetype = "dashed") +
  labs(x = "", y = "", title = "") +
  scale_y_continuous(expand = c(0,0), limits = c(80,140)) +
  scale_colour_manual(values = palette_7, name = "", guide = F) +
  facet_grid(. ~ fac, drop = T, scales = "free") +
  theme_classic() +
  theme(
    legend.position = "bottom",
    axis.text.y = element_blank(),
    axis.line.y = element_blank(),
    axis.ticks.y = element_blank(),
    strip.background = element_rect(color="white",size=1.5, linetype="solid"),
    strip.text.x = element_text(
      size = 2, color = "white", face = "bold.italic"
    )
  )

plt_e = ggplot() +
  geom_point(aes(x = "Average\ncommunity", y = 0.1), colour = "grey30", size = 2.5) +
  geom_hline(aes(yintercept = 0.1), linetype = "dashed") +
  labs(x = "", y = "Stability", title = "") +
  scale_y_log10(expand = c(0,0), limits = c(0.00001, 1), breaks = c(1e-4, 1e-3, 1e-2, 1e-1,1), labels=function(x) parse(text=paste("10^",round(log
  facet_grid(. ~ "", scales = "free") +
  theme_classic() +
  theme(strip.background = element_rect(color="white", fill="white", size=1.5, linetype="solid"))

plt_f = ggplot(data = scen_df) +
  geom_pointrange(aes(x = scen, y = stab_mn, ymin = stab_low, ymax = stab_upr, colour = fac), size = 0.5) +
  geom_hline(aes(yintercept = 0.1), linetype = "dashed") +
  labs(x = "", y = "", title = "") +
  scale_y_log10(expand = c(0,0), limits = c(0.00001, 1)) +
  scale_colour_manual(values = palette_7, name = "", guide = F) +
  facet_grid(. ~ fac, drop = T, scales = "free") +
  theme_classic() +

```

```

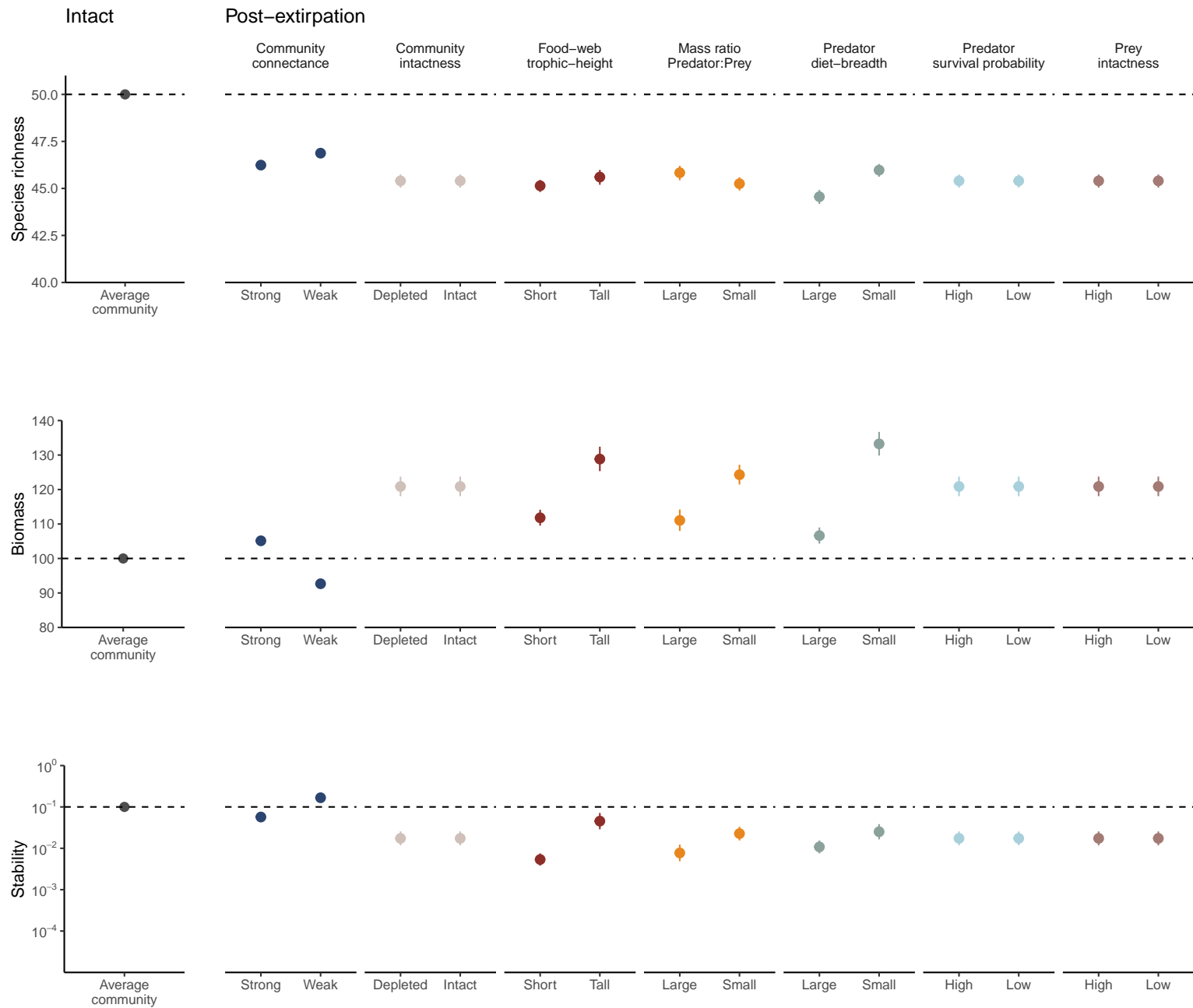
theme(
  legend.position = "bottom",
  axis.text.y = element_blank(),
  axis.line.y = element_blank(),
  axis.ticks.y = element_blank(),
  strip.background = element_rect(color="white",size=1.5, linetype="solid"),
  strip.text.x = element_text(
    size = 2, color = "white", face = "bold.italic"
  )
)

```

```

ggarrange(plt_a, plt_b, plt_c, plt_d, plt_e, plt_f, ncol = 2, nrow = 3, widths = c(1.5, 8), align = "hv", common.legend = T, legend = "bottom")

```



```

plt_a = ggplot() +
  geom_point(aes(x = "Average\npredator", y = 1), colour = "grey30", size = 2.5) +
  geom_hline(aes(yintercept = 1), linetype = "dashed") +
  labs(x = "", y = "Probability predator\nis extant", title = "Intact") +
  scale_y_continuous(expand = c(0,0), limits = c(0.8,1.01)) +
  facet_grid(. ~ "", scales = "free") +
  theme_classic() +
  theme(strip.background = element_rect(color="white", fill="white", size=1.5, linetype="solid"))

plt_b = ggplot(data = scen_df) +
  geom_pointrange(aes(x = scen, y = m1_pred, ymin = m1_low, ymax = m1_upr, colour = fac), size = 0.5) +
  geom_hline(aes(yintercept = 1), linetype = "dashed") +
  labs(x = "", y = "", title = "Post-reintroduction") +
  scale_y_continuous(expand = c(0,0), limits = c(0.8,1.01)) +
  scale_colour_manual(values = palette_7, name = "", guide = F) +
  facet_grid(. ~ fac, drop = T, scales = "free") +
  theme_classic() +
  theme(
    legend.position = "bottom",
    axis.text.y = element_blank(),
    axis.line.y = element_blank(),
    axis.ticks.y = element_blank(),
    strip.background = element_rect(color="white",size=1.5, linetype="solid")
  )

plt_c = ggplot() +
  geom_point(aes(x = "Average\npredator", y = 10), colour = "grey30", size = 2.5) +
  geom_hline(aes(yintercept = 10), linetype = "dashed") +
  labs(x = "", y = "Predator\nabundance", title = "") +
  scale_y_continuous(expand = c(0,0), limits = c(5,12)) +
  facet_grid(. ~ "", scales = "free") +
  theme_classic() +
  theme(strip.background = element_rect(color="white", fill="white", size=1.5, linetype="solid"))

plt_d = ggplot(data = scen_df) +
  geom_pointrange(aes(x = scen, y = ab_mn, ymin = ab_low, ymax = ab_upr, colour = fac), size = 0.5) +
  geom_hline(aes(yintercept = 10), linetype = "dashed") +
  labs(x = "", y = "", title = "") +

```

```

scale_y_continuous(expand = c(0,0), limits = c(5, 12)) +
scale_colour_manual(values = palette_7, name = "", guide = F) +
facet_grid(. ~ fac, drop = T, scales = "free") +
theme_classic() +
theme(
  legend.position = "bottom",
  axis.text.y = element_blank(),
  axis.line.y = element_blank(),
  axis.ticks.y = element_blank(),
  strip.background = element_rect(color="white",size=1.5, linetype="solid"),
  strip.text.x = element_text(
    size = 2, color = "white", face = "bold.italic"
  )
)

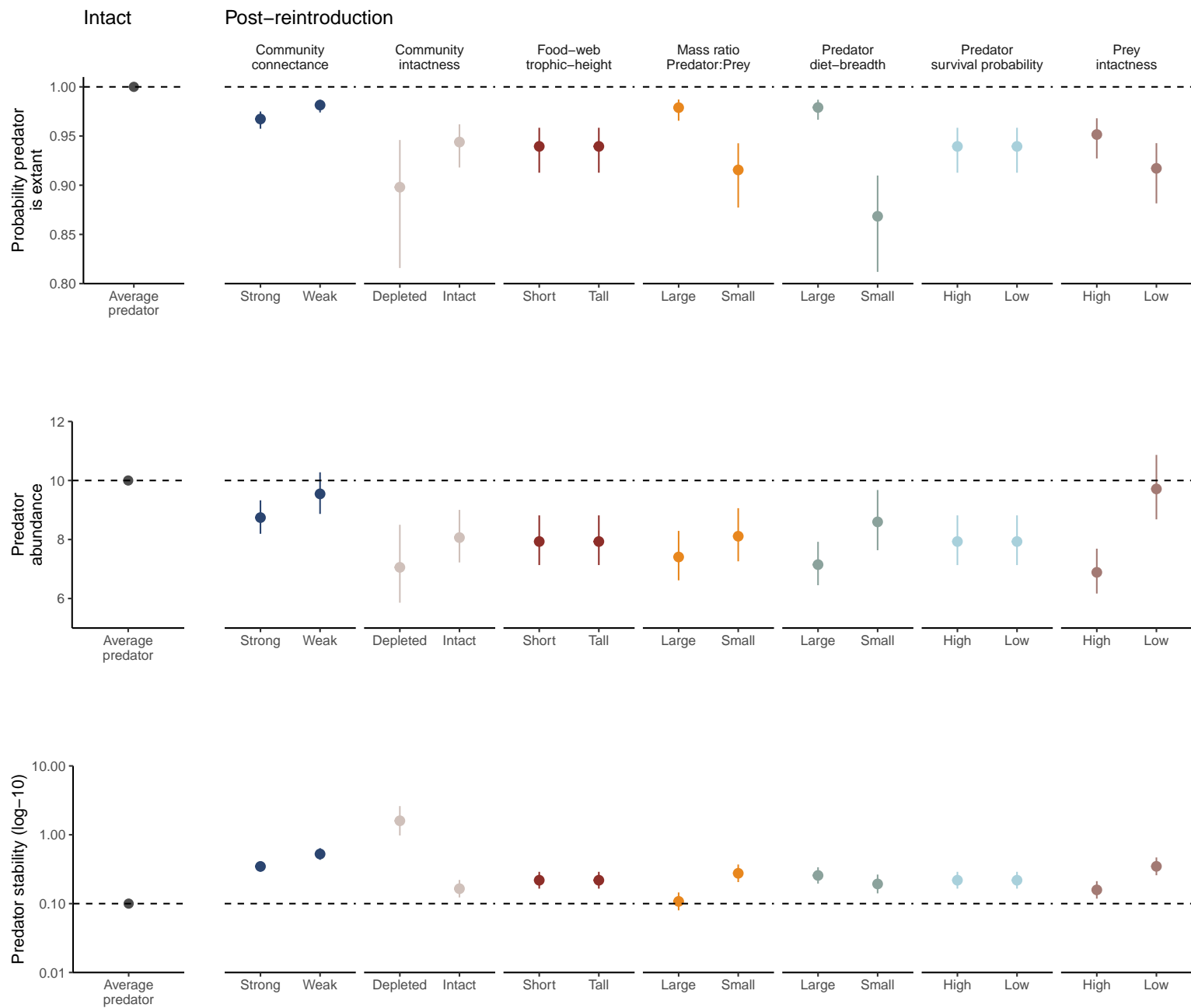
plt_e = ggplot() +
  geom_point(aes(x = "Average\npredator", y = 0.1), colour = "grey30", size = 2.5) +
  geom_hline(aes(yintercept = 0.1), linetype = "dashed") +
  labs(x = "", y = "Predator stability (log-10)", title = "") +
  scale_y_log10(expand = c(0,0), limits = c(0.01,10)) +
  facet_grid(. ~ "", scales = "free") +
  theme_classic() +
  theme(strip.background = element_rect(color="white", fill="white", size=1.5, linetype="solid"))

plt_f = ggplot(data = scen_df) +
  geom_pointrange(aes(x = scen, y = var_mn, ymin = var_low, ymax = var_upr, colour = fac), size = 0.5) +
  geom_hline(aes(yintercept = 0.1), linetype = "dashed") +
  labs(x = "", y = "", title = "") +
  scale_y_log10(expand = c(0,0), limits = c(0.01, 10)) +
  scale_colour_manual(values = palette_7, name = "", guide = F) +
  facet_grid(. ~ fac, drop = T, scales = "free") +
  theme_classic() +
  theme(
    legend.position = "bottom",
    axis.text.y = element_blank(),
    axis.line.y = element_blank(),
    axis.ticks.y = element_blank(),
    strip.background = element_rect(color="white",size=1.5, linetype="solid"),
    strip.text.x = element_text(
      size = 2, color = "white", face = "bold.italic"
    )
  )

```

```
)  
)
```

```
ggarrange(plt_a, plt_b, plt_c, plt_d, plt_e, plt_f, ncol = 2, nrow = 3, widths = c(1.5, 8), align = "hv", common.legend = T, legend = "bottom")
```



```

plt_a = ggplot() +
  geom_point(aes(x = "Average\ncommunity", y = 50), colour = "grey30", size = 2.5) +
  geom_hline(aes(yintercept = 50), linetype = "dashed") +
  labs(x = "", y = "Species richness", title = "Intact") +
  scale_y_continuous(expand = c(0,0), limits = c(40,51)) +
  facet_grid(. ~ "", scales = "free") +
  theme_classic() +
  theme(strip.background = element_rect(color="white", fill="white", size=1.5, linetype="solid"),
        plot.title = element_text(size=10, face = "bold"))

plt_b = ggplot(data = scen_df) +
  geom_pointrange(aes(x = scen, y = rich_mn, ymin = rich_low, ymax = rich_upr, colour = fac), size = 0.5, alpha = 0.3) +
  geom_pointrange(aes(x = scen, y = rich2_mn, ymin = rich2_low, ymax = rich2_upr, colour = fac), size = 0.5, position = position_nudge(x = 0.2), shape = "triangle-up") +
  geom_hline(aes(yintercept = 50), linetype = "dashed") +
  labs(x = "", y = "", title = "Post-reintroduction") +
  scale_y_continuous(expand = c(0,0), limits = c(40,51)) +
  scale_colour_manual(values = palette_7, name = "", guide = F) +
  facet_grid(. ~ fac, drop = T, scales = "free") +
  theme_classic() +
  theme(
    legend.position = "bottom",
    axis.text.y = element_blank(),
    axis.line.y = element_blank(),
    axis.ticks.y = element_blank(),
    strip.background = element_rect(color="white",size=1.5, linetype="solid"),
    plot.title = element_text(size=10, face = "bold")
  )

plt_c = ggplot() +
  geom_point(aes(x = "Average\ncommunity", y = 100), colour = "grey30", size = 2.5) +
  geom_hline(aes(yintercept = 100), linetype = "dashed") +
  labs(x = "", y = "Biomass", title = "") +
  scale_y_continuous(expand = c(0,0), limits = c(80,140)) +
  facet_grid(. ~ "", scales = "free") +
  theme_classic() +
  theme(strip.background = element_rect(color="white", fill="white", size=1.5, linetype="solid"))

plt_d = ggplot(data = scen_df) +
  geom_pointrange(aes(x = scen, y = bio_mn, ymin = bio_low, ymax = bio_upr, colour = fac), size = 0.5, alpha = 0.3) +
  geom_pointrange(aes(x = scen, y = bio2_mn, ymin = bio2_low, ymax = bio2_upr, colour = fac), size = 0.5, position = position_nudge(x = 0.2), shape = "triangle-up") +
  geom_hline(aes(yintercept = 100), linetype = "dashed") +

```

```

labs(x = "", y = "", title = "") +
scale_y_continuous(expand = c(0,0), limits = c(80,140)) +
scale_colour_manual(values = palette_7, name = "", guide = F) +
facet_grid(. ~ fac, drop = T, scales = "free") +
theme_classic() +
theme(
  legend.position = "bottom",
  axis.text.y = element_blank(),
  axis.line.y = element_blank(),
  axis.ticks.y = element_blank(),
  strip.background = element_rect(color="white",size=1.5, linetype="solid"),
  strip.text.x = element_text(
    size = 2, color = "white", face = "bold.italic"
  )
)

plt_e = ggplot() +
  geom_point(aes(x = "Average\ncommunity", y = 0.1), colour = "grey30", size = 2.5) +
  geom_hline(aes(yintercept = 0.1), linetype = "dashed") +
  labs(x = "", y = "Stability", title = "") +
  scale_y_log10(expand = c(0,0), limits = c(0.00001,1), breaks = c(1e-4, 1e-3, 1e-2, 1e-1,1), labels=function(x) parse(text=paste("10^",round(log10(x),1)))) +
  facet_grid(. ~ "", scales = "free") +
  theme_classic() +
  theme(strip.background = element_rect(color="white", fill="white", size=1.5, linetype="solid"))

plt_f = ggplot(data = scen_df) +
  geom_pointrange(aes(x = scen, y = stab_mn, ymin = stab_low, ymax = stab_upr, colour = fac), size = 0.5, alpha = 0.3) +
  geom_pointrange(aes(x = scen, y = stab2_mn, ymin = stab2_low, ymax = stab2_upr, colour = fac), size = 0.5, position = position_nudge(x = 0.2), s
  geom_hline(aes(yintercept = 0.1), linetype = "dashed") +
  labs(x = "", y = "", title = "") +
  scale_y_log10(expand = c(0,0), limits = c(0.00001, 1)) +
  scale_colour_manual(values = palette_7, name = "", guide = F) +
  facet_grid(. ~ fac, drop = T, scales = "free") +
  theme_classic() +
  theme(
    legend.position = "bottom",
    axis.text.y = element_blank(),
    axis.line.y = element_blank(),
    axis.ticks.y = element_blank(),
    strip.background = element_rect(color="white",size=1.5, linetype="solid"),
    strip.text.x = element_text(

```

```
    size = 2, color = "white", face = "bold.italic"  
  )  
)  
  
ggarrange(plt_a, plt_b, plt_c, plt_d, plt_e, plt_f, ncol = 2, nrow = 3, widths = c(1.5, 8), align = "hv", common.legend = T, legend = "bottom")
```

