CO_sgs_code_forMS

Load packages

```
require(effects)
require(lattice)
require(latticeExtra)
require(gridExtra)
require(grid)
require(ggpubr)
require(lme4)
require(ggeffects)
require(tidyverse)
```

Load model output data

```
#### set wd ####
setwd("/Users/Alice/Dropbox/Grad School/Research/Trait Project/CO_sgs Analysis")
#get model result data into the environment
load("./SCRIPTS/models_11232020.RData") #change the file name to the most current version of model runs
```

Make a figure of model results for LDMC, RDMC, and TLP for forb and gram survival

```
#make figure for graminoid survival

#get 2.5 and 97.5 percentiles of the distribution
meanSPEI_G <- mean(CO_grams$SPEI_s)
sdSPEI_G <- sd(CO_grams$SPEI_s)
#get 97.5 quantile of the distribution
SPEI_97_5_G <- qnorm(.975, meanSPEI_G, sdSPEI_G)
SPEI_2_5_G <- qnorm(.025, meanSPEI_G, sdSPEI_G)

spei_vals <- c(SPEI_2_5_G, SPEI_97_5_G)

#for TLP_s
TLP_vals <- seq(min(CO_grams$TLP_s, na.rm = TRUE), max(CO_grams$TLP_s, na.rm = TRUE), length.out = 20)
TLP_G_dat <- ggpredict(m1_grams, terms = c("TLP_s[TLP_vals]", "SPEI_s[spei_vals]"), type = "fixed", back


#for LDMC_s
LDMC_vals <- seq(min(CO_grams$LDMC_s, na.rm = TRUE), max(CO_grams$LDMC_s, na.rm = TRUE), length.out = 2
LDMC_G_dat <- ggpredict(m2_grams, terms = c("LDMC_s[LDMC_vals]", "SPEI_s[spei_vals]"), type = "fixed",

#for RDMC_s
RDMC_vals <- seq(min(CO_grams$RDMC_s, na.rm = TRUE), max(CO_grams$RDMC_s, na.rm = TRUE), length.out = 2
RDMC_G_dat <- ggpredict(m9, terms = c("RDMC_s[RDMC_vals]", "SPEI_s[spei_vals]"), type = "fixed", back.t


#make a data.frame to contain all of the values for each trait
GramDat <- data.frame(trait = c("scaled(Turgor Loss Point) (MPa)"), x = TLP_G_dat$x, GramSurv = TLP_G_d
```

```r
GramDat <- rbind(GramDat, data.frame(trait = c("scaled(Leaf Dry Matter Content) (g/g)"), x = LDMC_G_dat$

GramDat <- rbind(GramDat, data.frame(trait = c("scaled(Root Dry Matter Content) (g/g)"), x = RDMC_G_dat$

#make data for rug plot
RugDat_G <-  data.frame(rug = CO_grams$TLP_s, trait = "scaled(Turgor Loss Point) (MPa)")
RugDat_G <- rbind(RugDat_G, data.frame(rug = CO_grams$LDMC_s, trait = "scaled(Leaf Dry Matter Content)
RugDat_G <- rbind(RugDat_G, data.frame(rug = CO_grams$RDMC_s, trait = "scaled(Root Dry Matter Content)

#text for labels
dat_text <- data.frame(
  label = c("A", "B", "C"),
  trait = c("scaled(Turgor Loss Point) (MPa)", "scaled(Leaf Dry Matter Content) (g/g)", "scaled(Root Dry
  x    = c(min(TLP_G_dat$x),min(LDMC_G_dat$x),min(RDMC_G_dat$x)),
  y     = c(1,1,1)
)

#make a multipanel figure that shows only the graminoid survival probs for 3 traits
gramSurvFigure <- ggplot(data = GramDat) +
  geom_ribbon(aes(x = x, ymin = CI_low, ymax = CI_high, fill = SPEI), alpha = 0.3) +
  geom_line(aes(x=x, GramSurv, col = SPEI))  +
  geom_rug(aes(x = rug), data = RugDat_G) +
  ggtitle(NULL) +
  xlab(NULL) +
  ylab("P(Graminoid Survival)") +
  scale_y_continuous(limits = c(0,1)) +
  scale_color_manual(labels = c("dry year", "wet year"), values = c("goldenrod1", "royalblue2"), guide =
  scale_fill_manual(values = c("goldenrod1", "royalblue2"), guide = FALSE) +
  facet_wrap(vars(trait), scales = "free_x", strip.position =  "bottom") +
  theme_classic()+
  theme(legend.position = "bottom", legend.title = element_blank(), legend.background = element_rect(fil
  geom_text(data= dat_text, mapping = aes(x = x, y = y, label = label), size = 3, fontface = "bold")
```

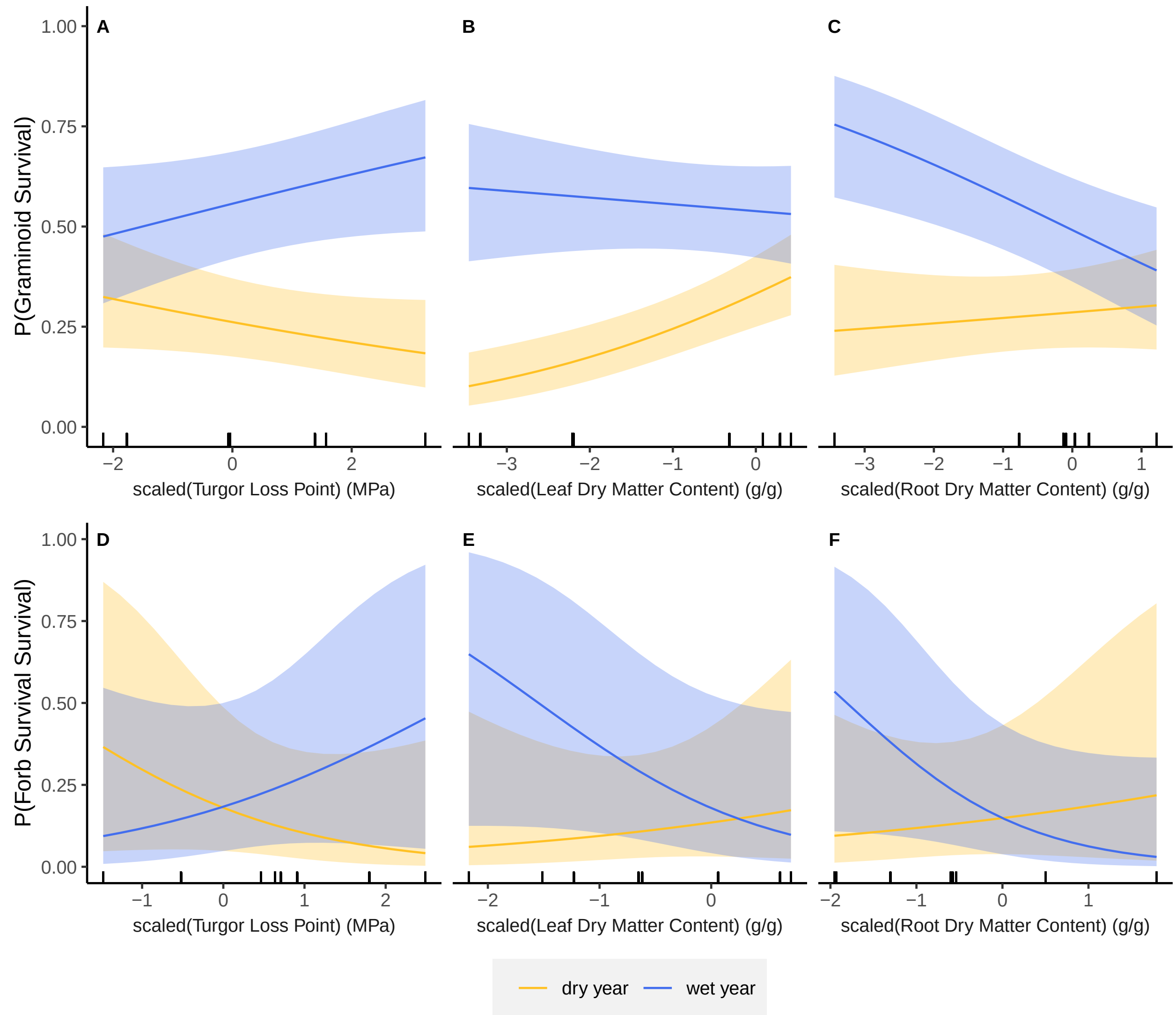
make figure for forb survival

```r
#get 2.5 and 97.5 percentiles of the distribution
meanSPEI_F <- mean(CO_point_all$SPEI_s)
sdSPEI_F <- sd(CO_point_all$SPEI_s)
#get 97.5 quantile of the distribution
SPEI_97_5_F <- qnorm(.975, meanSPEI_F, sdSPEI_F)
SPEI_2_5_G <- qnorm(.025, meanSPEI_F, sdSPEI_F)

spei_vals <- c(SPEI_2_5_G, SPEI_97_5_G)

#for TLP_s
TLP_vals <- seq(min(CO_point_all$TLP_s, na.rm = TRUE), max(CO_point_all$TLP_s, na.rm = TRUE), length.out
TLP_F_dat <- ggpredict(m3, terms = c("TLP_s[TLP_vals]", "SPEI_s[spei_vals]"), type = "fixed", back.trans


#for LDMC_s
LDMC_vals <- seq(min(CO_point_all$LDMC_s, na.rm = TRUE), max(CO_point_all$LDMC_s, na.rm = TRUE), length
LDMC_F_dat <- ggpredict(m4, terms = c("LDMC_s[LDMC_vals]", "SPEI_s[spei_vals]"), type = "fixed", back.t

#for RDMC_s
```

```r
RDMC_vals <- seq(min(CO_point_all$RDMC_s, na.rm = TRUE), max(CO_point_all$RDMC_s, na.rm = TRUE), length
RDMC_F_dat <- ggpredict(m11, terms = c("RDMC_s[RDMC_vals]", "SPEI_s[spei_vals]"), type = "fixed", back.


#make a data.frame to contain all of the values for each trait
ForbDat <- data.frame(trait = c("scaled(Turgor Loss Point) (MPa)"), x = TLP_F_dat$x, ForbSurv = TLP_F_da

ForbDat <- rbind(ForbDat, data.frame(trait = c("scaled(Leaf Dry Matter Content) (g/g)"), x = LDMC_F_dat

ForbDat <- rbind(ForbDat, data.frame(trait = c("scaled(Root Dry Matter Content) (g/g)"), x = RDMC_F_dat

#make data for rug plot
RugDat_F <-  data.frame(rug = CO_point_all$TLP_s, trait = "scaled(Turgor Loss Point) (MPa)")
RugDat_F <- rbind(RugDat_F, data.frame(rug = CO_point_all$LDMC_s, trait = "scaled(Leaf Dry Matter Conte
RugDat_F <- rbind(RugDat_F, data.frame(rug = CO_point_all$RDMC_s, trait = "scaled(Root Dry Matter Conte

#text for labels
dat_text <- data.frame(
  label = c("D", "E", "F"),
  trait = c("scaled(Turgor Loss Point) (MPa)", "scaled(Leaf Dry Matter Content) (g/g)", "scaled(Root Dry
  x    = c(min(TLP_F_dat$x),min(LDMC_F_dat$x),min(RDMC_F_dat$x)),
  y     = c(1,1,1)
)


#make a multipanel figure that shows only the graminoid survival probs for 3 traits
forbSurvFigure <- ggplot(data = ForbDat) +
  geom_ribbon(aes(x = x, ymin = CI_low, ymax = CI_high, fill = SPEI), alpha = 0.3) +
  geom_line(aes(x=x, ForbSurv, col = SPEI))  +
  geom_rug(aes(x = rug), data = RugDat_F) +
  labs(title = NULL) +
  xlab(NULL) +
  ylab("P(Forb Survival Survival)") +
  scale_y_continuous(limits = c(0,1)) +
  scale_color_manual(labels = c("dry year", "wet year"), values = c("goldenrod1", "royalblue2")) +
  geom_text(data= dat_text, mapping = aes(x = x, y = y, label = label), size = 3, fontface = "bold") +
  scale_fill_manual(values = c("goldenrod1", "royalblue2"), guide = FALSE) +
  facet_wrap(~trait, scales = "free_x", strip.position =  "bottom") +
  theme_classic()+
  theme(legend.position = "bottom", legend.title = element_blank(), legend.background = element_rect(fi
```

Combine into one figure

```r
(mainObs <- ggarrange(gramSurvFigure, forbSurvFigure, ncol = 1, nrow = 2))
```



Make climate variability figure for use in the conceptual diagram (fig. 1)

```
source("/Users/Alice/Dropbox/Grad School/Research/Trait Project/Data/Climate Data/CrossSiteClimateCompa


#figure of annual precip variability at CO site
CO_precip <- ggplot(data = CO[!is.na(CO$Ann.Sum.Precip),])+
  geom_line(aes(x = Year, y = Ann.Sum.Precip), col = "gray25") +
  ylab(expression("MAP (mm)" %->% "")) +
  scale_x_continuous(labels = NULL, breaks = NULL) +
  scale_y_continuous(labels = NULL, breaks = NULL) +
  theme_classic()

setwd("/Users/Alice/Dropbox/Grad School/Research/Trait Project/CO_sgs Analysis")
pdf("./Manuscript/Figures/CO_MAP.pdf", width = 3, height = 3)
CO_precip
dev.off()
```

```
## pdf
##   2
```

Make a plot of random effects of individual plant size on survival for LDMC model (best model for graminoid survival)

```
#get random effect data
#refit model w/ factors instead of logical values
m2_fac <- glmer(as.factor(survives_tplus1) ~ SPEI_s * LDMC_s +  area_s + neighbors_10_s + as.factor(nea

sppAreaPreds_s <- ggpredict(m2_fac, terms = c("area_s[all]", "species"), type = "random")
sppAreaPreds_s <- data.frame("x" = sppAreaPreds_s$x, "preds" = sppAreaPreds_s$predicted,"spp" = sppArea

globPreds_a_s <- ggpredict(m2_fac, terms = c("area_s[all]"), type = "random")
globPreds_a_s <- data.frame("x" = globPreds_a_s$x, "preds" = globPreds_a_s$predicted, "spp" = as.factor

AreaEffectSurv <- ggplot() +
  geom_line(data = sppAreaPreds_s, aes(x = x, y = preds, col = spp), alpha = .8)+
  geom_line(data = globPreds_a_s, aes(x = x, y = preds), lwd = 1.25) +
  #geom_line(aes(x = globPreds_a_s$x, y = globPreds_a_s$CI_low)) +
  geom_polygon(aes(x = c(globPreds_a_s$x,rev(globPreds_a_s$x)), y = c( globPreds_a_s$CI_low, rev(globPre
  theme_classic() +
  xlab(c(expression(size[year_t] ))) +
  ylab("P(Graminoid Survival)") +
  scale_color_manual(values = c("grey60", "grey60", "grey60", "grey60", "grey60", "grey60", "grey60", "g
  theme(axis.ticks.x.bottom = element_blank(),
        axis.text.x.bottom = element_blank(),
        legend.position = "none")
```

Make plot of fixed effect of neighborhood density for effect of LDMC*SPEI on graminoid survival

```
sppNeighPreds_s <- ggpredict(m2_fac, terms = c("neighbors_10_s[all]", "species"), type = "random")
sppNeighPreds_s <- data.frame("x" = sppNeighPreds_s$x, "preds" = sppNeighPreds_s$predicted,"spp" = sppNe

globPreds_n_s <- ggpredict(m2_fac, terms = c("neighbors_10_s[all]"), type = "fixed")
globPreds_n_s <- data.frame("x" = globPreds_n_s$x, "preds" = globPreds_n_s$predicted, "spp" = as.factor

NeighEffectSurv <- ggplot() +
  geom_line(data = sppNeighPreds_s, aes(x = x, y = preds, col = spp), alpha = .8)+
  geom_line(data = globPreds_n_s, aes(x = x, y = preds), lwd = 1.25) +
  geom_polygon(aes(x = c(globPreds_n_s$x,rev(globPreds_n_s$x)), y = c( globPreds_n_s$CI_low, rev(globPre
  theme_classic() +
  ylim(c(0,1))+
  xlab("Conspecific Local Neighborhood Competition") +
  ylab("P(Graminoid Survival)") +
  scale_color_manual(values = c("grey60", "grey60", "grey60", "grey60", "grey60", "grey60", "grey60", "g
  theme(axis.ticks.x.bottom = element_blank(),
        axis.text.x.bottom = element_blank(),
        legend.position = "none")
```

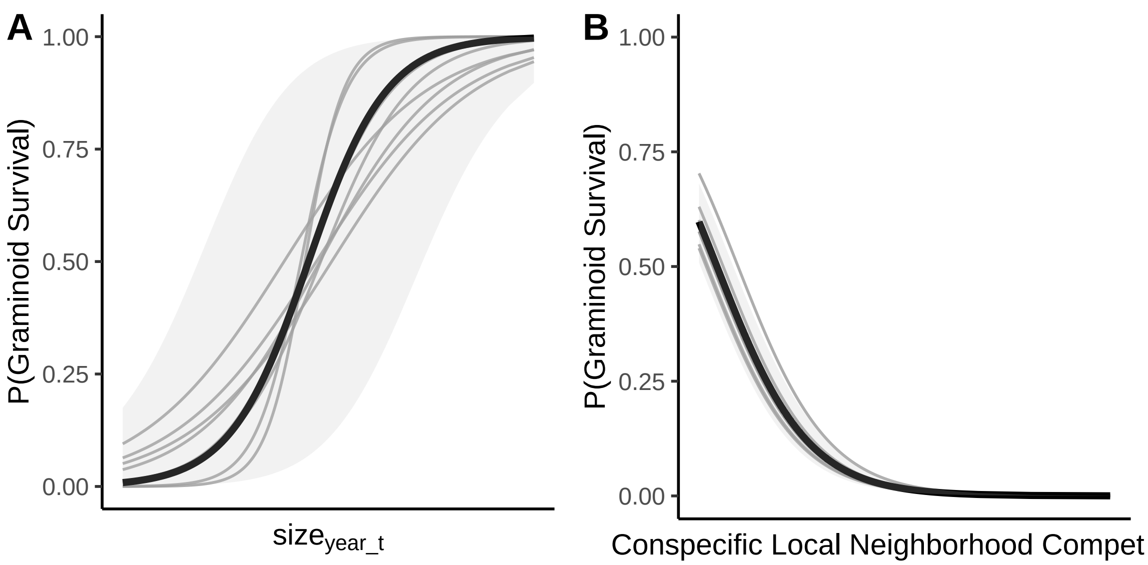
Make plot of fixed effect of neighborhood by species on growth for TLP model

```
mGrowTLP_fac<- lme4::lmer(logDiffArea ~  neighbors_10_s + TLP_s + SPEI_s * TLP_s + as.factor(nearEdge_t

sppNeighPreds_g <- ggpredict(mGrowTLP_fac, terms = c("neighbors_10_s[all]", "species"), type = "random"
sppNeighPreds_g <- data.frame("x" = sppNeighPreds_g$x, "preds" = sppNeighPreds_g$predicted,"spp" = sppNe

globPreds_n_g <- ggpredict(mGrowTLP_fac, terms = c("neighbors_10_s[all]"), type = "fixed")
globPreds_n_g <- data.frame("x" = globPreds_n_g$x, "preds" = globPreds_n_g$predicted, "spp" = as.factor
```

```
NeighEffectGrowth <- ggplot() +
  geom_line(data = sppNeighPreds_g, aes(x = x, y = preds, col = spp), alpha = .8)+
  geom_line(data = globPreds_n_g, aes(x = x, y = preds), lwd = 1.25) +
  geom_polygon(aes(x = c(globPreds_n_g$x,rev(globPreds_n_g$x)), y = c( globPreds_n_g$CI_low, rev(globPre
  theme_classic() +
  xlab(c(expression("Conspecific Local Neighborhood Competition"))) +
  ylab(expression("log" ~ bgroup("(",frac(size[year_t+1],size[year_t]),")"))) +
  scale_color_manual(values = c("grey60", "grey60", "grey60", "grey60", "grey60", "grey60", "grey60", "g
  theme(axis.ticks = element_blank(),
        axis.text = element_blank(),
        legend.position = "none")
```

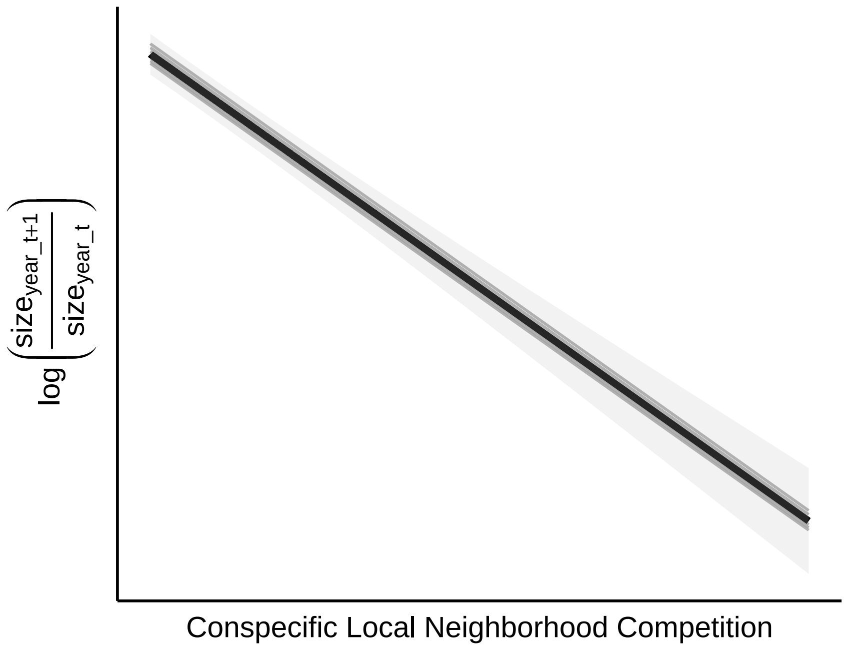
Make figure 3 (combination of figures for effect of size and local neighborhood on LDMC*SPEI effect of graminoid survival)

```
(effectsSurv <- ggarrange(AreaEffectSurv, NeighEffectSurv,
          labels = c("A", "B"),
          ncol = 2, nrow = 1))
```



Make figure 7 in supplement (figure for effect of local neighborhood on TLP*SPEI effect of graminoid growth)

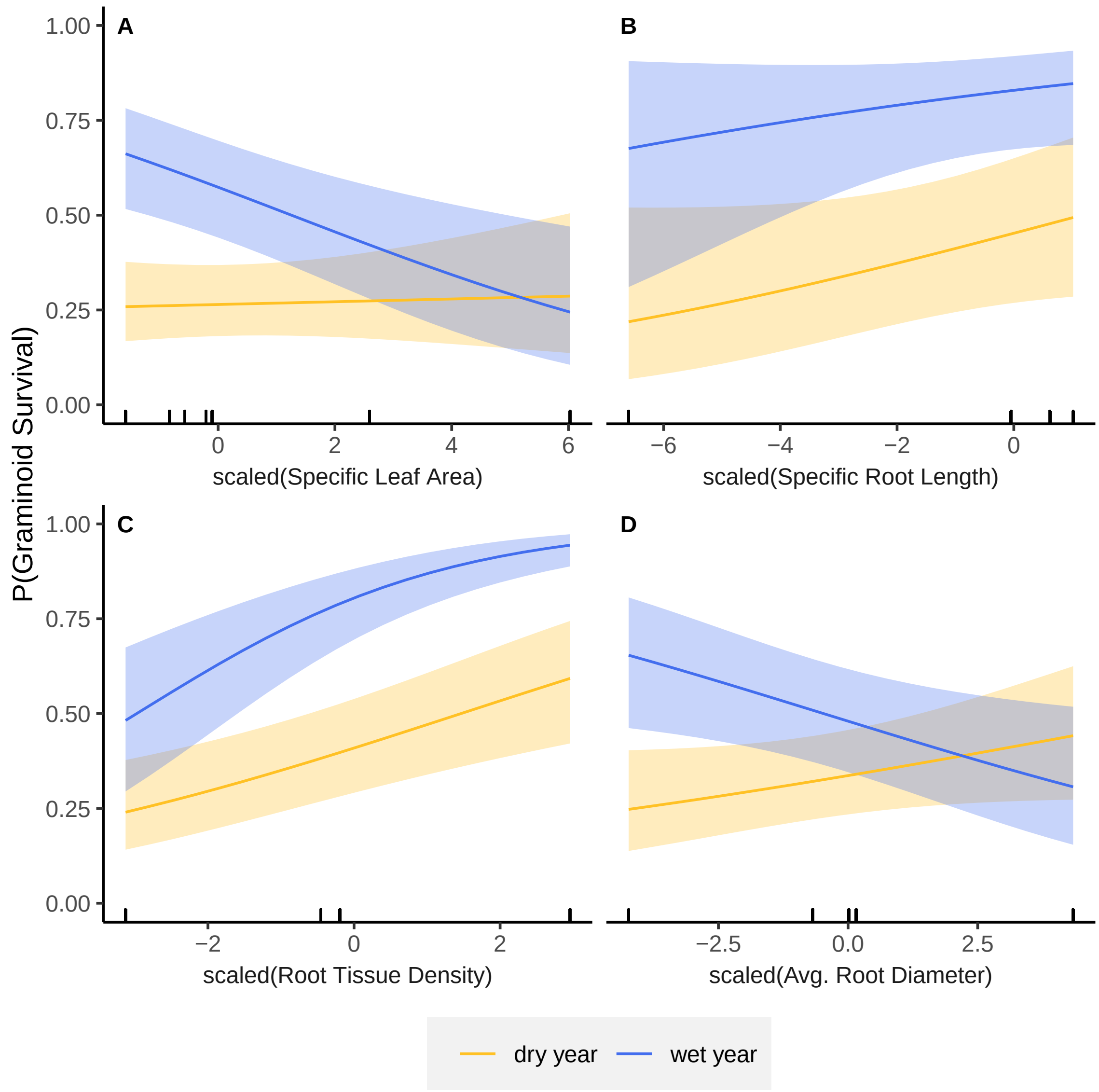
```
(effectsGrowth <- NeighEffectGrowth)
```



plot of all graminoid survival model results (for traits not in fig. 2)

```
##polygons
#TLP
#get 2.5 and 97.5 percentiles of the distribution
meanSPEI_G <- mean(CO_grams$SPEI_s)
sdSPEI_G <- sd(CO_grams$SPEI_s)
#get 97.5 quantile of the distribution
SPEI_97_5_G <- qnorm(.975, meanSPEI_G, sdSPEI_G) #2.10
SPEI_2_5_G <- qnorm(.025, meanSPEI_G, sdSPEI_G) #-1.48

spei_vals <- c(SPEI_2_5_G, SPEI_97_5_G)
SLA_vals <- seq(min(CO_grams$SLA_s), max(CO_grams$SLA_s), length.out = 20)

SLA_G_dat <- ggpredict(m5, terms = c("SLA_s[SLA_vals]", "SPEI_s[spei_vals]"), type = "fixed", back.trans

#for RTD_s
RTD_vals <- seq(min(CO_grams$RTD_s, na.rm = TRUE), max(CO_grams$RTD_s, na.rm = TRUE), length.out = 20)
RTD_G_dat <- ggpredict(m10, terms = c("RTD_s[RTD_vals]", "SPEI_s[spei_vals]"), type = "fixed", back.tran

#for SRL_s
SRL_vals <- seq(min(CO_grams$SRL_s, na.rm = TRUE), max(CO_grams$SRL_s, na.rm = TRUE), length.out = 20)
SRL_G_dat <- ggpredict(m13, terms = c("SRL_s[SRL_vals]", "SPEI_s[spei_vals]"), type = "fixed", back.tran

#for RDiam_s
RDiam_vals <- seq(min(CO_grams$RDiam_s, na.rm = TRUE), max(CO_grams$RDiam_s, na.rm = TRUE), length.out =
RDiam_G_dat <- ggpredict(m14, terms = c("RDiam_s[RDiam_vals]", "SPEI_s[spei_vals]"), type = "fixed", bac

#make a data.frame to contain all of the values for each trait
GramDat <- data.frame(trait = c("scaled(Specific Leaf Area)"), x = SLA_G_dat$x, GramSurv = SLA_G_dat$pre

GramDat <- rbind(GramDat, data.frame(trait = c("scaled(Specific Root Length)"), x = SRL_G_dat$x, GramSur
GramDat <- rbind(GramDat, data.frame(trait = c("scaled(Root Tissue Density)"), x = RTD_G_dat$x, GramSurv
GramDat <- rbind(GramDat, data.frame(trait = c("scaled(Avg. Root Diameter)"), x = RDiam_G_dat$x, GramSur
```

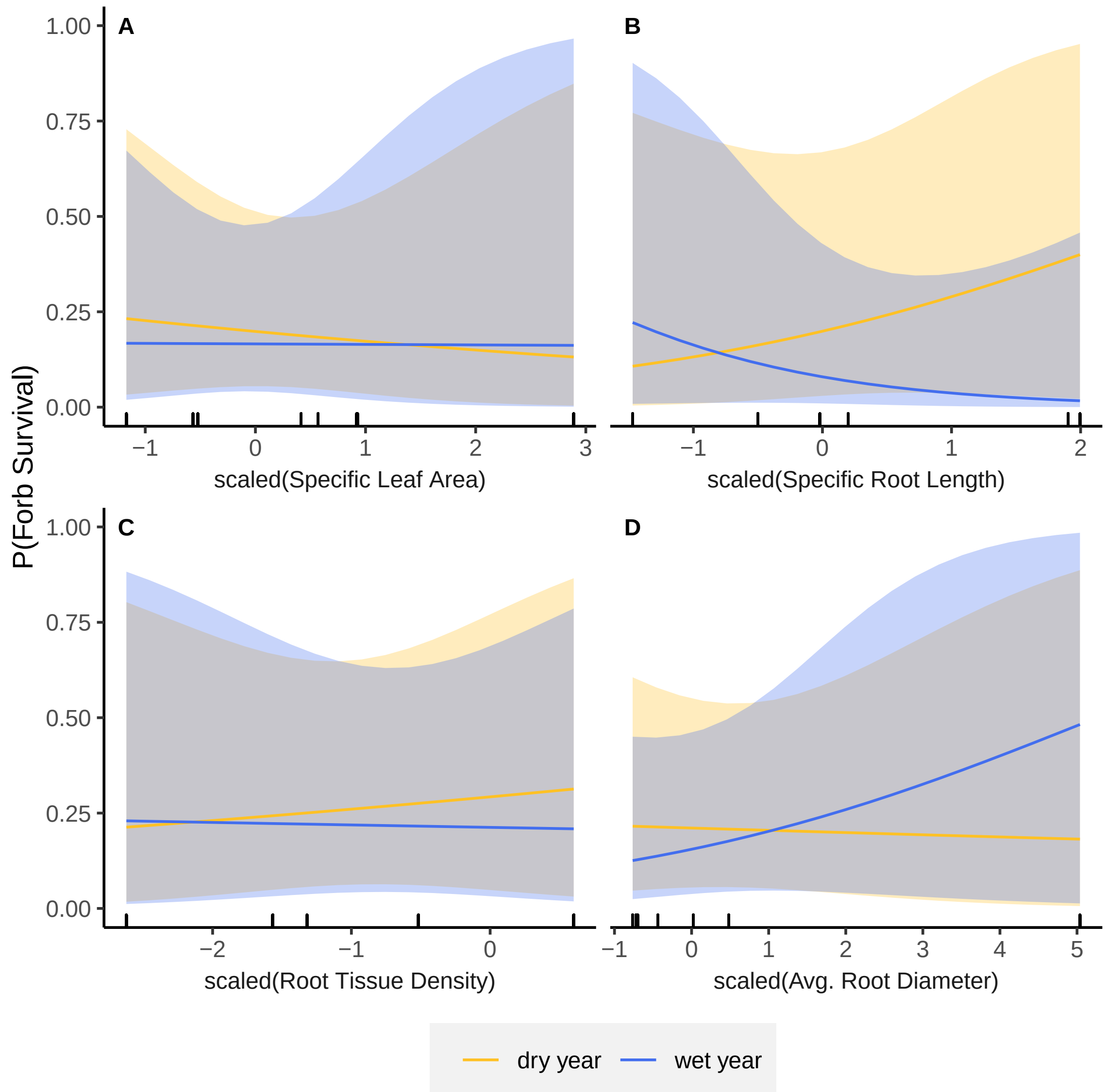
```r
#make data for rug plot
RugDat <-  data.frame(rug = CO_grams$SLA_s, trait = "scaled(Specific Leaf Area)")
RugDat <- rbind(RugDat, data.frame(rug = CO_grams$SRL_s, trait = "scaled(Specific Root Length)"))
RugDat <- rbind(RugDat, data.frame(rug = CO_grams$RTD_s, trait = "scaled(Root Tissue Density)"))
RugDat <- rbind(RugDat, data.frame(rug = CO_grams$RDiam_s, trait = "scaled(Avg. Root Diameter)"))
#text for labels
dat_text <- data.frame(
  label = c("A", "B", "C", "D"),
  trait = c("scaled(Specific Leaf Area)", "scaled(Specific Root Length)", "scaled(Root Tissue Density)"
  x    = c(min(SLA_G_dat$x),min(SRL_G_dat$x),min(RTD_G_dat$x),min(RDiam_G_dat$x)),
  y     = c(1,1,1,1)
)


#make a multipanel figure
(GramSurvExtraFig <- ggplot(data = GramDat) +
  geom_ribbon(aes(x = x, ymin = CI_low, ymax = CI_high, fill = SPEI), alpha = 0.3) +
  geom_line(aes(x, GramSurv, col = SPEI))  +
  geom_rug(aes(x = rug), data = RugDat) +
  labs(title = NULL) +
  xlab(NULL) +
  ylab("P(Graminoid Survival)") +
  scale_y_continuous(limits = c(0,1)) +
  scale_color_manual(labels = c("dry year", "wet year"), values = c("goldenrod1", "royalblue2")) +
  scale_fill_manual(values = c("goldenrod1", "royalblue2"), guide = FALSE) +
  facet_wrap(vars(trait), scales = "free_x", strip.position =  "bottom") +
  theme_classic()+
  theme(legend.position = "bottom", legend.title = element_blank(), legend.background = element_rect(fil
  geom_text(data= dat_text, mapping = aes(x = x, y = y, label = label), size = 3, fontface = "bold"))
```



plot of all forb survival model results (for traits not in fig. 2)

```
#get 2.5 and 97.5 percentiles of the distribution
meanSPEI <- mean(CO_point_all$SPEI_s)
sdSPEI <- sd(CO_point_all$SPEI_s)
#get 97.5 quantile of the distribution
SPEI_97_5 <- qnorm(.975, meanSPEI, sdSPEI)
SPEI_2_5 <- qnorm(.025, meanSPEI, sdSPEI)

spei_vals <- c(SPEI_2_5, SPEI_97_5)

#for SLA_s
SLA_vals <- seq(min(CO_point_all$SLA_s, na.rm = TRUE), max(CO_point_all$SLA_s, na.rm = TRUE), length.ou
SLA_F_dat <- ggpredict(m6, terms = c("SLA_s[SLA_vals]", "SPEI_s[spei_vals]"), type = "fixed", back.trans


#for RTD_s
RTD_vals <- seq(min(CO_point_all$RTD_s, na.rm = TRUE), max(CO_point_all$RTD_s, na.rm = TRUE), length.ou
```

```
RTD_F_dat <- ggpredict(m12, terms = c("RTD_s[RTD_vals]", "SPEI_s[spei_vals]"), type = "fixed", back.tra

#for SRL_s
SRL_vals <- seq(min(CO_point_all$SRL_s, na.rm = TRUE), max(CO_point_all$SRL_s, na.rm = TRUE), length.ou
SRL_F_dat <- ggpredict(m15, terms = c("SRL_s[SRL_vals]", "SPEI_s[spei_vals]"), type = "fixed", back.tra

#for RDiam_s
RDiam_vals <- seq(min(CO_point_all$RDiam_s, na.rm = TRUE), max(CO_point_all$RDiam_s, na.rm = TRUE), leng
RDiam_F_dat <- ggpredict(m16, terms = c("RDiam_s[RDiam_vals]", "SPEI_s[spei_vals]"), type = "fixed", ba

#make a data.frame to contain all of the values for each trait
ForbDat <- data.frame(trait = c("scaled(Specific Leaf Area)"), x = SLA_F_dat$x, GramSurv = SLA_F_dat$pr

ForbDat <- rbind(ForbDat, data.frame(trait = c("scaled(Specific Root Length)"), x = SRL_F_dat$x, GramSu
ForbDat <- rbind(ForbDat, data.frame(trait = c("scaled(Root Tissue Density)"), x = RTD_F_dat$x, GramSur
ForbDat <- rbind(ForbDat, data.frame(trait = c("scaled(Avg. Root Diameter)"), x = RDiam_F_dat$x, GramSu
#make data for rug plot
RugDat <-  data.frame(rug = CO_point_all$SLA_s, trait = "scaled(Specific Leaf Area)")
RugDat <- rbind(RugDat, data.frame(rug = CO_point_all$SRL_s, trait = "scaled(Specific Root Length)"))
RugDat <- rbind(RugDat, data.frame(rug = CO_point_all$RTD_s, trait = "scaled(Root Tissue Density)"))
RugDat <- rbind(RugDat, data.frame(rug = CO_point_all$RDiam_s, trait = "scaled(Avg. Root Diameter)"))
#text for labels
dat_text <- data.frame(
  label = c("A", "B", "C", "D"),
  trait = c("scaled(Specific Leaf Area)", "scaled(Specific Root Length)", "scaled(Root Tissue Density)"
  x    = c(min(SLA_F_dat$x),min(SRL_F_dat$x),min(RTD_F_dat$x),min(RDiam_F_dat$x)),
  y     = c(1,1,1,1)
)

#make a multipanel figure
(ForbSurvExtraFig <- ggplot(data = ForbDat) +
  geom_ribbon(aes(x = x, ymin = CI_low, ymax = CI_high, fill = SPEI), alpha = 0.3) +
  geom_line(aes(x=x, GramSurv, col = SPEI))  +
  geom_rug(aes(x = rug), data = RugDat) +
  labs(title = NULL) +
  xlab(NULL) +
  ylab("P(Forb Survival)") +
  scale_y_continuous(limits = c(0,1)) +
  scale_color_manual(labels = c("dry year", "wet year"), values = c("goldenrod1", "royalblue2")) +
  scale_fill_manual(values = c("goldenrod1", "royalblue2"), guide = FALSE) +
  facet_wrap(vars(trait), scales = "free_x", strip.position =  "bottom") +
  theme_classic()+
  theme(legend.position = "bottom", legend.title = element_blank(), legend.background = element_rect(fi
  geom_text(data= dat_text, mapping = aes(x = x, y = y, label = label), size = 3, fontface = "bold"))
```



Now make a figure showing model results for all growth models

```
#get 2.5 and 97.5 percentiles of the distribution
meanSPEI <- mean(CO_grow_TLP$SPEI_s)
sdSPEI <- sd(CO_grow_TLP$SPEI_s)
#get 97.5 quantile of the distribution
SPEI_97_5 <- qnorm(.975, meanSPEI, sdSPEI)
SPEI_2_5 <- qnorm(.025, meanSPEI, sdSPEI)

spei_vals <- c(SPEI_2_5, SPEI_97_5)

#for TLP_s
TLP_vals <- seq(min(CO_grow_TLP$TLP_s, na.rm = TRUE), max(CO_grow_TLP$TLP_s, na.rm = TRUE), length.out =
TLP_grow_dat <- ggpredict(mGrowTLP, terms = c("TLP_s[TLP_vals]", "SPEI_s[spei_vals]"), type = "fixed")

#for LDMC_s
LDMC_vals <- seq(min(CO_grow_LDMC$LDMC_s, na.rm = TRUE), max(CO_grow_LDMC$LDMC_s, na.rm = TRUE), length
LDMC_grow_dat <- ggpredict(mGrowLDMC, terms = c("LDMC_s[LDMC_vals]", "SPEI_s[spei_vals]"), type = "fixe
```

```
#for RDMC_s
RDMC_vals <- seq(min(CO_grow_RDMC$RDMC_s, na.rm = TRUE), max(CO_grow_RDMC$RDMC_s, na.rm = TRUE), length
RDMC_grow_dat <- ggpredict(mGrowRDMC, terms = c("RDMC_s[RDMC_vals]", "SPEI_s[spei_vals]"), type = "fixed

#for SLA_s
SLA_vals <- seq(min(CO_grow_SLA$SLA_s, na.rm = TRUE), max(CO_grow_SLA$SLA_s, na.rm = TRUE), length.out =
SLA_grow_dat <- ggpredict(mGrowSLA, terms = c("SLA_s[SLA_vals]", "SPEI_s[spei_vals]"), type = "fixed")

#for RTD_s
RTD_vals <- seq(min(CO_grow_RTD$RTD_s, na.rm = TRUE), max(CO_grow_RTD$RTD_s, na.rm = TRUE), length.out =
RTD_grow_dat <- ggpredict(mGrowRTD, terms = c("RTD_s[RTD_vals]", "SPEI_s[spei_vals]"), type = "fixed")

#for SRL_s
SRL_vals <- seq(min(CO_grow_SRL$SRL_s, na.rm = TRUE), max(CO_grow_SRL$SRL_s, na.rm = TRUE), length.out =
SRL_grow_dat <- ggpredict(mGrowSRL, terms = c("SRL_s[SRL_vals]", "SPEI_s[spei_vals]"), type = "fixed")

#for RDiam_s
RDiam_vals <- seq(min(CO_grow_RDiam$RDiam_s, na.rm = TRUE), max(CO_grow_RDiam$RDiam_s, na.rm = TRUE), le
RDiam_grow_dat <- ggpredict(mGrowRDiam, terms = c("RDiam_s[RDiam_vals]", "SPEI_s[spei_vals]"), type = "

#make a data.frame to contain all of the values for each trait
GrowthDat <- data.frame(trait = c("scaled(Turgor Loss Point)"), x = TLP_grow_dat$x, Growth = TLP_grow_da
GrowthDat <- rbind(GrowthDat, data.frame(trait = c("scaled(Leaf Dry Matter Content)"), x = LDMC_grow_dat
GrowthDat <- rbind(GrowthDat, data.frame(trait = c("scaled(Root Dry Matter Content)"), x = RDMC_grow_dat
GrowthDat <- rbind(GrowthDat, data.frame(trait = c("scaled(Specific Leaf Area)"), x = SLA_grow_dat$x, Gr
GrowthDat <- rbind(GrowthDat, data.frame(trait = c("scaled(Specific Root Length)"), x = SRL_grow_dat$x,
GrowthDat <- rbind(GrowthDat, data.frame(trait = c("scaled(Root Tissue Density)"), x = RTD_grow_dat$x, G
GrowthDat <- rbind(GrowthDat, data.frame(trait = c("scaled(Avg. Root Diameter)"), x = RDiam_grow_dat$x,

#make a data.frame with data for the rug plot
RugDat <- data.frame(rug = CO_grow_TLP$TLP_s, trait = "scaled(Turgor Loss Point)")
RugDat <- rbind(RugDat, data.frame(rug = CO_grow_LDMC$LDMC_s, trait = "scaled(Leaf Dry Matter Content)"
RugDat <- rbind(RugDat, data.frame(rug = CO_grow_RDMC$RDMC_s, trait = "scaled(Root Dry Matter Content)"
RugDat <- rbind(RugDat, data.frame(rug = CO_grow_SLA$SLA_s, trait = "scaled(Specific Leaf Area)"))
RugDat <- rbind(RugDat, data.frame(rug = CO_grow_SRL$SRL_s, trait = "scaled(Specific Root Length)"))
RugDat <- rbind(RugDat, data.frame(rug = CO_grow_RTD$RTD_s, trait = "scaled(Root Tissue Density)"))
RugDat <- rbind(RugDat, data.frame(rug = CO_grow_RDiam$RDiam_s, trait = "scaled(Avg. Root Diameter)"))

#text for labels
dat_text <- data.frame(
  label = c("A", "B", "C", "D","E","F","G"),
  trait = c("scaled(Turgor Loss Point)", "scaled(Leaf Dry Matter Content)", "scaled(Root Dry Matter Con
  x    = c(min(TLP_grow_dat$x), min(LDMC_grow_dat$x), min(RDMC_grow_dat$x), min(SLA_grow_dat$x),min(SRL_
  y     = c(1.2,1.2,1.2,1.2,1.2,1.2,1.2)
)

#make a multipanel figure
(GrowthExtraFig <- ggplot(data = GrowthDat) +
  geom_ribbon(aes(x = x, ymin = CI_low, ymax = CI_high, fill = SPEI), alpha = 0.3) +
  geom_line(aes(x=x, Growth, col = SPEI))  +
  geom_rug(aes(x = rug), data = RugDat) +
  labs(title = NULL) +
  xlab(NULL) +
```

```
  ylab(expression("Graminoid Growth: log" ~ bgroup("(",frac(size[year_t+1],size[year_t]),")")))  +
  scale_color_manual(labels = c("dry year", "wet year"), values = c("goldenrod1", "royalblue2")) +
  scale_fill_manual(values = c("goldenrod1", "royalblue2"), guide = FALSE) +
  facet_wrap(vars(trait), scales = "free_x", strip.position =  "bottom") +
  theme_classic()+
  theme(legend.position = "bottom", legend.title = element_blank(), legend.background = element_rect(fil
  geom_text(data= dat_text, mapping = aes(x = x, y = y, label = label), size = 3, fontface = "bold"))
```