Vignette

Phenological characterization and modeling of diverse lentil (*Lens culinaris* Medik.) germplasm grown in multiple environments

Derek M. Wright

2019-12-03

Contents

This vignette contains the `R` code and analysis done for the paper: **Phenological characterization and modeling of diverse lentil (*Lens culinaris* Medik.) germplasm grown in multiple environments**

Preliminary Steps

Load the nessesary R packages, Prepare the data for analysis.

```
# Load libraries
library(tidyverse)  # data wrangling
library(scales)     # rescale()
library(rworldmap)  # mapBubbles()
library(ggrepel)    # geom_text_repel() + geom_label_repel()
library(magick)     # image editing
library(GGally)     # ggpairs() + ggmatrix()
library(ggpubr)     # ggarrange()
library(ggbeeswarm) # geom_quasirandom()
library(agricolae)  # AMMI()
library(FactoMineR) # PCA() & HCPC()
library(plot3D)     # 3D plots
library(stringr)    # str_pad()
# General color palettes
colors <- c("darkred",   "darkorange3", "darkgoldenrod2", "deeppink3",
            "steelblue", "darkorchid4", "cornsilk4",      "darkgreen")
# Expts color palette
colors_Expt <- c("lightgreen",      "palegreen4",      "darkgreen",   "darkolivegreen3",
                 "darkolivegreen4", "springgreen4",     "orangered2",  "orangered4",
                 "palevioletred",    "mediumvioletred", "orange2",     "orange4",
                 "slateblue1",       "slateblue4",      "aquamarine3", "aquamarine4",
                 "deepskyblue3",     "deepskyblue4" )
# Locations
names_Location <- c("Rosthern, Canada", "Sutherland, Canada",  "Central Ferry, USA",
                    "Bhopal, India",    "Jessore, Bangladesh", "Bardiya, Nepal",
                    "Cordoba, Spain",   "Marchouch, Morocco",  "Metaponto, Italy" )
# Experiments
names_Expt <- c("Rosthern, Canada 2016",    "Rosthern, Canada 2017",
                "Sutherland, Canada 2016",  "Sutherland, Canada 2017",
                "Sutherland, Canada 2018",  "Central Ferry, USA 2018",
                "Bhopal, India 2016",       "Bhopal, India 2017",
                "Jessore, Bangladesh 2016", "Jessore, Bangladesh 2017",
```

```r
                 "Bardiya, Nepal 2016",      "Bardiya, Nepal 2017",
                 "Cordoba, Spain 2016",      "Cordoba, Spain 2017",
                 "Marchouch, Morocco 2016",  "Marchouch, Morocco 2017",
                 "Metaponto, Italy 2016",    "Metaponto, Italy 2017" )
# Experiment short names
names_ExptShort <- c("Ro16", "Ro17", "Su16", "Su17", "Su18", "Us18",
                      "In16", "In17", "Ba16", "Ba17", "Ne16", "Ne17",
                      "Sp16", "Sp17", "Mo16", "Mo17", "It16", "It17" )
# Macro-Environments
macroEnvs <- c("Temperate", "South Asia", "Mediterranean")
# ggplot theme
theme_AGL <- theme_bw() + theme(strip.background = element_rect(fill = "White"))
# Create scaling function
traitScale <- function(x, trait) {
  xout <- rep(NA, nrow(x))
  for(i in unique(x$Expt)) {
    mn <- x %>% filter(Expt == i) %>% pull(trait) %>% min(na.rm = T)
    mx <- x %>% filter(Expt == i) %>% pull(trait) %>% max(na.rm = T)
    xout <- ifelse(x$Expt == i, rescale(x %>% pull(trait), c(1,5), c(mn,mx)), xout)
  }
  xout
}
# Prep data
# Note: DTF2 = non-flowering genotypes <- group_by(Expt) %>% max(DTF)
rr <- read.csv("data/data_Raw.csv") %>%
  mutate(Rep          = factor(Rep),
         Year         = factor(Year),
         PlantingDate = as.Date(PlantingDate),
         Location     = factor(Location, levels = names_Location),
         Expt         = factor(Expt,     levels = names_Expt),
         ExptShort    = plyr::mapvalues(Expt, names_Expt, names_ExptShort),
         ExptShort    = factor(ExptShort, levels = names_ExptShort),
         DTF2_scaled = traitScale(., "DTF2"),
         RDTF         = round(1 / DTF2, 6),
         VEG          = DTF - DTE,
         REP          = DTM - DTF)
# Average raw data
dd <- rr %>% group_by(Entry, Name, Expt, ExptShort, Location, Year) %>%
  summarise_at(vars(DTE, DTF, DTS, DTM, VEG, REP, RDTF, DTF2),
               funs(mean), na.rm = T) %>% ungroup() %>%
  mutate(DTF2_scaled = traitScale(., "DTF2"))
# Prep environmental data
ee <- read.csv("data/data_Env.csv") %>%
  mutate(Date      = as.Date(Date),
         ExptShort = plyr::mapvalues(Expt, names_Expt, names_ExptShort),
         ExptShort = factor(ExptShort, levels = names_ExptShort),
         Expt      = factor(Expt,      levels = names_Expt),
         Location  = factor(Location,  levels = names_Location),
         DayLength_rescaled = rescale(DayLength, to = c(0, 40)) )
# Prep field trial info
xx <- dd %>% group_by(Expt) %>%
  summarise_at(vars(DTE, DTF, DTS, DTM), funs(min, mean, max), na.rm = T) %>%
  ungroup()
```

```r
ff <- read.csv("data/data_Info.csv") %>%
  mutate(Start = as.Date(Start) ) %>%
  left_join(xx, by = "Expt")
for(i in unique(ee$Expt)) {
  ee <- ee %>%
    filter(Expt != i | (Expt == i & DaysAfterPlanting <= ff$DTM_max[ff$Expt == i]))
}
xx <- ee
for(i in unique(ee$Expt)) {
  xx <- xx %>%
    filter(Expt != i | (Expt == i & DaysAfterPlanting <= ff$DTF_max[ff$Expt == i]))
}
xx <- xx %>% group_by(Location, Year) %>%
  summarise(T_mean = mean(Temp_mean, na.rm = T), T_sd = sd(Temp_mean, na.rm = T),
            P_mean = mean(DayLength, na.rm = T), P_sd = sd(DayLength, na.rm = T) ) %>%
  ungroup() %>%
  mutate(Expt = paste(Location, Year)) %>%
  select(-Location, -Year)
ff <- ff %>% left_join(xx, by = "Expt") %>%
  mutate(ExptShort = plyr::mapvalues(Expt, names_Expt, names_ExptShort),
         ExptShort = factor(ExptShort, levels = names_ExptShort),
         Expt      = factor(Expt,      levels = names_Expt),
         MacroEnv  = factor(MacroEnv,  levels = macroEnvs),
         T_mean    = round(T_mean, 1),
         P_mean    = round(P_mean, 1))
# Lentil Diversity Panel metadata
ldp <- read.csv("data/data_LDP.csv")
# Country info
ct <- read.csv("data/data_Countries.csv") %>% filter(Country %in% ldp$Origin)
```

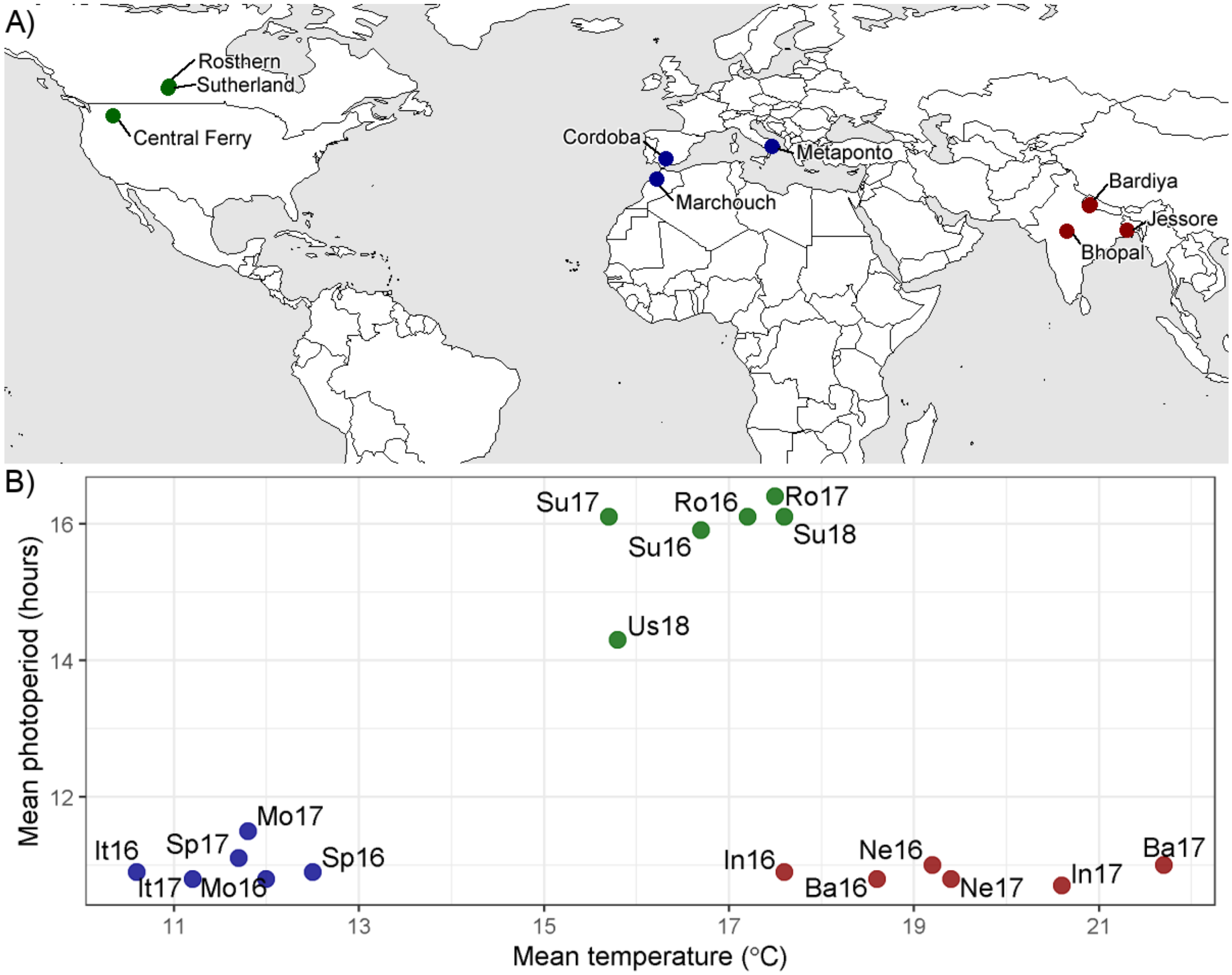
- `ldp` = Lentil Diversity Panel Metadata
- `rr` = Raw Phenotype Data
- `dd` = Averaged Phenotype Data
- `ee` = Environmental Data
- `ff` = Field Trial Info
- `ct` = Country Info

Supplemental tables 1 & 2

```r
write.csv(select(ldp, Entry, Name, Origin, Source),
          "Supplemental_Table_01.csv", row.names = F)
write.csv(select(ff, Location, Year,
            `Short Name` = ExptShort, `Planting Date` = Start, Latitude = Lat,
            Longitude = Lon, `Temperature (mean)` = T_mean, `Photoperiod (mean)` = P_mean,
            `Number of Seeds Sown` = Number_of_Seeds_Sown, `Plot Type` = Plot_Type),
          "Supplemental_Table_02.csv", row.names = F)
```

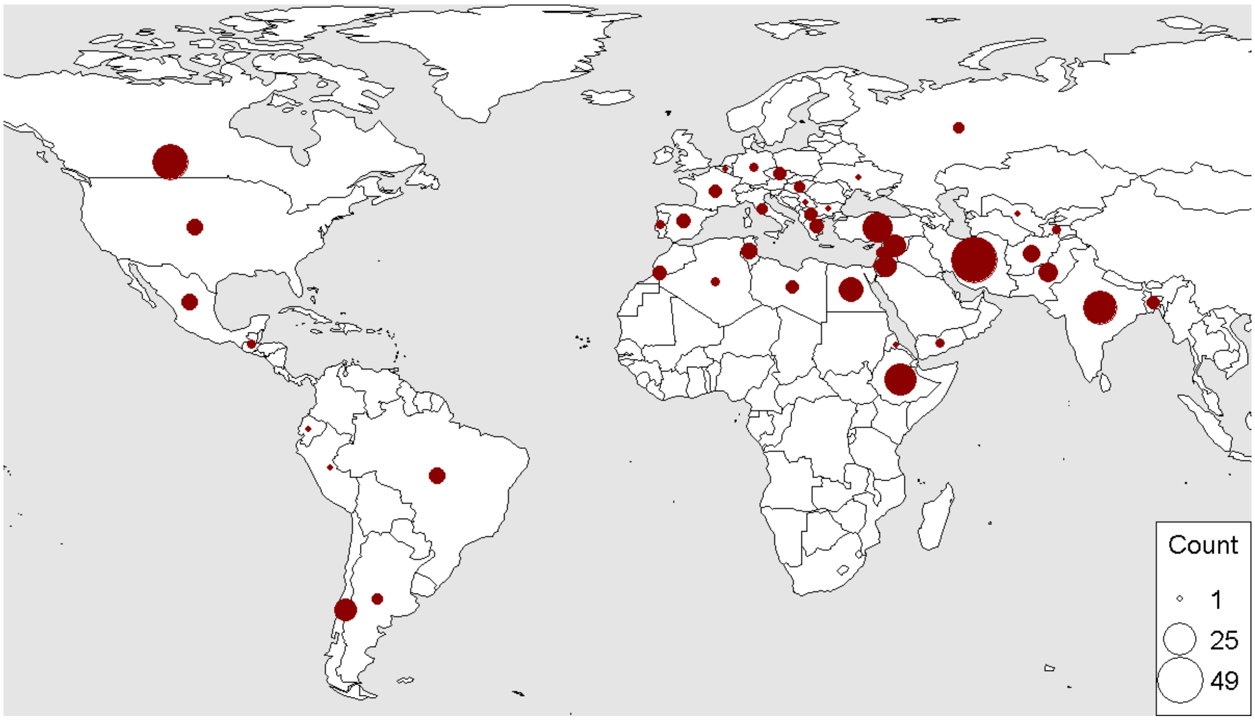
Figure 1 Field Trial Info

```r
# Prep data
xx <- ff %>% mutate(size = 1)
# Plot A) Map
invisible(png("Additional/Temp/Temp_F01_1.png", width = 1200, height = 450, res = 150))
par(mai = c(0,0,0,0), xaxs = "i", yaxs = "i")
mapBubbles(dF = xx, nameX = "Lon", nameY = "Lat", nameZColour = "MacroEnv",
           nameZSize = "size", symbolSize = 0.5, pch = 20, fill = F,
           colourPalette = c("darkgreen","darkred","darkblue"), addColourLegend = F, addLegend = F,
           xlim = c(-140,110), ylim = c(10,35),
           oceanCol = "grey90", landCol = "white", borderCol = "black")
invisible(dev.off())
# Plot B) mean T and P
mp <- ggplot(ff, aes(x = T_mean, y = P_mean)) +
  geom_point(aes(color = MacroEnv), size = 3, alpha = 0.8) +
  geom_text_repel(aes(label = ExptShort)) +
  scale_x_continuous(breaks = c(11,13,15,17,19,21)) +
  scale_color_manual(values = c("darkgreen","darkred","darkblue")) +
  theme_AGL +
  theme(legend.position = "none") +
  labs(x = expression(paste("Mean temperature (", degree, "C)", sep = "")),
       y = "Mean photoperiod (hours)")
ggsave("Additional/Temp/Temp_F01_2.png", mp, width = 7, height = 3)
#
# Labels were added to "Additional/Temp/Temp_F1_1.png" in image editing software
# Append A) and B)
im1 <- image_read("Additional/Temp/Temp_F01_1_1.png") %>%
  image_annotate("A)", size = 30, boxcolor = "white")
im2 <- image_read("Additional/Temp/Temp_F01_2.png") %>%
  image_scale("1200x") %>%
  image_annotate("B)", size = 30, boxcolor = "white")
im <- image_append(c(im1,im2), stack = T)
image_write(im, "Figure_01_FieldTrialInfo.png")
```



Additional Figure 1 LDP Origin Map

```r
# Prep data
x1 <- ldp %>% filter(Origin != "ICARDA") %>%
  group_by(Origin) %>% summarise(Count = n()) %>%
  left_join(select(ct, Origin = Country, Lat, Lon), by = "Origin") %>%
  ungroup() %>% as.data.frame()
x1[is.na(x1)] <- 0
# Plot map
invisible(png("Additional/Additional_Figure_01_LDPOriginMap.png",
              width = 1200, height = 685, res = 150))
par(mai = c(0,0,0,0), xaxs = "i",yaxs = "i")
mapBubbles(dF = x1, nameX = "Lon", nameY = "Lat",
        nameZSize = "Count", nameZColour = "darkred",
        xlim = c(-140,110), ylim = c(5,20),
        oceanCol = "grey90", landCol = "white", borderCol = "black")
invisible(dev.off())
```



Supplemental Figure 1 Scaling

```
# Prep data
levs <- c("Days from sowing to flower (days)", "Scaled (1-5)")
xx <- dd %>% select(Entry, Expt, ExptShort, DTF, DTF2_scaled) %>%
  left_join(select(ff, Expt, MacroEnv), by = "Expt") %>%
  gather(Trait, Value, DTF, DTF2_scaled) %>%
  mutate(Trait = plyr::mapvalues(Trait, c("DTF", "DTF2_scaled"), levs),
         Trait = factor(Trait, levels = levs) )
# Plot DTF
mp <- ggplot(xx, aes(x = ExptShort, y = Value)) +
  geom_violin(fill = "grey", alpha = 0.3, color = NA) +
  geom_quasirandom(aes(color = MacroEnv), size = 0.1, alpha = 0.5) +
  scale_color_manual(values = c("darkgreen","darkred","darkblue")) +
  facet_grid(Trait ~ MacroEnv, scales = "free") +
  theme_AGL +
  theme(legend.position = "none",
        axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1)) +
  labs(x = NULL, y = NULL)
ggsave("Supplemental_Figure_01_Scaling.png", mp, width = 8, height = 5)
```

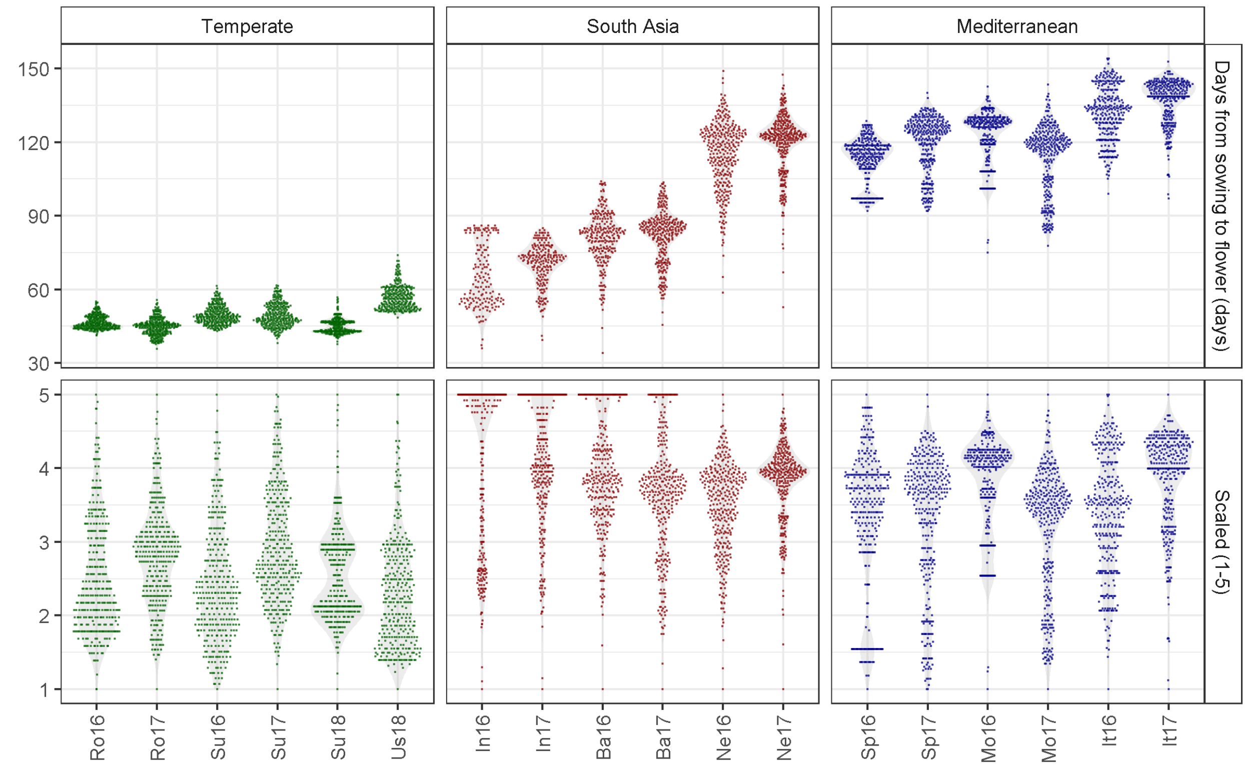
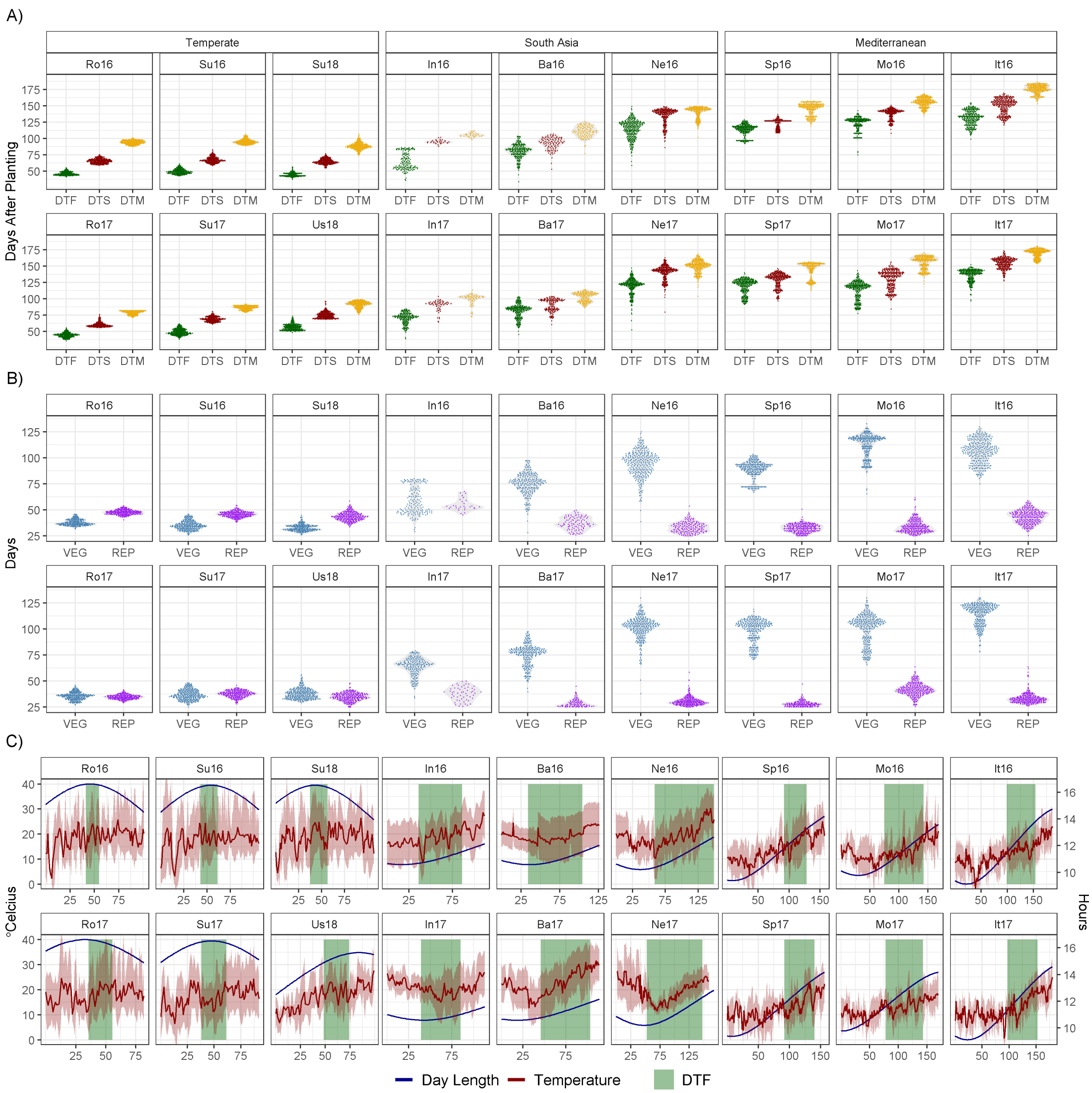


Figure 2 Data Overview

```
# Prep data
xx <- dd %>% select(Entry, Year, Expt, ExptShort, Location, DTF, DTS, DTM) %>%
  left_join(select(ff, Expt, MacroEnv), by = "Expt") %>%
  gather(Trait, Value, DTF, DTS, DTM) %>%
  mutate(Trait = factor(Trait, levels = c("DTF", "DTS", "DTM")))
# Create plot function
ggDistroDTF <- function(x) {
  mp <- ggplot(x, aes(x = Trait, y = Value) ) +
    geom_violin(color = NA, fill = "grey", alpha = 0.3) +
    geom_quasirandom(size = 0.02, aes(color = Trait)) +
    facet_wrap(ExptShort ~ ., scales = "free_x", dir = "v", ncol = 3, nrow = 2) +
    scale_color_manual(values = c("darkgreen", "darkred", "darkgoldenrod2")) +
    scale_y_continuous(sec.axis = dup_axis(name = "\n"),
                       limits = c(30,190), breaks = c(50,75,100,125,150,175)) +
    theme_AGL +
    theme(axis.text.x = element_blank(), axis.ticks.x = element_blank()) +
    labs(x = NULL, y = NULL)
  mp
}
# Plot A) DTF, DTS and DTM
mp1.1 <- ggDistroDTF(xx %>% filter(MacroEnv == "Temperate"))
mp1.2 <- ggDistroDTF(xx %>% filter(MacroEnv == "South Asia"))
```

```r
mp1.3 <- ggDistroDTF(xx %>% filter(MacroEnv == "Mediterranean"))
mp1 <- ggmatrix(list(mp1.1, mp1.2, mp1.3), nrow = 1, ncol = 3,
                title = "A)", ylab = "Days After Planting",
                xAxisLabels = macroEnvs) +
  theme_AGL +
  theme(plot.margin = unit(c(0,1,0,0), "cm"),
        plot.title = element_text(hjust = -0.04))
# Prep data
xx <- dd %>% select(Entry, Name, Expt, ExptShort, Location, Year, VEG, REP) %>%
  left_join(select(ff, Expt, MacroEnv), by = "Expt") %>%
  gather(Trait, Value, VEG, REP) %>%
  mutate(Trait = factor(Trait, levels = c("VEG", "REP")))
# Create plot function
ggDistroREP <- function(x) {
  mp <- ggplot(x, aes(x = Trait, y = Value)) +
    geom_violin(color = NA, fill = "grey", alpha = 0.3) +
    geom_quasirandom(size = 0.02, aes(color = Trait)) +
    facet_wrap(ExptShort ~ ., scales = "free_x", dir = "v", ncol = 3, nrow = 2) +
    scale_color_manual(values = c("steelblue", "purple")) +
    scale_y_continuous(sec.axis = dup_axis(name = "\n"),
                       limits = c(25,135), breaks = c(25,50,75,100,125)) +
    theme_AGL +
    labs(x = NULL, y = "Days")
  mp
}
# Plot B) REP and VEG
mp2.1 <- ggDistroREP(xx %>% filter(MacroEnv == "Temperate"))
mp2.2 <- ggDistroREP(xx %>% filter(MacroEnv == "South Asia"))
mp2.3 <- ggDistroREP(xx %>% filter(MacroEnv == "Mediterranean"))
mp2 <- ggmatrix(list(mp2.1, mp2.2, mp2.3),
                nrow = 1, ncol = 3, title = "B)", ylab = "Days") +
  theme_AGL +
  theme(plot.margin = unit(c(0,1,0,0), "cm"),
        plot.title = element_text(hjust = -0.04))
# Create plot function
ggEnvPlot <- function(x, nr = 2, nc = 3, mybreaks) {
  yy <- ff %>% filter(Expt %in% unique(x$Expt)) %>%
    select(ExptShort, Location, Year, min=DTF_min, max=DTF_max) %>%
    mutate(Trait = "DTF")
  mp <- ggplot(x) +
    geom_rect(data = yy, aes(xmin = min, xmax = max, fill = Trait),
              ymin = 0, ymax = 40, alpha = 0.4) +
    geom_line(aes(x = DaysAfterPlanting, y = DayLength_rescaled, color = "Day Length")) +
    geom_line(aes(x = DaysAfterPlanting, y = Temp_mean, color = "Temperature") ) +
    geom_ribbon(aes(x = DaysAfterPlanting, ymin = Temp_min, ymax = Temp_max),
                fill = "darkred", alpha = 0.3) +
    facet_wrap(ExptShort ~ ., scales = "free_x", dir = "v", nrow = 2, ncol = 3) +
    scale_x_continuous(breaks = mybreaks) +
    scale_color_manual(name = NULL, values = c("darkblue", "darkred")) +
    scale_fill_manual(name = NULL, values = "darkgreen") +
    coord_cartesian(ylim=c(0, 40)) +
    theme_AGL +
    theme(legend.position = "bottom",
```

```
          legend.text = element_text(size = 12)) +
    labs(y = NULL, x = NULL) +
    guides(colour = guide_legend(order = 1, override.aes = list(size = 1.25)),
           fill = guide_legend(order = 2))
  mp
}
# Plot C) T and P
mp3.1 <- ggEnvPlot(ee %>% filter(MacroEnv == "Temperate"), mybreaks = c(25,50,75)) +
  labs(title = "C)", y = expression(paste(degree, "Celcius"))) +
  theme(plot.margin = unit(c(0,0,0,0), "cm"),
        plot.title = element_text(hjust = -0.11))
mp3.2 <- ggEnvPlot(ee %>% filter(MacroEnv == "South Asia"), mybreaks = c(25,75,125)) +
  theme(plot.margin = unit(c(0,0,0,0), "cm"),
        axis.text.y = element_blank(),
        axis.ticks.y = element_blank())
mp3.3 <- ggEnvPlot(ee %>% filter(MacroEnv == "Mediterranean"), mybreaks = c(50,100,150)) +
  scale_y_continuous(sec.axis = sec_axis(~ (16.62 - 9.11) * . / (40 - 0) + 9.11,
                     name = "Hours", breaks = c(10, 12, 14, 16))) +
  theme(plot.margin = unit(c(0,0,0,0), "cm"),
        axis.text.y.left = element_blank(),
        axis.ticks.y.left = element_blank())
mp3 <- ggarrange(mp3.1, mp3.2, mp3.3, nrow = 1, ncol = 3, align = "h",
                 legend = "bottom", common.legend = T, widths = c(1.1,1,1.1))
ggsave("Additional/Temp/Temp_F02_1.png", mp1, width = 12, height = 4)
ggsave("Additional/Temp/Temp_F02_2.png", mp2, width = 12, height = 4)
ggsave("Additional/Temp/Temp_F02_3.png", mp3, width = 12, height = 4)
# Append A), B) and C)
mp1 <- image_read("Additional/Temp/Temp_F02_1.png")
mp2 <- image_read("Additional/Temp/Temp_F02_2.png")
mp3 <- image_read("Additional/Temp/Temp_F02_3.png")
mp <- image_append(c(mp1, mp2, mp3), stack = T)
image_write(mp, "Figure_02_DataOverview.png")
```



Additional Figures - Entry Phenology

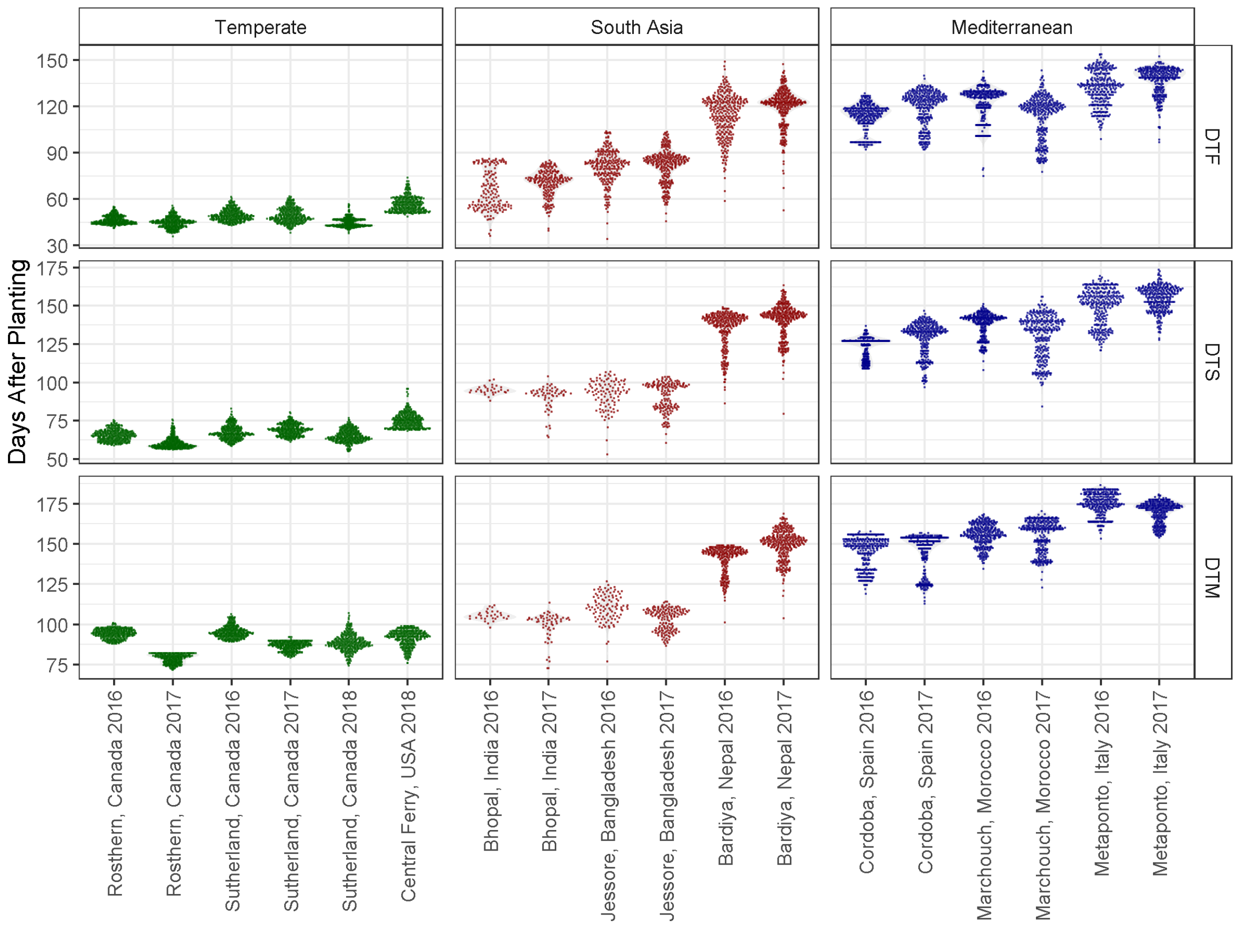
```r
# Create plotting function
gg_phenol <- function(x, xE, colnums) {
  mycols <- c("darkgreen", "darkred", "darkblue")
  ggplot(xE, aes(x = Trait, y = Value, group = Entry, color = MacroEnv)) +
    geom_line(data = x, color = "grey", alpha = 0.5) +
    geom_line() + geom_point() +
    facet_grid(MacroEnv ~ ExptShort) +
```

```
    scale_color_manual(values = mycols[colnums]) +
    theme_AGL +
    theme(legend.position = "none") +
    ylim(c(min(x$Value, na.rm = T), max(x$Value, na.rm = T))) +
    labs(x = NULL, y = "Days")
}
# Prep data
xx <- dd %>% select(Entry, Name, ExptShort, DTF, DTS, DTM) %>%
  left_join(select(ff, ExptShort, MacroEnv), by = "ExptShort") %>%
  gather(Trait, Value, DTF, DTS, DTM) %>%
  mutate(Trait = factor(Trait, levels = c("DTF","DTS","DTM")))
x1 <- xx %>% filter(MacroEnv == "Temperate")
x2 <- xx %>% filter(MacroEnv == "South Asia")
x3 <- xx %>% filter(MacroEnv == "Mediterranean")
# Create PDF
pdf("Additional/pdf_Phenology.pdf", width = 8, height = 6)
#par(mar=c(1.5, 2.5, 1.5, 0.5))
for(i in 1:324) {
  xE1 <- xx %>% filter(Entry == i, !is.na(Value), MacroEnv == "Temperate")
  xE2 <- xx %>% filter(Entry == i, !is.na(Value), MacroEnv == "South Asia")
  xE3 <- xx %>% filter(Entry == i, !is.na(Value), MacroEnv == "Mediterranean")
  mp1 <-  gg_phenol(x1, xE1, 1)
  mp2 <-  gg_phenol(x2, xE2, 2)
  mp3 <-  gg_phenol(x3, xE3, 3)
  figlab <- paste("Entry", str_pad(i, 3, "left", "0"), "|", unique(xE1$Name))
  mp <- ggarrange(mp1, mp2, mp3, nrow = 3, ncol = 1) %>%
    annotate_figure(top = figlab)
  print(mp)
  ggsave(paste0("Additional/Entry_Phenology/Phenology_Entry", str_pad(i, 3, "left", "0"), ".png"),
         mp, width = 8, height = 6)
}
dev.off()#dev.set(dev.next())
```

Additional Figure 2 DTF DTS DTM

```
# Prep data
xx <- dd %>% select(Entry, Expt, ExptShort, DTF, DTS, DTM) %>%
  left_join(select(ff, Expt, MacroEnv), by = "Expt") %>%
  gather(Trait, Value, DTF, DTS, DTM) %>%
  mutate(Trait = factor(Trait, levels = c("DTF", "DTS", "DTM")) )
# Plot DTF, DTS, DTM
mp <- ggplot(xx, aes(x = Expt, y = Value)) +
  geom_violin(fill = "grey", alpha = 0.25, color = NA) +
  geom_quasirandom(size = 0.1, alpha = 0.5, aes(color = MacroEnv)) +
  facet_grid(Trait ~ MacroEnv, scales = "free") +
  scale_color_manual(values = c("darkgreen", "darkred", "darkblue")) +
  theme_AGL +
  theme(legend.position = "none",
        axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1)) +
```

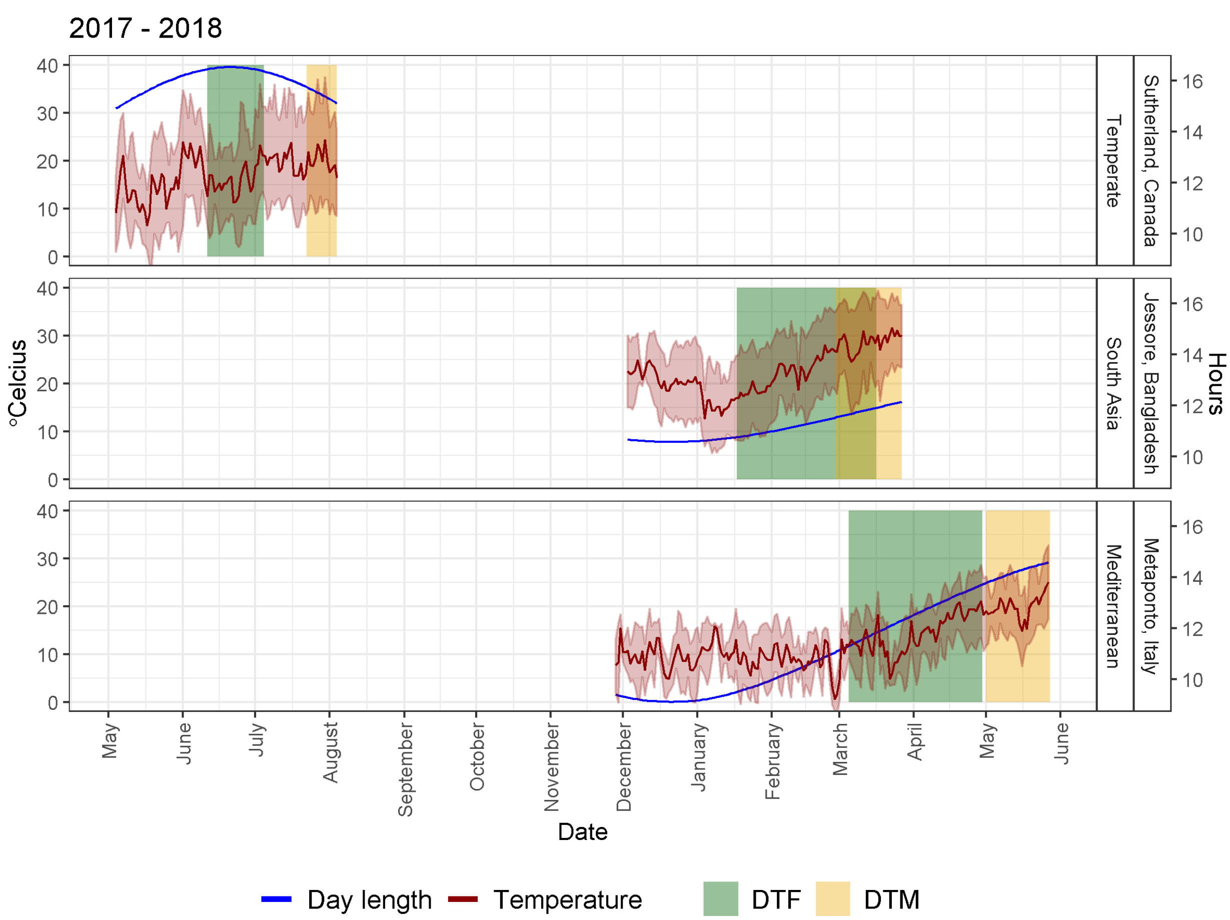
```
  labs(x = NULL, y = "Days After Planting")
ggsave("Additional/Additional_Figure_02_DTFDTSDTM.png", mp, width = 8, height = 6)
```



Additional Figure 3 MacroEnv Phenology

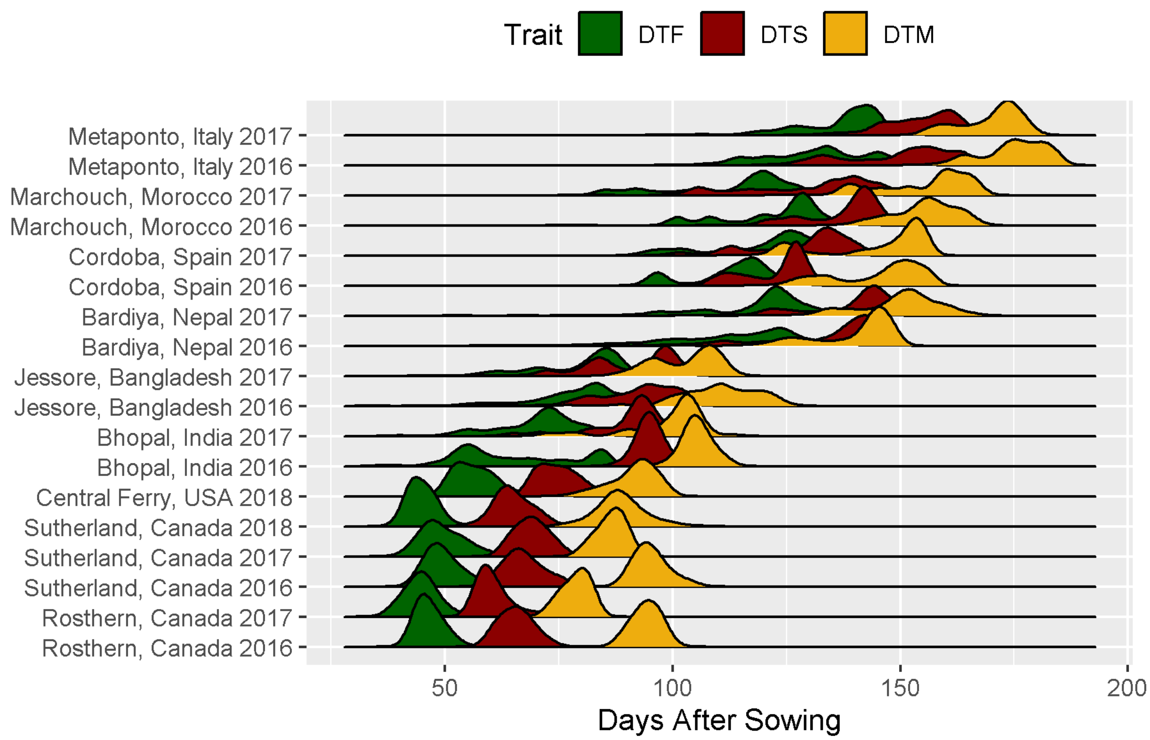
```
# Prep data
xx <- ee %>% filter(ExptShort %in% c("Su17", "Ba17", "It17"))
yy <- ff %>% filter(Expt %in% unique(xx$Expt)) %>%
  mutate(DTF_min = Start + DTF_min, DTF_max = Start + DTF_max,
         DTM_min = Start + DTM_min, DTM_max = Start + DTM_max)
y1 <- select(yy, Expt, Location, Year, MacroEnv, min = DTF_min, max = DTF_max) %>%
  mutate(Trait = "DTF")
y2 <- select(yy, Expt, Location, Year, MacroEnv, min = DTM_min, max = DTM_max) %>%
  mutate(Trait = "DTM")
yy <- bind_rows(y1, y2)
# Plot DTF, DTM, T and P
mp <- ggplot(xx) +
  geom_rect(data = yy, aes(xmin = min, xmax = max, fill = Trait),
```

```r
            ymin = 0, ymax = 40, alpha = 0.4) +
  geom_line(aes(x = Date, y = DayLength_rescaled, color = "Blue")) +
  geom_line(aes(x = Date, y = Temp_mean, color = "darkred") ) +
  geom_ribbon(aes(x = Date, ymin = Temp_min, ymax = Temp_max),
              fill = alpha("darkred", 0.25), color = alpha("darkred", 0.25)) +
  facet_grid(Location +MacroEnv  ~ ., scales = "free_x", space = "free_x") +
  scale_color_manual(name = NULL, values = c("Blue", "darkred"),
                     labels = c("Day length", "Temperature") ) +
  scale_fill_manual(name = NULL, values = c("darkgreen", "darkgoldenrod2")) +
  coord_cartesian(ylim = c(0,40)) +
  theme_AGL +
  theme(legend.position = "bottom",
        legend.text = element_text(size = 12),
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  scale_x_date(breaks = "1 month", labels = date_format("%B")) +
  scale_y_continuous(sec.axis = sec_axis(~ (16.62 - 9.11) * . / (40 - 0) + 9.11,
                     breaks = c(10, 12, 14, 16), name = "Hours")) +
  labs(title = "2017 - 2018", y = expression(paste(degree, "Celcius"), x = NULL)) +
  guides(colour = guide_legend(order = 1, override.aes = list(size = 1.25)),
         fill   = guide_legend(order = 2))
ggsave("Additional/Additional_Figure_03_MacroEnvPhenology.png", mp, width = 8, height = 6)
```



Additional Figure 4 ggridges

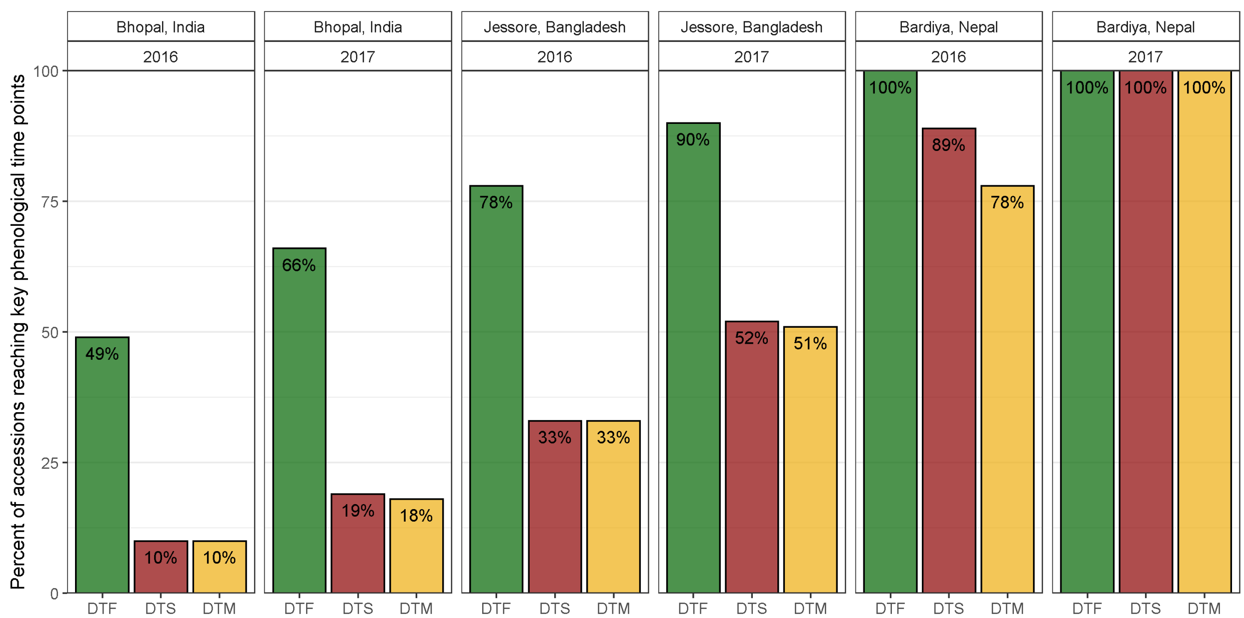
```
xx <- dd %>% select(Expt, DTF, DTS, DTM) %>%
  gather(Trait, Value, DTF, DTS, DTM) %>%
  mutate(Trait = factor(Trait, levels = c("DTF", "DTS", "DTM")))
mp <- ggplot(xx, aes(x = Value, y = Expt, fill = Trait)) +
  ggridges::geom_density_ridges() +
  scale_fill_manual(values = c("darkgreen", "darkred", "darkgoldenrod2")) +
  theme(legend.position = "top") +
  labs(y = NULL, x = "Days After Sowing")
ggsave("Additional/Additional_Figure_04_ggridges.png", mp, width = 6, height = 4)
```



Supplemental Figure 2 Missing Data

```
# Prep data
xx <- dd %>%
  filter(Location %in% c("Bhopal, India", "Jessore, Bangladesh", "Bardiya, Nepal")) %>%
  mutate(DTF = ifelse(is.na(DTF), 0, 1),
         DTS = ifelse(is.na(DTS), 0, 1),
         DTM = ifelse(is.na(DTM), 0, 1) ) %>%
  group_by(Expt, Location, Year) %>%
  summarise_at(vars(DTF, DTS, DTM), funs(sum), na.rm = T) %>%
```

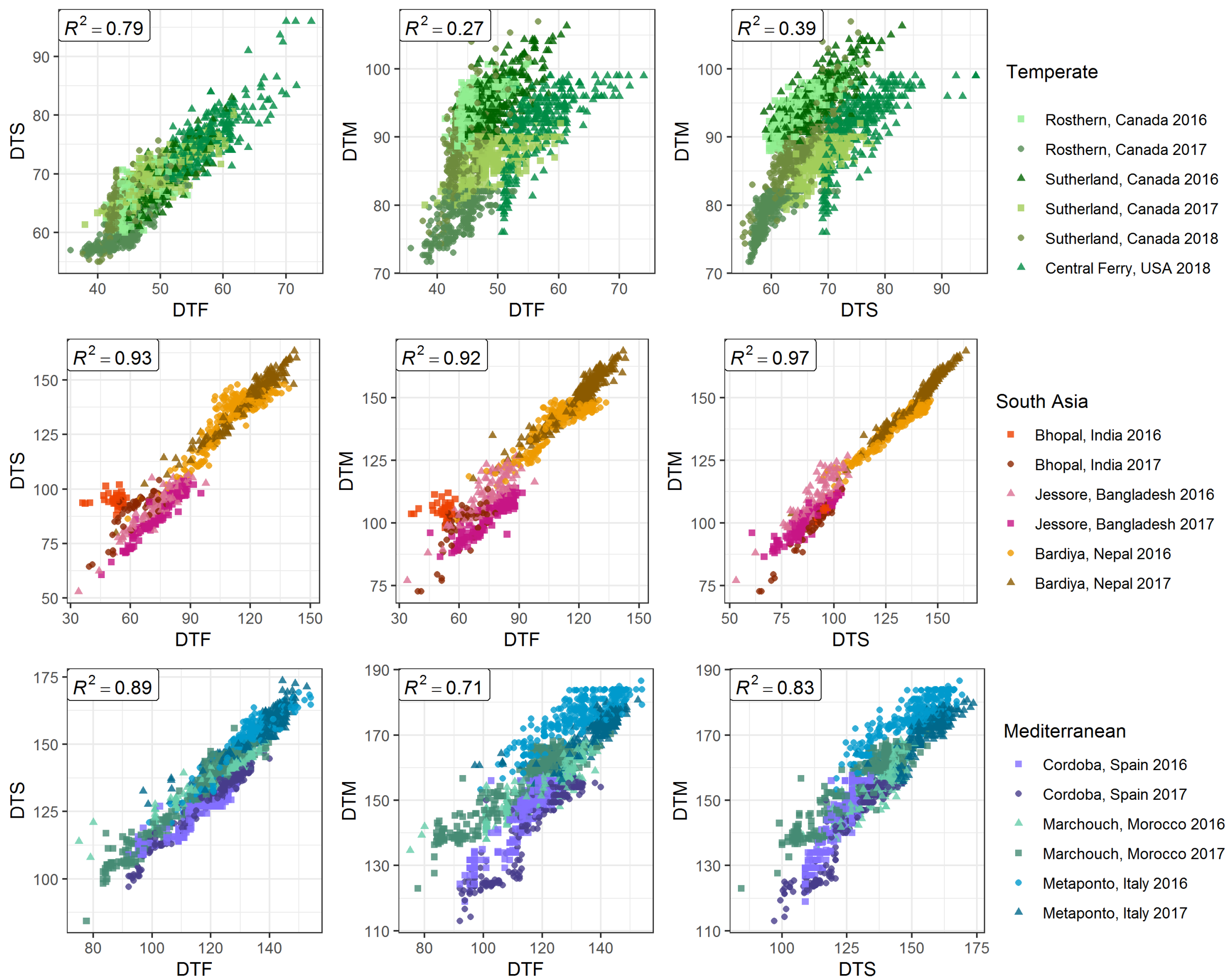
```r
  ungroup() %>%
  gather(Trait, Flowered, DTF, DTS, DTM) %>%
  mutate(Total = ifelse(Expt == "Bardiya, Nepal 2016", 323, 324),
         # One accession was not planted in Bardiya, Nepal 2016
         DidNotFlower = Total - Flowered,
         Percent = round(100 * Flowered / Total),
         Label = paste0(Percent, "%"),
         Trait = factor(Trait, levels = c("DTF", "DTS", "DTM")))
# Plot Percent Flowered
mp <- ggplot(xx, aes(x = Trait, y = Percent, fill = Trait)) +
  geom_bar(stat = "identity", color = "black", alpha = 0.7) +
  geom_text(aes(label = Label), nudge_y = -3, size = 3.5) +
  facet_grid(. ~ Location + Year) +
  scale_fill_manual(values = c("darkgreen", "darkred", "darkgoldenrod2")) +
  scale_y_continuous(limits = c(0,100), expand = c(0,0)) +
  theme_AGL +
  theme(legend.position = "none",
        panel.grid.major.x = element_blank() ) +
  labs(x = NULL, y = "Percent of accessions reaching key phenological time points")
ggsave("Supplemental_Figure_02_PercentFlowered.png", width = 10, height = 5)
```



Supplemental Figure 3 Correlation Plots

```r
# Prep data
xx <- dd %>% left_join(select(ff, Expt, MacroEnv), by = "Expt") %>%
  select(Entry, Expt, MacroEnv, DTF, DTS, DTM)
# Create plotting function
ggCorPlot <- function(x, legend.title, colNums) {
```

```
  # Plot A)
  r2 <- round(cor(x$DTF, x$DTS, use ="complete", method = "pearson")^2, 2)
  tp1 <- ggplot(x) + theme_AGL +
    geom_point(aes(x = DTF, y = DTS, color = Expt, shape = Expt), alpha = 0.8) +
    geom_label(x = -Inf, y = Inf, hjust = 0, vjust = 1, parse = T,
               label = paste("italic(R)^2 == ", r2) ) +
    scale_color_manual(name = legend.title, values = colors_Expt[colNums]) +
    scale_shape_manual(name = legend.title, values = c(15,16,17,15,16,17))
  # Plot B)
  r2 <- round(cor(x$DTF, x$DTM, use ="complete.obs", method = "pearson")^2, 2)
  tp2 <- ggplot(x) + theme_AGL +
    geom_point(aes(x = DTF, y = DTM, color = Expt, shape = Expt), alpha = 0.8) +
    geom_label(x = -Inf, y = Inf, hjust = 0, vjust = 1, parse = T,
               label = paste("italic(R)^2 == ", r2) ) +
    scale_color_manual(name = legend.title, values = colors_Expt[colNums]) +
    scale_shape_manual(name = legend.title, values = c(15,16,17,15,16,17))
  # Plot C)
  r2 <- round(cor(x$DTS, x$DTM, use = "complete", method = "pearson")^2, 2)
  tp3 <- ggplot(x) + theme_AGL +
    geom_point(aes(x = DTS, y = DTM, color = Expt, shape = Expt), alpha = 0.8) +
    geom_label(x = -Inf, y = Inf, hjust = 0, vjust = 1, parse = T,
               label = paste("italic(R)^2 == ", r2) ) +
    scale_color_manual(name = legend.title, values = colors_Expt[colNums]) +
    scale_shape_manual(name = legend.title, values = c(15,16,17,15,16,17))
  # Append A), B) and C)
  mp <- ggarrange(tp1, tp2, tp3, nrow = 1, ncol = 3,
                  common.legend = T, legend = "right")
  mp
}
# Plot Correlations
mp1 <- ggCorPlot(xx %>% filter(MacroEnv == "Temperate"),     "Temperate",      1:6 )
mp2 <- ggCorPlot(xx %>% filter(MacroEnv == "South Asia"),    "South Asia",     7:12)
mp3 <- ggCorPlot(xx %>% filter(MacroEnv == "Mediterranean"), "Mediterranean", 13:18)
mp <- ggarrange(mp1, mp2, mp3, nrow = 3, ncol = 1, common.legend = T, legend = "right")
ggsave("Supplemental_Figure_03_Correlations.png", mp, width = 10, height = 8)
```



Additional Figures - Correlations

```
# Prep data
xx <- dd %>% left_join(select(ff, Expt, MacroEnv), by = "Expt") %>%
  mutate(DTE = ifelse(Location == "Cordoba, Spain", NA, DTE))
x1 <- xx %>% filter(MacroEnv == "Temperate")
x2 <- xx %>% filter(MacroEnv == "South Asia")
x3 <- xx %>% filter(MacroEnv == "Mediterranean")
# Create plotting functions
my_lower <- function(data, mapping, cols = colors_Expt, ...) {
  ggplot(data = data, mapping = mapping) +
    geom_point(alpha = 0.5, size = 0.3, aes(color = Expt)) +
    theme_bw() +
    scale_color_manual(values = cols)
}
my_middle <- function(data, mapping, cols = colors_Expt, ...) {
  ggplot(data = data, mapping = mapping) +
```

```r
    geom_density(alpha = 0.5) + theme_bw() +
    scale_color_manual(name = NULL, values = cols) +
    scale_fill_manual(name = NULL, values = cols) +
    guides(color = F, fill = guide_legend(nrow = 3, byrow = T))
}
# See: https://github.com/ggobi/ggally/issues/139
my_upper <- function(data, mapping, color = I("black"), sizeRange = c(1,5), ...) {
  # Prep data
  x <- eval_data_col(data, mapping$x)
  y <- eval_data_col(data, mapping$y)
  #
  r2 <- cor(x, y, method = "pearson", use = "complete.obs")^2
  rt <- format(r2, digits = 2)[1]
  cex <- max(sizeRange)
  tt <- as.character(rt)
  # plot the cor value
  p <- ggally_text(label = tt, mapping = aes(), color = color,
                   xP = 0.5, yP = 0.5, size = 6,  ... ) + theme_bw()
  # Create color palette
  corColors <- RColorBrewer::brewer.pal(n = 10, name = "RdBu")[2:9]
  if        (r2 <= -0.9)              { corCol <- alpha(corColors[1], 0.5)
  } else if (r2 >= -0.9 & r2 <= -0.6) { corCol <- alpha(corColors[2], 0.5)
  } else if (r2 >= -0.6 & r2 <= -0.3) { corCol <- alpha(corColors[3], 0.5)
  } else if (r2 >= -0.3 & r2 <= 0)    { corCol <- alpha(corColors[4], 0.5)
  } else if (r2 >= 0    & r2 <= 0.3)  { corCol <- alpha(corColors[5], 0.5)
  } else if (r2 >= 0.3  & r2 <= 0.6)  { corCol <- alpha(corColors[6], 0.5)
  } else if (r2 >= 0.6  & r2 <= 0.9)  { corCol <- alpha(corColors[7], 0.5)
  } else                              { corCol <- alpha(corColors[8], 0.5) }
  # Plot
  p <- p +
    theme(panel.background = element_rect(fill = corCol),
          panel.grid.major = element_blank(),
          panel.grid.minor = element_blank(),
          axis.text = element_text(size = 5))
  p
}
# Plot Correlations for each Expt
for(i in 1:length(names_Expt)) {
  mp <- ggpairs(xx %>% filter(Expt == names_Expt[i]),
               columns = c("DTE", "DTF", "DTS", "DTM", "REP"),
        upper  = list(continuous = my_upper),
        diag   = list(continuous = my_middle),
        lower  = list(continuous = wrap(my_lower, cols = "black")),
        title  = i) +
    theme(strip.background = element_rect(fill = "White"))
  ggsave(paste0("Additional/Corr/Corr_", str_pad(i,2,"left","0"), "_", names_Expt[i], ".png"),
         mp, width = 6, height = 6)
}
# Plot A) Temperate
mp1 <- ggpairs(x1, columns = c("DTE", "DTF", "DTS", "DTM", "REP"),
               aes(color = Expt, fill = Expt),
        upper=list(continuous = my_upper),
        diag =list(continuous = wrap(my_middle, cols = colors_Expt[1:6])),
```

```
       lower=list(continuous = wrap(my_lower,  cols = colors_Expt[1:6])),
       title = "A) Temperate",
       legend = c(2,2)) +
  theme(strip.background = element_rect(fill = "White"),
        legend.position = "bottom")
ggsave("Additional/Corr/Corr_Temperate.png", mp1, width = 6, height = 6)
# Plot B) South Asia
mp2 <- ggpairs(x2, columns = c("DTE", "DTF", "DTS", "DTM", "REP"),
               aes(color = Expt, fill = Expt),
        upper  = list(continuous = my_upper),
        diag   = list(continuous = wrap(my_middle, cols = colors_Expt[7:12])),
        lower  = list(continuous = wrap(my_lower,  cols = colors_Expt[7:12])),
        title  = "B) South Asia",
        legend = c(2,2)) +
  theme(strip.background = element_rect(fill = "White"),
        legend.position = "bottom")
ggsave("Additional/Corr/Corr_SouthAsia.png", mp2, width = 6, height = 6)
# Plot C) Mediterranean
mp3 <- ggpairs(x3, columns = c("DTE", "DTF", "DTS", "DTM", "REP"),
               aes(color = Expt, fill = Expt),
        upper  = list(continuous = my_upper),
        diag   = list(continuous = wrap(my_middle, cols = colors_Expt[13:18])),
        lower  = list(continuous = wrap(my_lower,  cols = colors_Expt[13:18])),
        title  = "C) Mediterranean",
        legend = c(2,2)) +
  theme(strip.background = element_rect(fill = "White"),
        legend.position = "bottom")
ggsave("Additional/Corr/Corr_Mediterranean.png", mp3, width = 6, height = 6)
# Plot All
mp4 <- ggpairs(xx, columns = c("DTE", "DTF", "DTS", "DTM", "REP"),
        aes(color = ExptShort, fill = ExptShort),
        upper  = list(continuous = my_upper),
        diag   = list(continuous = my_middle),
        lower  = list(continuous = my_lower),
        title  = "D) ALL",
        legend = c(2,2)) +
  theme(strip.background = element_rect(fill = "White"),
        legend.position = "bottom")
ggsave("Additional/Corr/Corr_All.png", mp4, width = 6, height = 6)
```

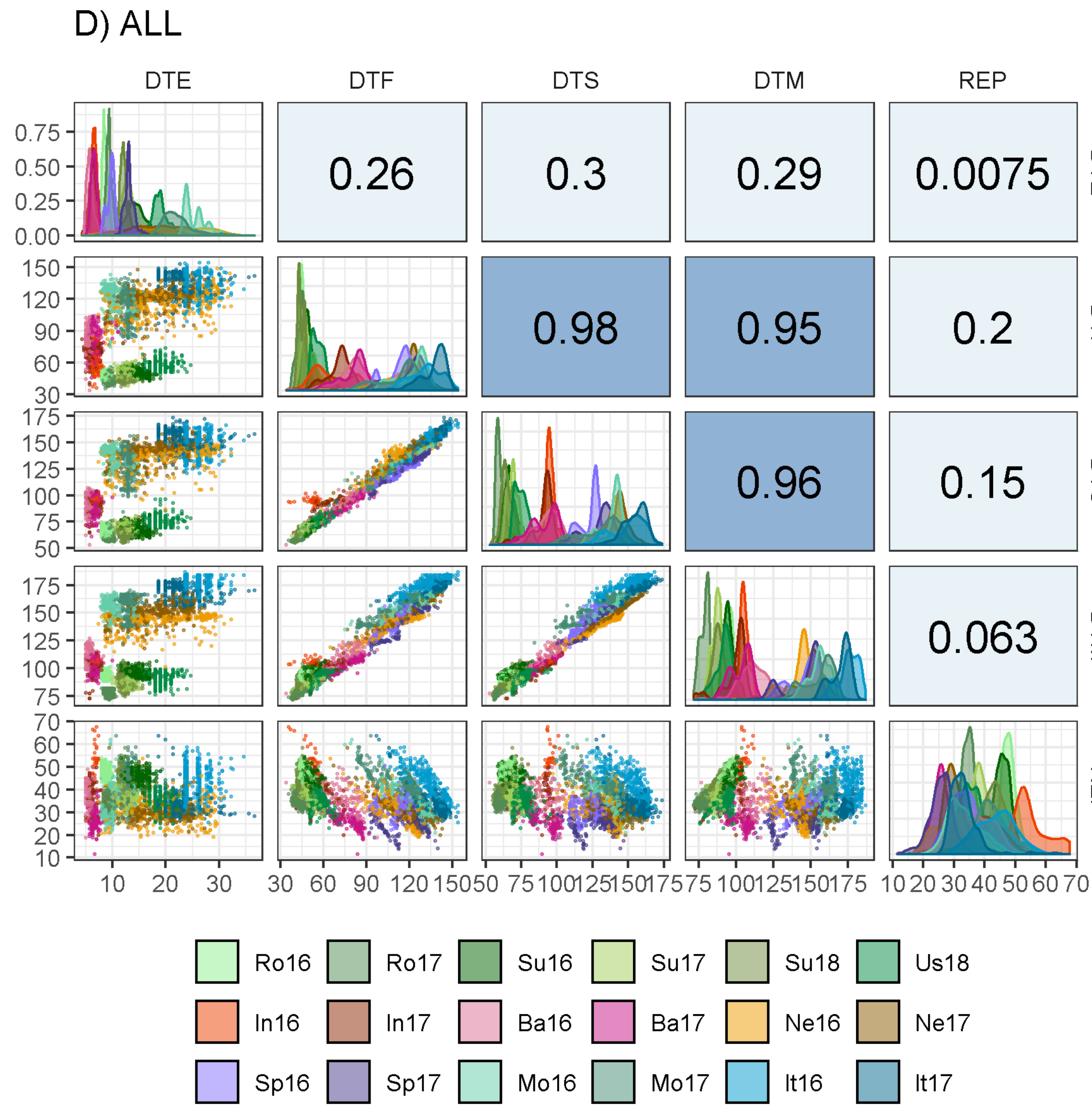


Figure 3 PCA

```
# Prep data
xx <- dd %>% select(Entry, Expt, DTF2_scaled) %>%
  spread(Expt, DTF2_scaled)
xx <- xx %>% column_to_rownames("Entry") %>% as.matrix()
# PCA
mypca <- PCA(xx, ncp = 10, graph = F)
# Heirarcical clustering
mypcaH <- HCPC(mypca, nb.clust = 8, graph = F)
perc <- round(mypca[[1]][,2], 1)
x1 <- mypcaH[[4]]$X %>%
```

```r
  rownames_to_column("Entry") %>%
  mutate(Entry = as.numeric(Entry)) %>%
  rename(PC1=Dim.1, PC2=Dim.2, PC3=Dim.3, PC4=Dim.4, PC5=Dim.5,
         PC6=Dim.6, PC7=Dim.7, PC8=Dim.8, PC9=Dim.9, PC10=Dim.10,
         Cluster=clust) %>%
  left_join(select(ldp, Entry, Name, Origin), by = "Entry") %>%
  left_join(select(ct, Origin=Country, Region), by = "Origin") %>%
  select(Entry, Name, Origin, Region, everything())
write.csv(x1, "data/data_PCA_Results.csv", row.names = F)
#
x2 <- dd %>% left_join(select(x1, Entry, Cluster), by = "Entry") %>%
  group_by(Expt, ExptShort, Cluster) %>%
  summarise(mean = mean(DTF2_scaled, na.rm = T), sd = sd(DTF2_scaled, na.rm = T)) %>%
  ungroup() %>%
  mutate(ClusterNum = plyr::mapvalues(Cluster, as.character(1:8), summary(x1$Cluster)))
x3 <- x1 %>% count(Cluster) %>%
  mutate(Cluster = factor(Cluster, levels = rev(levels(Cluster))), y = n/2)
for(i in 2:nrow(x3)) { x3$y[i] <- sum(x3$n[1:(i-1)]) + (x3$n[i]/2) }
# Plot A) PCA 1v2
find_hull <- function(df) df[chull(df[,"PC1"], df[,"PC2"]), ]
polys <- plyr::ddply(x1, "Cluster", find_hull) %>% mutate(Cluster = factor(Cluster))
mp1.1 <- ggplot(x1) +
  geom_polygon(data = polys, alpha = 0.15, aes(x = PC1, y = PC2, fill = Cluster)) +
  geom_point(aes(x = PC1, y = PC2, colour = Cluster)) +
  scale_fill_manual(values = colors) +
  scale_color_manual(values = colors) +
  theme_classic() +
  theme(legend.position = "none") +
  labs(x = paste0("PC1 (", perc[1], "%)"),
       y = paste0("PC2 (", perc[2], "%)"))
# Plot A) PCA 1v3
find_hull <- function(df) df[chull(df[,"PC1"], df[,"PC3"]), ]
polys <- plyr::ddply(x1, "Cluster", find_hull) %>% mutate(Cluster = factor(Cluster))
mp1.2 <- ggplot(x1) +
  geom_polygon(data = polys, alpha = 0.15, aes(x = PC1, y = PC3, fill = Cluster)) +
  geom_point(aes(x = PC1, y = PC3, colour = Cluster)) +
  scale_fill_manual(values = colors) +
  scale_color_manual(values = colors) +
  theme_classic() +
  theme(legend.position = "none") +
  labs(x = paste0("PC1 (", perc[1], "%)"),
       y = paste0("PC3 (", perc[3], "%)"))
# Plot A) PCA 2v3
find_hull <- function(df) df[chull(df[,"PC2"], df[,"PC3"]), ]
polys <- plyr::ddply(x1, "Cluster", find_hull) %>% mutate(Cluster = factor(Cluster))
mp1.3 <- ggplot(x1) +
  geom_polygon(data = polys, alpha = 0.15, aes(x = PC2, y = PC3, fill = Cluster)) +
  geom_point(aes(x = PC2, y = PC3, colour = Cluster)) +
  scale_fill_manual(values = colors) +
  scale_color_manual(values = colors) +
  theme_classic() +
  theme(legend.position = "none") +
  labs(x = paste0("PC2 (", perc[2], "%)"),
```

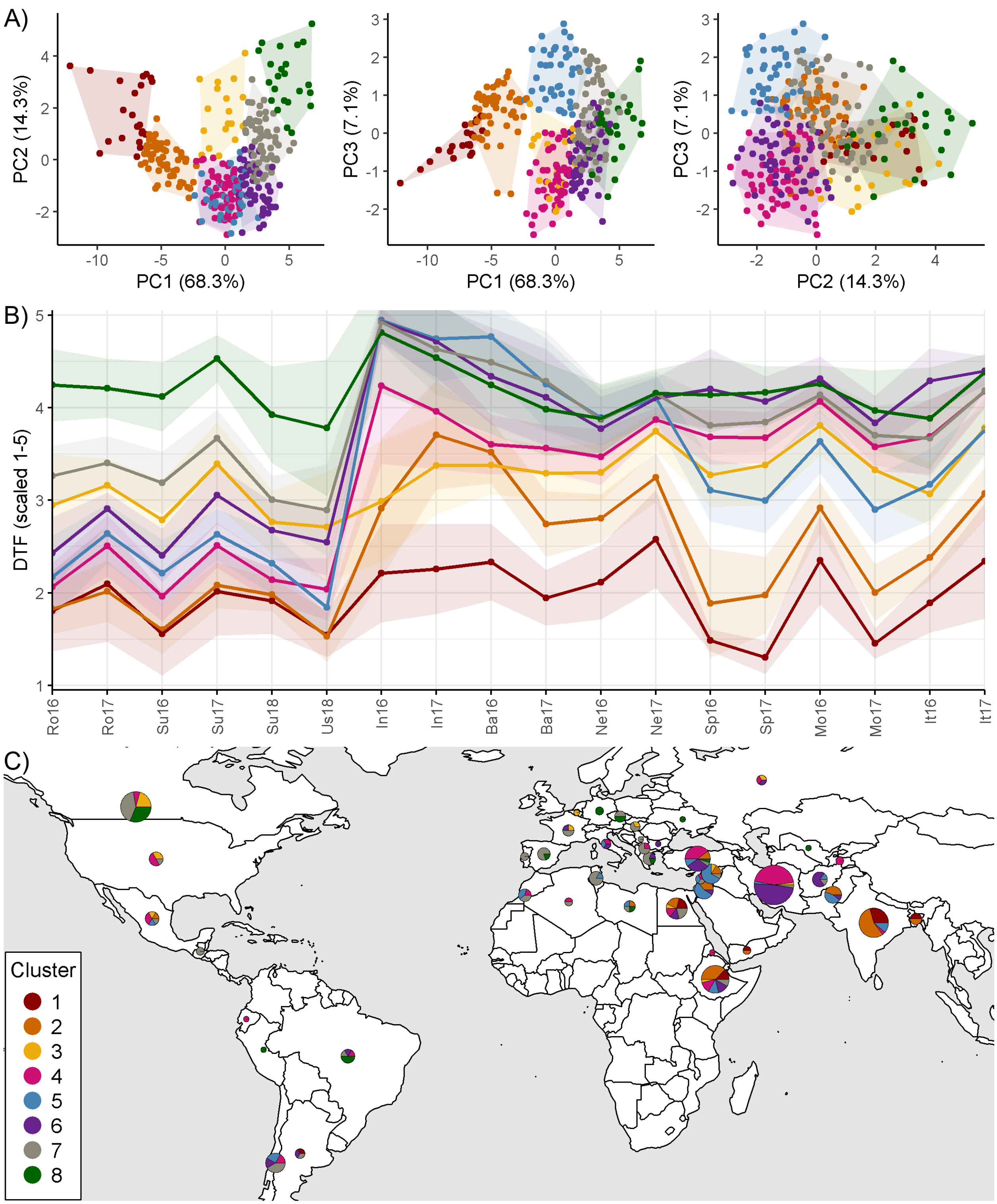
```r
       y = paste0("PC3 (", perc[3], "%)"))
# Append
mp1 <- ggarrange(mp1.1, mp1.2, mp1.3, nrow = 1, ncol = 3, hjust = 0)
# Plot B) DTF
mp2 <- ggplot(x2, aes(x = ExptShort, y = mean, group = Cluster)) +
  geom_ribbon(aes(ymin = mean - sd, ymax = mean + sd, fill = Cluster),
              alpha = 0.1, color = NA) +
  geom_point(aes(color = Cluster)) +
  geom_line(aes(color = Cluster), size = 1) +
  scale_color_manual(values = colors) +
  scale_fill_manual(values = colors) +
  coord_cartesian(ylim = c(0.95,5.05), expand = F) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5),
        legend.position = "none", strip.placement = "outside",
        axis.line = element_line(), axis.ticks = element_line()) +
  labs(y = "DTF (scaled 1-5)", x = NULL)
#
ggsave("Additional/Temp/Temp_F03_1.png", mp1, width = 8, height =   1 * 6 / 2.5)
ggsave("Additional/Temp/Temp_F03_2.png", mp2, width = 8, height = 1.5 * 6 / 2.5)
# Plot C)
xx <- ldp %>% left_join(select(x1, Entry, Cluster), by = "Entry") %>%
  mutate(test1 = factor(paste(Origin, Cluster)))
xx <- xx %>% filter(Origin != "ICARDA") %>%
  group_by(Origin, Cluster) %>% summarise(Count = n()) %>%
  spread(Cluster, Count) %>%
  left_join(select(ct, Origin=Country, Lat, Lon), by = "Origin") %>%
  ungroup() %>% as.data.frame()
xx[is.na(xx)] <- 0
invisible(png("Additional/Temp/Temp_F03_3.png", width = 2400, height = 1100, res = 150)) # 1200, 550
par(mai = c(0,0,0,0), xaxs = "i", yaxs = "i")
mapPies(dF = xx, nameX = "Lon", nameY = "Lat", zColours = colors,
        nameZs = c("1","2","3","4","5","6","7","8"), symbolSize = 1, lwd = 2,
        xlim = c(-140,110), ylim = c(0,20), addCatLegend = F,
        oceanCol = "grey90", landCol = "white", borderCol = "black")
```

```
symbolMaxSize= 5  maxSumValues= 49  symbolScale= 0.7142857
List of 2
 $ x: num [1:100] -125 -125 -125 -125 -125 ...
 $ y: num [1:100] 57.3 57.6 57.9 58.2 58.5 ...
```

```r
legend(-139.5, 15.5, title = "Cluster", legend = 1:8, col = colors,
       pch = 16, cex = 2, pt.cex = 4, box.lwd = 2)
invisible(dev.off())
#
im1 <- image_read("Additional/Temp/Temp_F03_1.png") %>%
  image_annotate("A)", size = 60, location = "+0+0")
im2 <- image_read("Additional/Temp/Temp_F03_2.png") %>%
  image_annotate("B)", size = 60, location = "+0+0")
im3 <- image_read("Additional/Temp/Temp_F03_3.png") %>%
  image_annotate("C)", size = 60, location = "+0+0")
im <- image_append(c(im1, im2, im3), stack = T)
image_write(im, "Figure_03_PCA.png")
```

```
#
summary(x1$Cluster)
```

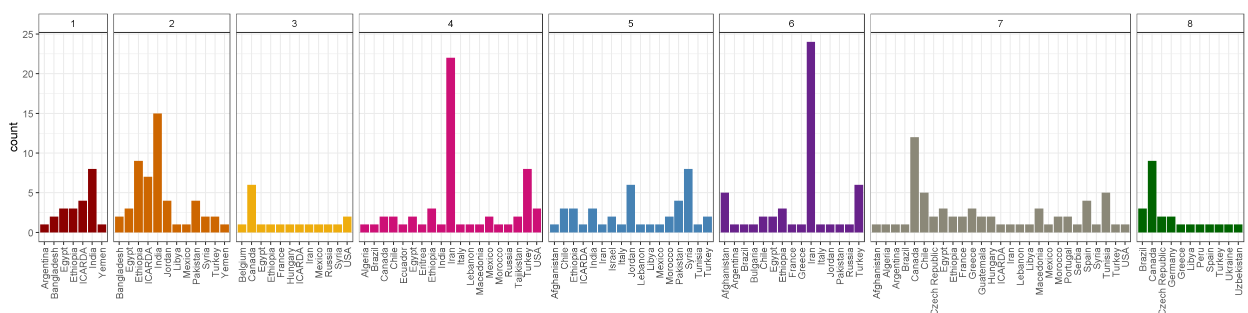
```
 1  2  3  4  5  6  7  8
22 51 18 56 41 51 62 23
```



```r
x1 <- read.csv("data/data_PCA_Results.csv") %>% mutate(Cluster = factor(Cluster))
```

Additional Figure 5 Cluster Origins

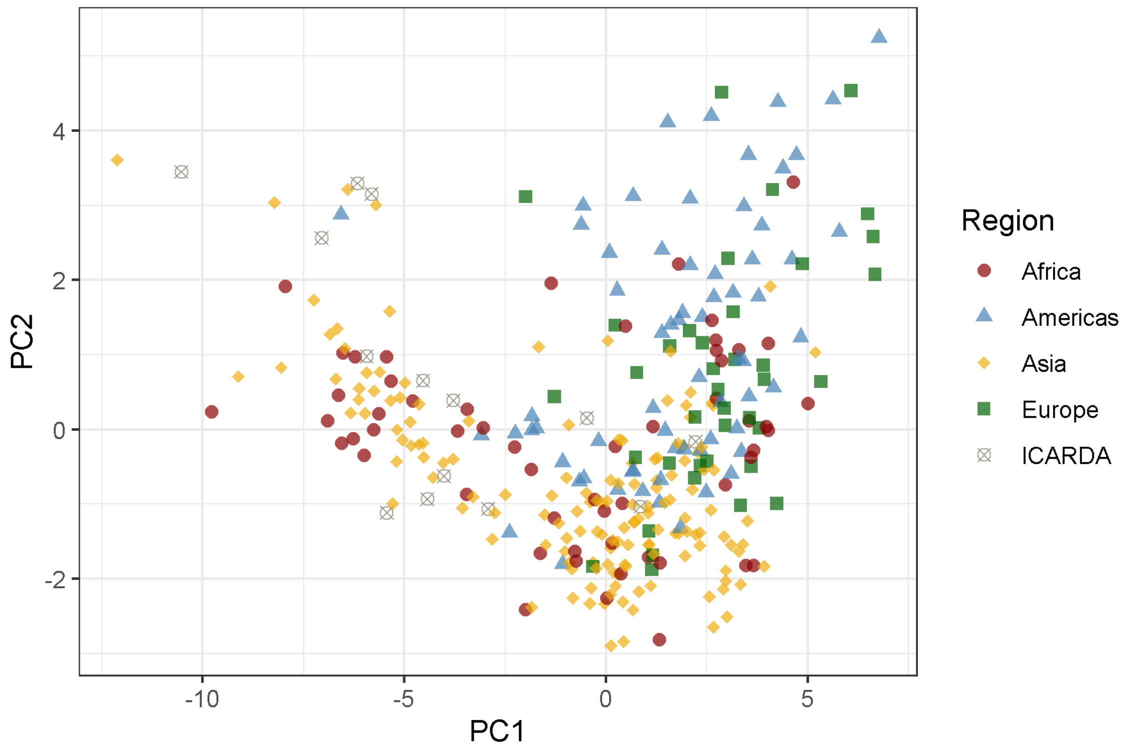
```r
# Prep data
pca <- read.csv("data/data_PCA_Results.csv") %>% mutate(Cluster = factor(Cluster))
xx <- ldp %>% left_join(select(pca, Entry, Cluster), by = "Entry") %>%
  mutate(test1 = factor(paste(Origin, Cluster)))
x1 <- xx %>% filter(Origin != "ICARDA") %>%
  group_by(Origin, Cluster) %>% summarise(Count = n()) %>%
  spread(Cluster, Count) %>%
  left_join(select(ct, Origin=Country, Lat, Lon), by = "Origin") %>%
  ungroup() %>% as.data.frame()
x1[is.na(x1)] <- 0
# Plot A) Bars
mp <- ggplot(xx, aes(x = Origin, fill = Cluster)) +
  geom_bar(stat = "count") +
  facet_grid(. ~ Cluster, scales = "free", space = "free") +
  scale_fill_manual(values = colors) +
  theme_AGL +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5),
        legend.position = "none") +
  labs(x = NULL)
ggsave("Additional/Additional_Figure_05_ClusterOrigins.png", width = 16, height = 4)
```



Additional Figure 6 PCA

```r
# Prep data
xx <- read.csv("data/data_PCA_Results.csv") %>%
  mutate(Region = ifelse(Origin == "ICARDA", "ICARDA", as.character(Region)))
# Plot PCA
mp <- ggplot(xx, aes(x = PC1, y = PC2, color = Region, shape = Region)) +
```

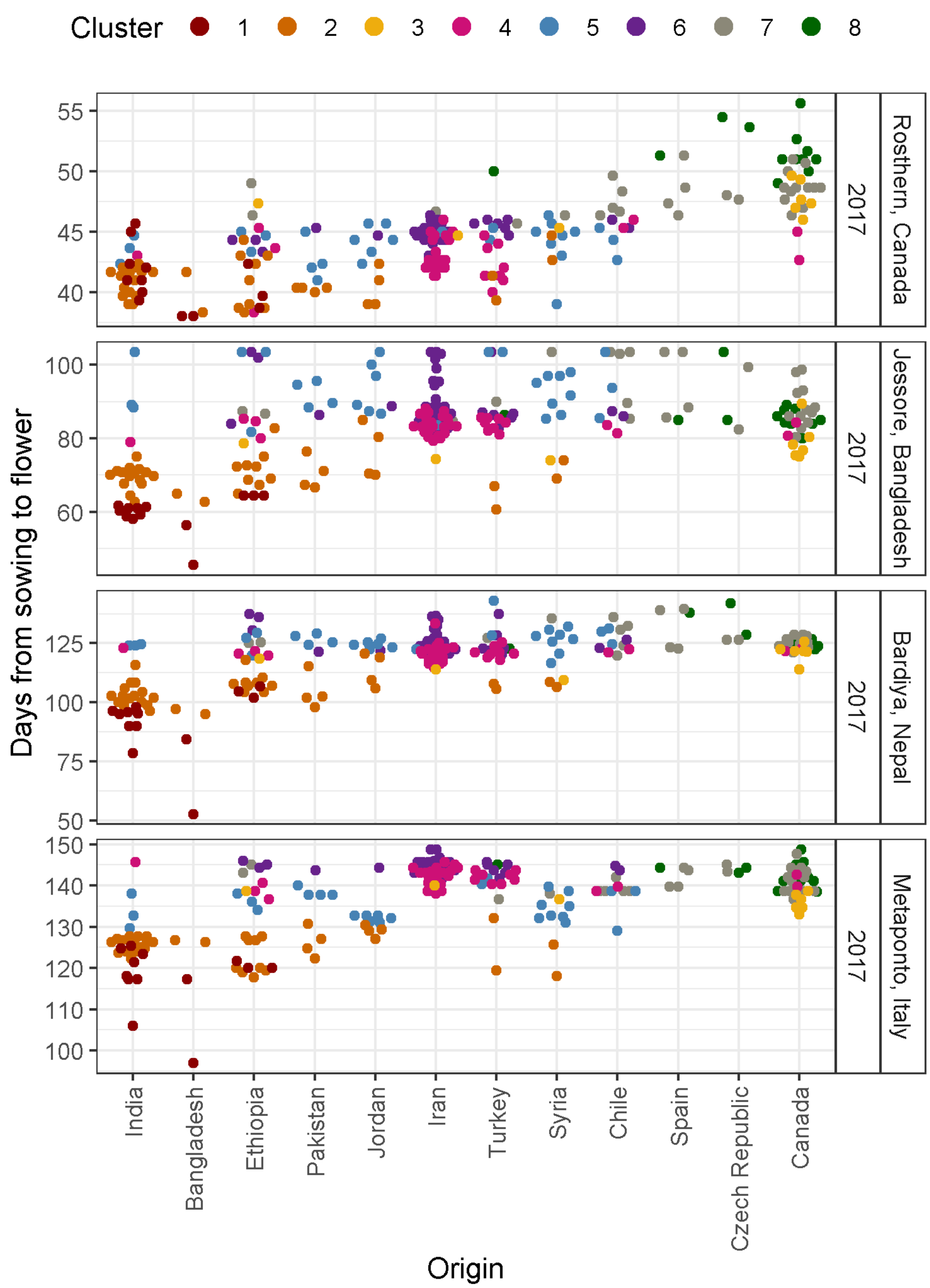
```
  geom_point(alpha = 0.7, size = 2) +
  scale_color_manual(values = colors[c(1,5,3,8,7)]) +
  scale_shape_manual(values = c(16,17,18,15,13)) +
  theme_AGL
ggsave("Additional/Additional_Figure_06_PCA.png", mp, width = 6, height = 4)
```



Additional Figure 7 DTF By Cluster

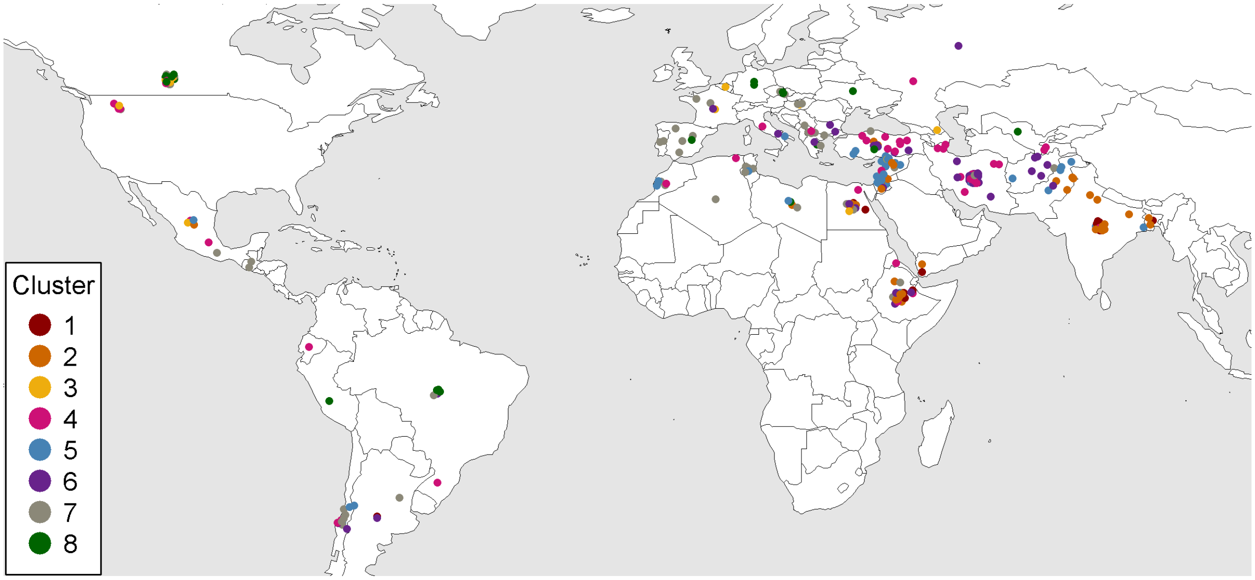
```
# Prep data
x1 <- read.csv("data/data_PCA_Results.csv") %>% mutate(Cluster = factor(Cluster))
yy <- c("India", "Bangladesh", "Ethiopia", "Pakistan", "Jordan",
        "Iran", "Turkey", "Syria", "Chile", "Spain", "Czech Republic", "Canada" )
xx <- dd %>% left_join(ldp, by = "Entry") %>%
  filter(ExptShort %in% c("Ro17", "Ba17", "Ne17", "It17"), Origin != "Unknown") %>%
  left_join(select(x1, Entry, Cluster), by = "Entry") %>%
  mutate(Origin = factor(Origin, levels = unique(Origin)[rev(order(unique(Origin)))])) %>%
  filter(Origin %in% yy) %>%
  mutate(Origin = factor(Origin, levels = yy))
# Plot
mp <- ggplot(xx, aes(y = DTF2, x = Origin)) +
  geom_quasirandom(aes(color = Cluster)) +
  facet_grid(Location+Year ~ ., scales = "free_y") +
  scale_color_manual(values = colors) +
```

```
  theme_AGL +
  theme(legend.position = "top",
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  guides(colour = guide_legend(nrow = 1, override.aes = list(size = 3))) +
  labs(y = "Days from sowing to flower")
ggsave("Additional/Additional_Figure_07_DTFByCluster.png", mp, width = 5, height = 7)
```



Additional Figure 8 LDP Origins By Cluster

```
# Prep data
x1 <- read.csv("data/data_PCA_Results.csv") %>% mutate(Cluster = factor(Cluster))
xx <- ldp %>% select(Entry, Name, Lat, Lon) %>% left_join(x1, by = "Entry") %>%
  left_join(select(ct, Origin=Country, cLat=Lat, cLon=Lon), by = "Origin") %>%
  mutate(Lat = ifelse(is.na(Lat), cLat, Lat),
         Lon = ifelse(is.na(Lon), cLon, Lon),
         Lat = ifelse(duplicated(Lat), jitter(Lat, 1, 1), Lat),
         Lon = ifelse(duplicated(Lon), jitter(Lon, 1, 1), Lon), Size = 1)
# Plot Map
invisible(png("Additional/Additional_Figure_08_LDPOriginsByCluster.png", width = 2400, height = 1100, r
par(mai = c(0,0,0,0), xaxs = "i", yaxs = "i")
mapBubbles(dF = xx, nameX = "Lon", nameY = "Lat", nameZColour = "Cluster",
           nameZSize = "Size", symbolSize = 0.5, pch = 20, colourPalette = colors[1:8],
           xlim = c(-140,110), ylim = c(0,20), addLegend = F, addColourLegend = F, colourLegendTitle = 
           oceanCol = "grey90", landCol = "white", borderCol = "black")
legend(-139.5, 15.5, title = "Cluster", legend = 1:8, col = colors,
       pch = 16, cex = 2, pt.cex = 4, box.lwd = 2)
invisible(dev.off())
```



Additional Figures - Entry Regressions

```
myfills <- alpha(c("darkgreen", "darkred", "darkblue"), 0.5)
mymin <- min(rr$RDTF, na.rm = T); mymax <- max(rr$RDTF, na.rm = T)
mp <- list()
for(i in 1:324) {
  xx <- rr %>% filter(Entry == i) %>%
```

```r
    left_join(select(ff, Expt, MacroEnv, T_mean, P_mean), by = "Expt") %>%
    mutate(myfill = MacroEnv)
  x1 <- xx %>% filter(MacroEnv != "South Asia")
  x2 <- xx %>% filter(MacroEnv != "Temperate")
  x3 <- xx %>% filter(MacroEnv != "Mediterranean")
  figlab <- paste("Entry", str_pad(i, 3, "left", "0"), "|", unique(xx$Name))
  # Plot A) 1/f = a + bT
  mp1 <- ggplot(xx, aes(x = T_mean, y = RDTF)) +
    geom_point(aes(shape = MacroEnv, color = MacroEnv)) +
    geom_smooth(data = x1, method = "lm", se = F, color = "black", lty = 3) +
    geom_smooth(data = x2, method = "lm", se = F, color = "black") +
    scale_y_continuous(sec.axis = dup_axis(~ 1/., name = NULL, breaks = c(35,50,100,150)),
                       trans = "reverse", breaks = c(0.01,0.02,0.03), limits = c(mymax, mymin)) +
    scale_x_continuous(breaks = c(11,13,15,17,19,21)) +
    scale_shape_manual(name = "Macro-environment:", values = c(16,15,17)) +
    scale_color_manual(name = "Macro-environment:", values = myfills) +
    theme_AGL +
    labs(title = figlab, y = "1 / DTF",
         x = expression(paste("Temperature (", degree, "C)", sep = "")))
  # Plot B) 1/f = a + cP
  mp2 <- ggplot(xx, aes(x = P_mean, y = RDTF)) +
    geom_point(aes(shape = MacroEnv, color = MacroEnv)) +
    geom_smooth(data = x1, method = "lm", se = F, color = "black", lty = 3) +
    geom_smooth(data = x3, method = "lm", se = F, color = "black") +
    scale_y_continuous(sec.axis = dup_axis(~ 1/., name="DTF", breaks = c(35,50,100,150)),
                       trans = "reverse", breaks = c(0.01,0.02,0.03), limits = c(mymax, mymin)) +
    scale_x_continuous(breaks = c(11,12,13,14,15,16)) +
    scale_shape_manual(name = "Macro-environment:", values = c(16,15,17)) +
    scale_color_manual(name = "Macro-environment:", values = myfills) +
    theme_AGL +
    labs(title = "", y = NULL,x = "Photoperiod (hours)")
  #
  mp[[i]] <- ggarrange(mp1, mp2, ncol = 2, common.legend = T, legend = "bottom")
  ggsave(paste0("Additional/Entry_TP/Entry_", str_pad(i, 3, pad = "0"), ".png"),
         mp[[i]], width = 8, height = 4)
}
pdf("Additional/pdf_TP.pdf", width = 8, height = 4)
for(i in 1:324) { print(mp[[i]]) }
dev.off() #dev.set(dev.next())
```

PhotoThermal Plane

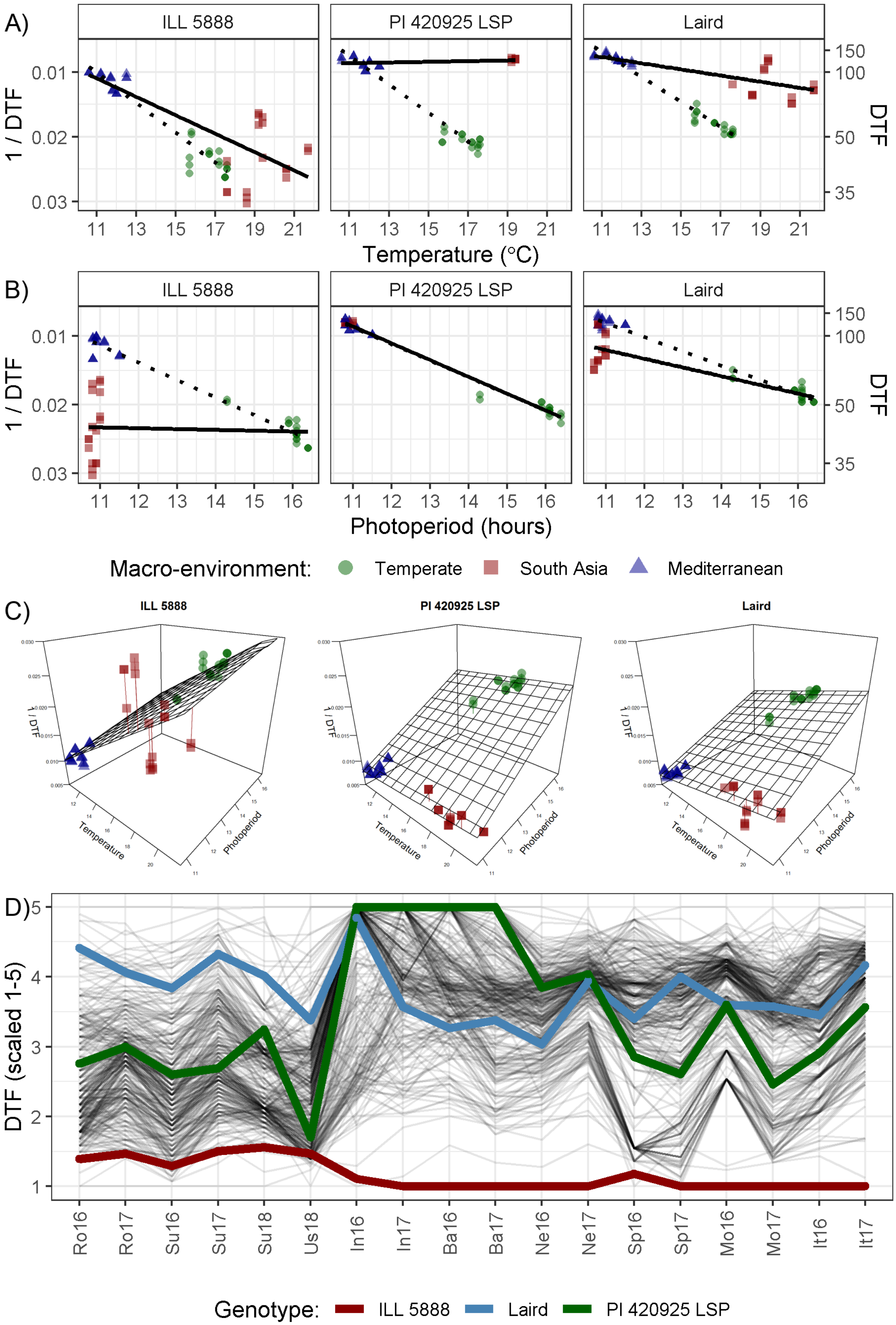
```r
# Prep data
xx <- rr %>% filter(!is.na(RDTF)) %>%
  left_join(select(ff, Expt, T_mean, P_mean, MacroEnv), by = "Expt")
# Create plotting function
gg_PTplane <- function(x, i) {
  x1 <- x %>% filter(Entry == i) %>%
    arrange(MacroEnv) %>%
```

```r
    mutate(myPal = as.character(plyr::mapvalues(MacroEnv,
             c("Temperate", "South Asia", "Mediterranean"),
             c("darkgreen", "darkred", "darkblue") ) ) )
  x <- x1$T_mean
  y <- x1$P_mean
  z <- x1$RDTF
  fit <- lm(z ~ x + y)
  # Create PhotoThermal plane
  fitp <- predict(fit)
  grid.lines <- 12
  x.p <- seq(min(x), max(x), length.out = grid.lines)
  y.p <- seq(min(y), max(y), length.out = grid.lines)
  xy <- expand.grid(x = x.p, y = y.p)
  z.p <- matrix(predict(fit, newdata = xy), nrow = grid.lines, ncol = grid.lines)
  pchs <- plyr::mapvalues(x1$Expt, names_Expt, c(rep(16,6),rep(15,6),rep(17,6))) %>%
    as.character() %>% as.numeric()
  # Plot with regression plane
  par(mar=c(1.5, 2.5, 1.5, 0.5))
  scatter3D(x, y, z, pch = pchs, cex = 2, zlim = c(0.005,0.03), main = unique(x1$Name),
    col = alpha(x1$myPal,0.5), colvar = as.numeric(x1$MacroEnv), colkey = F,
    theta = 40, phi = 25, ticktype = "detailed", cex.lab = 1, cex.axis = 0.5,
    xlab = "Temperature", ylab = "Photoperiod", zlab = "1 / DTF",
    surf = list(x = x.p, y = y.p, z = z.p, col = "black", facets = NA, fit = fitp) )
}
# Plot each Entry
for (i in 1:324) {
  png(paste0("Additional/Entry_3D/3D_Entry_", str_pad(i, 3, pad = "0"), ".png"),
      width = 1000, height = 1000, res = 200)
  gg_PTplane(xx, i)
  dev.off()
}
# Create PDF
pdf("Additional/pdf_3D.pdf")
par(mar=c(1.5, 2.5, 1.5, 0.5))
for (i in 1:324) {
  gg_PTplane(xx, i)
}
dev.off() #dev.set(dev.next())
# Create animation
lf <- list.files("Additional/Entry_3D")
mp <- image_read(paste0("Additional/Entry_3D/", lf))
animation <- image_animate(mp, fps = 4)
image_write(animation, "Additional/Animation_3D.gif")
# Plot ILL 5888 & ILL 4400 & Laird
xx <- xx %>% mutate(Name = gsub(" AGL", "", Name))
for (i in c(235, 94, 128)) {
  png(paste0("Additional/Temp/3D_Entry_", str_pad(i, 3, pad = "0"), ".png"),
      width = 1000, height = 1000, res = 200)
  gg_PTplane(xx, i)
  dev.off()
}
```

Supplemental Figure 4 Regressions

```
# Prep data
myfills <- alpha(c("darkgreen", "darkred", "darkblue"), 0.5)
yy <- c("ILL 5888 AGL", "PI 420925 LSP AGL",  "Laird AGL") # "ILL 4400 AGL",
xx <- rr %>% filter(Name %in% yy, !is.na(DTF)) %>%
  left_join(select(ff, Expt, MacroEnv, T_mean, P_mean), by = "Expt") %>%
  mutate(Name = gsub(" AGL", "", Name),
         Name = factor(Name, levels = gsub(" AGL", "", yy)),
         myfill = MacroEnv)
x1 <- xx %>% filter(MacroEnv != "South Asia")
x2 <- xx %>% filter(MacroEnv != "Temperate")
x3 <- xx %>% filter(MacroEnv != "Mediterranean")
# Plot A) 1/f = a + bT
mp1 <- ggplot(xx, aes(x = T_mean, y = RDTF)) +
  geom_point(aes(shape = MacroEnv, color = MacroEnv)) +
  geom_smooth(data = x1, method = "lm", se = F, color = "black", lty = 3) +
  geom_smooth(data = x2, method = "lm", se = F, color = "black") +
  scale_y_continuous(trans = "reverse", breaks = c(0.01,0.02,0.03),
        sec.axis = dup_axis(~ 1/., name = "DTF", breaks = c(35,50,100,150))) +
  scale_x_continuous(breaks = c(11,13,15,17,19,21)) +
  scale_shape_manual(name = "Macro-environment:", values = c(16,15,17)) +
  scale_color_manual(name = "Macro-environment:", values = myfills) +
  theme_AGL + facet_grid(. ~ Name) +
  theme(plot.margin = unit(c(0,0,0,0), "cm")) +
  guides(colour = guide_legend(override.aes = list(size = 3))) +
  labs(y = "1 / DTF", x = expression(paste("Temperature (", degree, "C)", sep = "")))
# Plot B) 1/f = a + cP
mp2 <- ggplot(xx, aes(x = P_mean, y = RDTF)) +
  geom_point(aes(shape = MacroEnv, color = MacroEnv)) +
  geom_smooth(data = x1, method = "lm", se = F, color = "black", lty = 3) +
  geom_smooth(data = x3, method = "lm", se = F, color = "black") +
  scale_y_continuous(trans = "reverse", breaks = c(0.01,0.02,0.03),
        sec.axis = dup_axis(~ 1/., name="DTF", breaks = c(35,50,100,150))) +
  scale_x_continuous(breaks = c(11,12,13,14,15,16)) +
  scale_shape_manual(name = "Macro-environment:", values = c(16,15,17)) +
  scale_color_manual(name = "Macro-environment:", values = myfills) +
  theme_AGL + facet_grid(. ~ Name) +
  theme(plot.margin = unit(c(0,0,0,0), "cm")) +
  guides(colour = guide_legend(override.aes = list(size = 3))) +
  labs(y = "1 / DTF",x = "Photoperiod (hours)")
# Append A) and B)
mp <- ggarrange(mp1, mp2, ncol = 1, common.legend = T, legend = "bottom")
ggsave("Additional/Temp/Temp_SF04_1.png", mp, width = 6, height = 4)
# Append C)s
im1 <- image_read("Additional/Temp/3D_Entry_094.png")
im2 <- image_read("Additional/Temp/3D_Entry_235.png") #3D_Entry_076.png
im3 <- image_read("Additional/Temp/3D_Entry_128.png")
im <- image_append(c(im1, im2, im3)) %>% image_scale("1800x")
image_write(im, "Additional/Temp/Temp_SF04_2.png")
# D)
xx <- dd %>% filter(Name %in% yy) %>%
  mutate(Name = factor(Name, levels = yy),
```

```
         Name = gsub(" AGL", "", Name))
mp3 <- ggplot(dd, aes(x = ExptShort, y = DTF2_scaled, group = Name)) +
  geom_line(color = "black", alpha = 0.1) +
  geom_line(data = xx, aes(color = Name), size = 2) +
  geom_point(data = xx, aes(color = Name)) +
  scale_color_manual(name = "Genotype:", values = colors[c(1,5,8)]) +
  theme_AGL + labs(y = "DTF (scaled 1-5)", x = NULL) +
  theme(legend.position = "bottom",
        plot.margin = unit(c(0,0,0,0), "cm"),
        axis.text.x = element_text(angle = 90, vjust = 0.5))
ggsave("Additional/Temp/Temp_SF04_3.png", mp3, width = 6, height = 3)
# Append A), B), C) and D)
im1 <- image_read("Additional/Temp/Temp_SF04_1.png") %>%
  image_annotate("A)", size = 50, location = "+0+10") %>%
  image_annotate("B)", size = 50, location = "+0+550")
im2 <- image_read("Additional/Temp/Temp_SF04_2.png") %>%
  image_annotate("C)", size = 50)
im3 <- image_read("Additional/Temp/Temp_SF04_3.png") %>%
  image_annotate("D)", size = 50)
im <- image_append(c(im1, im2, im3), stack = T)
image_write(im, "Supplemental_Figure_04_Regressions.png")
```



A)
ILL 5888
PI 420925 LSP
Laird
1 / DTF
DTF
Temperature (°C)
B)
Photoperiod (hours)
Macro-environment:
Temperate
South Asia
Mediterranean
C)
Temperature
Photoperiod
1 / DTF
D)
DTF (scaled 1-5)
Ro16
Ro17
Su16
Su17
Su18
Us18
In16
In17
Ba16
Ba17
Ne16
Ne17
Sp16
Sp17
Mo16
Mo17
It16
It17
Genotype:
ILL 5888
Laird
PI 420925 LSP

```r
e1 <- ee %>% filter(ExptShort == "It17")
nrow(e1)
```

```
[1] 181
```

```r
nrow(e1%>%filter(Temp_mean < 15))
```

```
[1] 126
```

Modeling DTF - functions

```r
# Create functions
# Plot Observed vs Predicted
gg_model_1 <- function(x, title = NULL, type = 1,
    mymin = min(c(x$DTF,x$Predicted_DTF)) - 2,
    mymax = max(c(x$DTF,x$Predicted_DTF)) + 2 ) {
  x <- x %>% mutate(Flowered = ifelse(is.na(DTF), "Did not Flower", "Flowered"))
  # Prep data
  if(type == 1) {
    myx <- "DTF"
    myy <- "Predicted_DTF"
    x <- x %>% filter(!is.na(DTF))
  }
  if(type == 2) {
    myx <- "RDTF"
    myy <- "Predicted_RDTF"
    x <- x %>% filter(!is.na(RDTF))
  }
  myPal <- colors_Expt[names_Expt %in% unique(x$Expt)]
  r2 <- round(modelR2(x = x[,myx], y = x[,myy]), 3)
  # Plot
  mp <- ggplot(x) +
    geom_point(aes(x = get(myx), y = get(myy), color = Expt)) +
    geom_abline() +
    geom_label(x = mymin, y = mymax, hjust = 0, vjust = 1, parse = T,
               label = paste("italic(R)^2 == ", r2)) +
    scale_x_continuous(limits = c(mymin, mymax), expand = c(0, 0)) +
    scale_y_continuous(limits = c(mymin, mymax), expand = c(0, 0)) +
    scale_color_manual(name = NULL, values = myPal) +
    theme_AGL + guides(colour = guide_legend(override.aes = list(size = 2))) +
    labs(y = "Predicted", x = "Observed")
  if(!is.null(title)) { mp <- mp + labs(title = title) }
  mp
}
# Facets by Expt
gg_model_2 <- function(x, myX = "DTF", myY = "Predicted_DTF", title = NULL,
                       x1 = 30, x2 = 30, y1 = 145, y2 = 120, legend.pos = "bottom") {
```

```r
  # Prep data
  pca <- read.csv("data/data_PCA_Results.csv") %>% select(Entry, Cluster) %>%
    mutate(Cluster = factor(Cluster))
  x <- x %>%
    filter(!is.na(get(myX))) %>%
    left_join(pca, by = "Entry")
  xf <- x %>% group_by(Expt) %>%
    summarise(Mean = mean(DTF)) %>% ungroup() %>%
    mutate(r2 = NA, RMSE = NA)
  for(i in 1:nrow(xf)) {
    xi <- x %>% filter(Expt == xf$Expt[i])
    xf[i,"r2"]   <- round(modelR2(x = xi[,myX],   y = xi[,myY]), 2)
    xf[i,"RMSE"] <- round(modelRMSE(x = xi[,myX], y = xi[,myY]), 1)
  }
  # Plot
  mp <- ggplot(x, aes(x = get(myX), y = get(myY))) +
    geom_point(aes(color = Cluster), size = 0.75, alpha = 0.7) + geom_abline() +
    geom_text(x = x1, y = y1, color = "black", hjust = 0, vjust = 0, size = 3,
              aes(label = paste("RMSE = ", RMSE, sep = "")), data = xf) +
    geom_text(x = x2, y = y2, color = "black", hjust = 0, vjust = 0, size = 3,
              aes(label = paste("italic(R)^2 == ", r2)), parse = T, data = xf) +
    facet_wrap(Expt ~ ., ncol = 6, labeller = label_wrap_gen(width = 17)) +
    scale_x_continuous(limits = c(min(x[,myX]), max(x[,myX]))) +
    scale_y_continuous(limits = c(min(x[,myX]), max(x[,myX]))) +
    scale_color_manual(values = colors) +
    theme_AGL +
    theme(legend.position = legend.pos,
          legend.margin     = unit(c(0,0,0,0), "cm")) +
    guides(colour = guide_legend(nrow = 1, override.aes = list(size = 3))) +
    labs(y = "Predicted", x = "Observed")
  if(!is.null(title)) { mp <- mp + labs(title = title) }
  mp
}
# R^2 function
modelR2 <- function(x, y) {
  1 - ( sum((x - y)^2, na.rm = T) / sum((x - mean(x, na.rm = T))^2, na.rm = T))
}
# RMSE function
modelRMSE <- function(x, y) {
  sqrt(sum((x-y)^2) / length(x))
}
```

$R^2 = 1 - \frac{SS_{residuals}}{SS_{total}} = 1 - \frac{\sum(x-y)^2}{\sum(x-\bar{x})}$

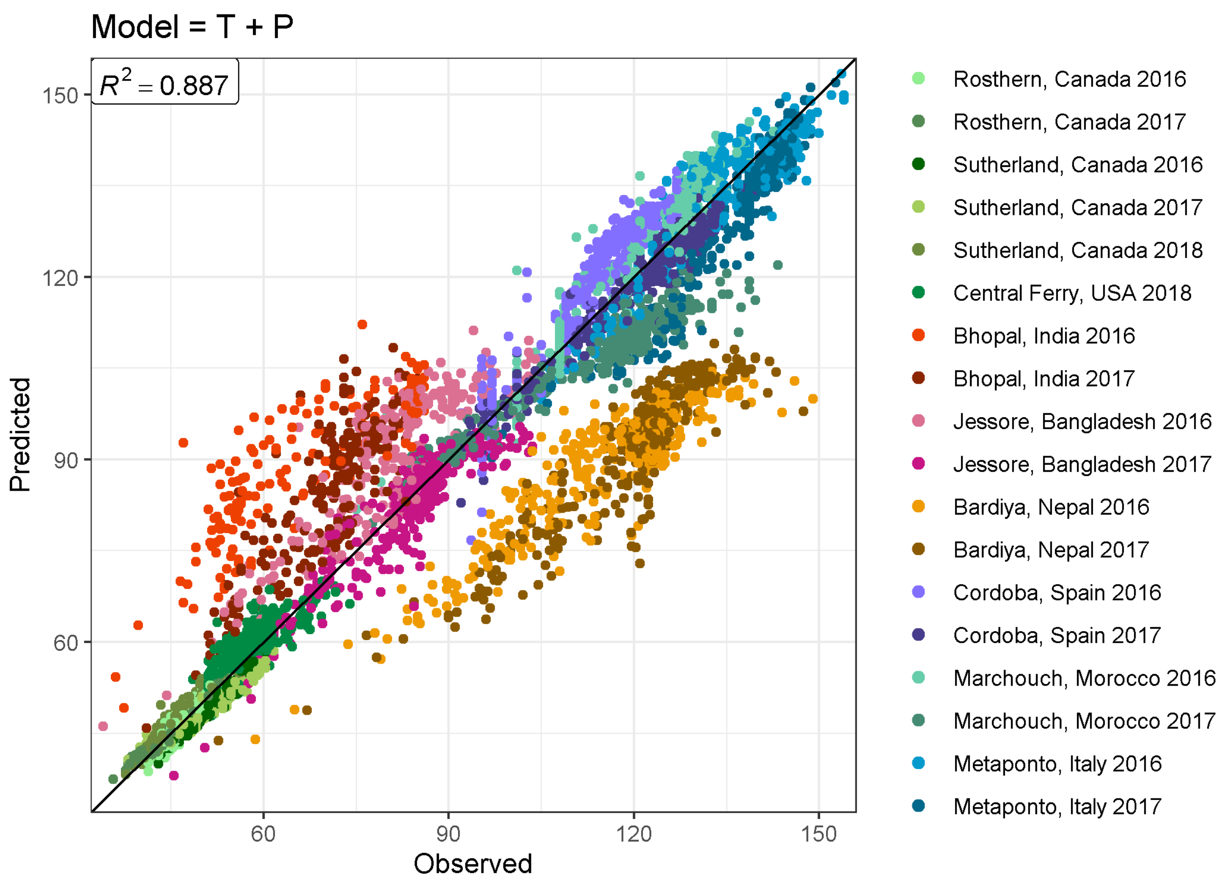
$RMSE = \frac{\sum(y-x)^2}{n}$

Modeling DTF (T + P) - All Site-years

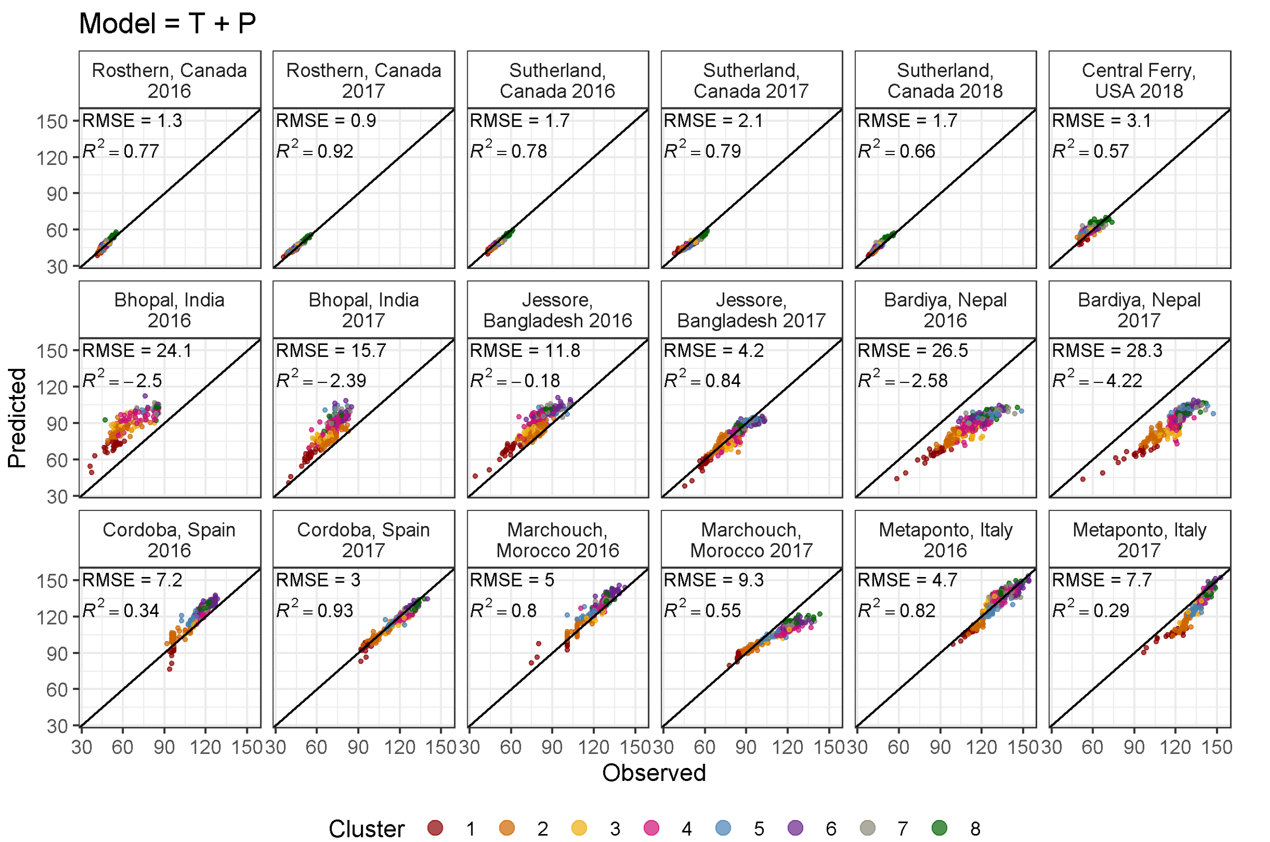
```
############################
# 1/f = a + bT + cP (All) #
############################
# Prep data
xx <- rr %>% filter(!is.na(RDTF)) %>%
  left_join(select(ff, Expt, T_mean, P_mean), by = "Expt") %>%
  select(Plot, Entry, Name, Rep, Expt, ExptShort, T_mean, P_mean, RDTF, DTF)
mr <- NULL; md <- NULL
mc <- select(ldp, Entry, Name) %>%
  mutate(a = NA, b = NA, c = NA, RR = NA, Environments = NA,
         aP = NA, bP = NA, cP = NA)
# Model
for(i in 1:324) {
  # Prep data
  xri <- xx %>% filter(Entry == i)
  xdi <- xri %>% group_by(Entry, Name, Expt, ExptShort) %>%
    summarise_at(vars(DTF, RDTF, T_mean, P_mean), funs(mean), na.rm = T) %>%
    ungroup()
  # Train Model
  mi <- lm(RDTF ~ T_mean + P_mean, data = xri)
  # Predict DTF
  xri <- xri %>% mutate(Predicted_RDTF = predict(mi),
                        Predicted_DTF = 1 / predict(mi))
  xdi <- xdi %>% mutate(Predicted_RDTF = predict(mi, newdata = xdi),
                        Predicted_DTF = 1 / predict(mi, newdata = xdi))
  # Save to table
  mr <- bind_rows(mr, xri)
  md <- bind_rows(md, xdi)
  # Save coefficients
  mc[i,c("a","b","c")] <- mi$coefficients
  # Calculate rr and # of environments used
  mc[i,"RR"] <- 1 - sum((xri$DTF - xri$Predicted_DTF)^2, na.rm = T) /
    sum((xri$Predicted_DTF - mean(xri$DTF, na.rm = T))^2, na.rm = T)
  mc[i,"Environments"] <- length(unique(xri$Expt[!is.na(xri$DTF)]))
  mc[i,"aP"] <- summary(mi)[[4]][1,4]
  mc[i,"bP"] <- summary(mi)[[4]][2,4]
  mc[i,"cP"] <- summary(mi)[[4]][3,4]
}
mr <- mr %>% mutate(Expt = factor(Expt, levels = names_Expt))
md <- md %>% mutate(Expt = factor(Expt, levels = names_Expt))
# Save Results
write.csv(mr, "data/model_T+P.csv",       row.names = F)
write.csv(md, "data/model_T+P_d.csv",     row.names = F)
write.csv(mc, "data/model_T+P_coefs.csv", row.names = F)
#
# Plot Each Entry
mp <- list()
for(i in 1:324) {
  mp1 <- gg_model_1(mr %>% filter(Entry == i), paste("Entry", i, "| DTF"))
  mp2 <- gg_model_1(mr %>% filter(Entry == i), paste("Entry", i, "| RDTF"), type = 2,
           mymin = min(c(mr$Predicted_RDTF, mr$RDTF)) - 0.001,
```

```r
          mymax = max(c(mr$Predicted_RDTF, mr$RDTF)) + 0.001)
  mp[[i]] <- ggarrange(mp1, mp2, ncol = 2, common.legend = T, legend = "right")
  fname <- paste0("Additional/Entry_Model/Model_Entry_", str_pad(i, 3, pad = "0"), ".png")
  ggsave(fname, mp[[i]], width = 10, height = 4.25)
}
pdf("Additional/pdf_Model.pdf", width = 10, height = 4.25)
for (i in 1:324) { print(mp[[i]]) }
dev.off() #dev.set(dev.next())
```

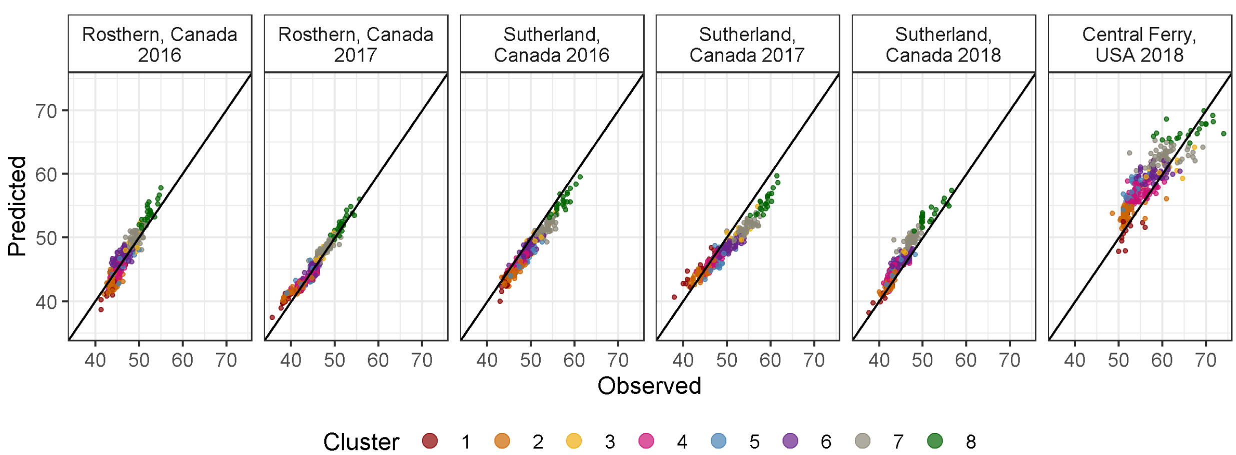
```r
# Prep data
xx <- read.csv("data/model_T+P_d.csv") %>% mutate(Expt = factor(Expt, levels = names_Expt))
# Plot Observed vs Predicted
mp <- gg_model_1(xx, title = "Model = T + P")
ggsave("Additional/Model/Model_1_1.png", mp, width = 7, height = 5)
```




```r
# Plot Observed vs Predicted
mp <- gg_model_2(xx, title = "Model = T + P")
ggsave("Additional/Model/Model_2_1.png", mp, width = 8, height = 5.5)
```



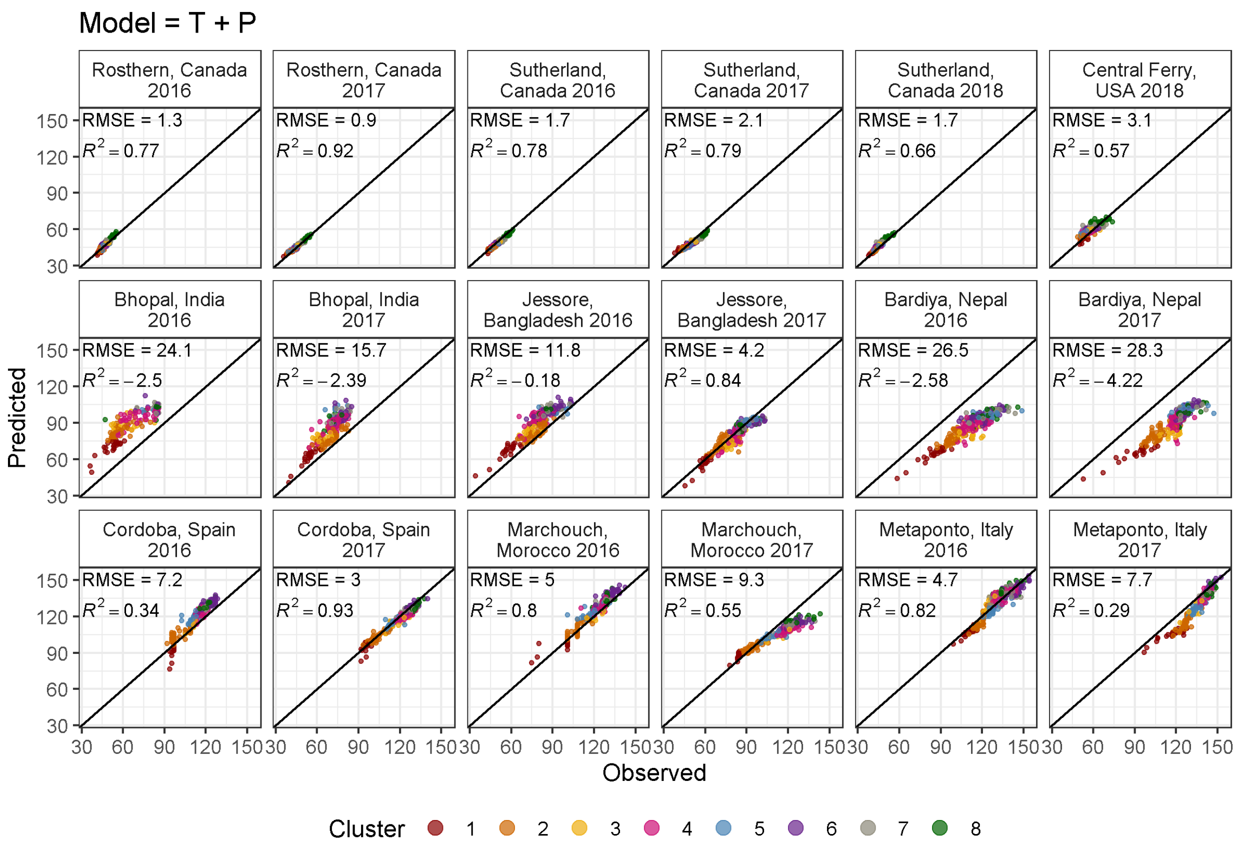

```r
mp <- gg_model_2(xx %>% filter(ExptShort %in% c("Ro16","Ro17","Su16","Su17","Su18","Us18")))
ggsave("Additional/Model/Model_3_1.png", mp ,width = 8, height = 3)
```



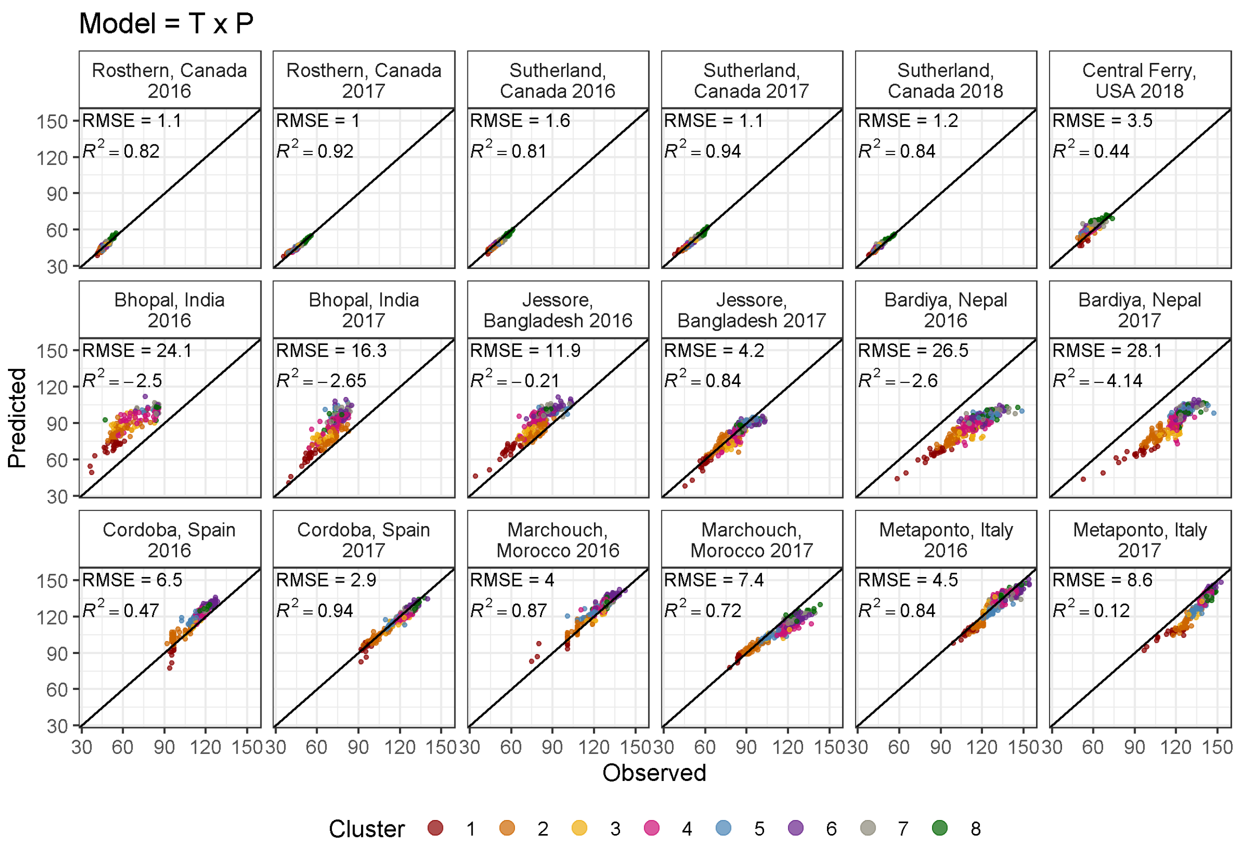
Modeling DTF (T x P) - All Site-years

```
###########################
# 1/f = a + bT + cP (All) #
###########################
# Prep data
xx <- rr %>% filter(!is.na(RDTF)) %>%
  left_join(select(ff, Expt, T_mean, P_mean), by = "Expt") %>%
  select(Plot, Entry, Name, Rep, Expt, ExptShort, T_mean, P_mean, RDTF, DTF)
mr <- NULL; md <- NULL
mc <- select(ldp, Entry, Name) %>%
  mutate(a = NA, b = NA, c = NA, d = NA, RR = NA, Environments = NA,
         aP = NA, bP = NA, cP= NA, dP = NA)
# Model
for(i in 1:324) {
  # Prep data
  xri <- xx %>% filter(Entry == i)
  xdi <- xri %>% group_by(Entry, Name, Expt, ExptShort) %>%
    summarise_at(vars(DTF, RDTF, T_mean, P_mean), funs(mean), na.rm = T) %>%
    ungroup()
  # Train Model
  mi <- lm(RDTF ~ T_mean * P_mean, data = xri)
  # Predict DTF
  xri <- xri %>% mutate(Predicted_RDTF = predict(mi),
                        Predicted_DTF = 1 / predict(mi))
  xdi <- xdi %>% mutate(Predicted_RDTF = predict(mi, newdata = xdi),
                        Predicted_DTF = 1 / predict(mi, newdata = xdi))
  # Save to table
  mr <- bind_rows(mr, xri)
  md <- bind_rows(md, xdi)
  # Save coefficients
  mc[i,c("a","b","c","d")] <- mi$coefficients
  # Calculate rr and # of environments used
  mc[i,"RR"] <- 1 - sum((xri$DTF - xri$Predicted_DTF)^2, na.rm = T) /
    sum((xri$Predicted_DTF - mean(xri$DTF, na.rm = T))^2, na.rm = T)
  mc[i,"Environments"] <- length(unique(xri$Expt[!is.na(xri$DTF)]))
  mc[i,"aP"] <- summary(mi)[[4]][1,4]
  mc[i,"bP"] <- summary(mi)[[4]][2,4]
  mc[i,"cP"] <- summary(mi)[[4]][3,4]
  mc[i,"dP"] <- summary(mi)[[4]][4,4]
}
mr <- mr %>% mutate(Expt = factor(Expt, levels = names_Expt))
md <- md %>% mutate(Expt = factor(Expt, levels = names_Expt))
# Save Results
write.csv(mr, "data/model_TxP.csv",       row.names = F)
write.csv(md, "data/model_TxP_d.csv",     row.names = F)
write.csv(mc, "data/model_TxP_coefs.csv", row.names = F)
```

```
# Prep data
xx <- read.csv("data/model_TxP_d.csv") %>% mutate(Expt = factor(Expt, levels = names_Expt))
# Plot Observed vs Predicted
mp <- gg_model_1(xx, title = "Model = T x P")
ggsave("Additional/Model/Model_1_2.png", mp, width = 7, height = 5)
```




```r
# Plot Observed vs Predicted
mp <- gg_model_2(xx, title = "Model = T x P")
ggsave("Additional/Model/Model_2_2.png", mp, width = 8, height = 5.5)
```



Supplemental Table 3 Model Constants

```
# Supplemental Table 03
x1 <- read.csv("data/model_T+P_coefs.csv")
x2 <- read.csv("data/model_TxP_coefs.csv")
xx <- bind_rows(x1, x2) %>% arrange(Entry) %>%
  select(Entry, Name, a, b, c, d, RR, Environments, aP, bP, cP, dP)
write.csv(xx, "Supplemental_Table_03.csv", na = "", row.names = F)
```

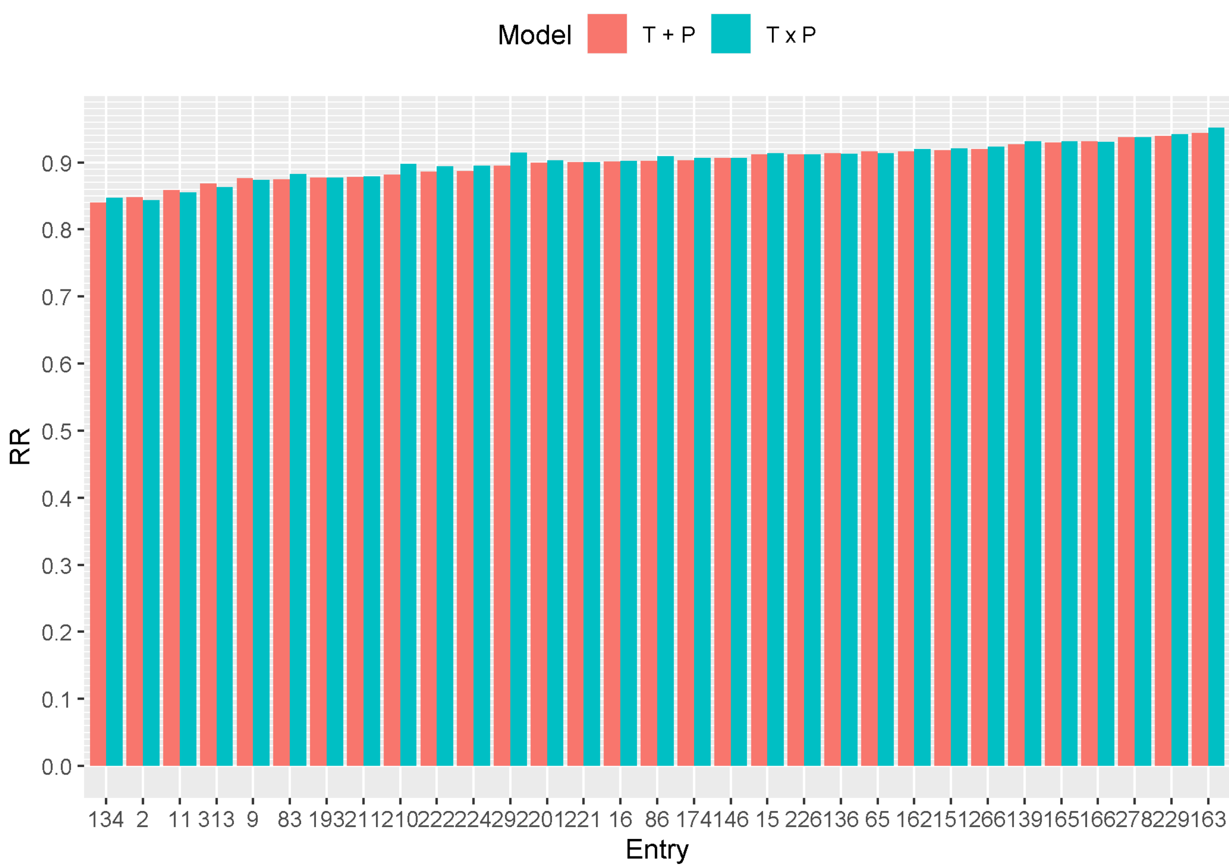
Additional Figure 9 significant T x P interactions

```
x1 <- read.csv("data/model_T+P_coefs.csv") %>% mutate(Model = "T + P")
x2 <- read.csv("data/model_TxP_coefs.csv") %>% mutate(Model = "T x P")
xx <- bind_rows(x1, x2) %>% arrange(Entry) %>%
  left_join(select(ldp, Entry, Origin), by = "Entry") %>%
  select(Entry, Name, Model, a, b, c, d, RR, Environments, aP, bP, cP, dP)
```

```r
ents <- x2 %>% filter(dP < 0.05) %>% pull(Entry)
xx <- xx %>% filter(Entry %in% ents)
write.csv(xx, "Additional/Additional_Table_01.csv", na = "", row.names = F)
xx <- xx %>% arrange(RR) %>% mutate(Entry = factor(Entry, levels = unique(Entry)))
length(ents)
```

```
[1] 31
```

```r
mp <- ggplot(xx, aes(x = Entry, y = RR, fill = Model)) +
  geom_bar(stat = "identity", position = "dodge") +
  scale_y_continuous(breaks = seq(0,1,0.1), minor_breaks = seq(0,1,0.01)) +
  theme(legend.position = "top")
ggsave("Additional/Additional_Figure_09_TxPRR.png" ,mp, width = 7, height = 5)
```



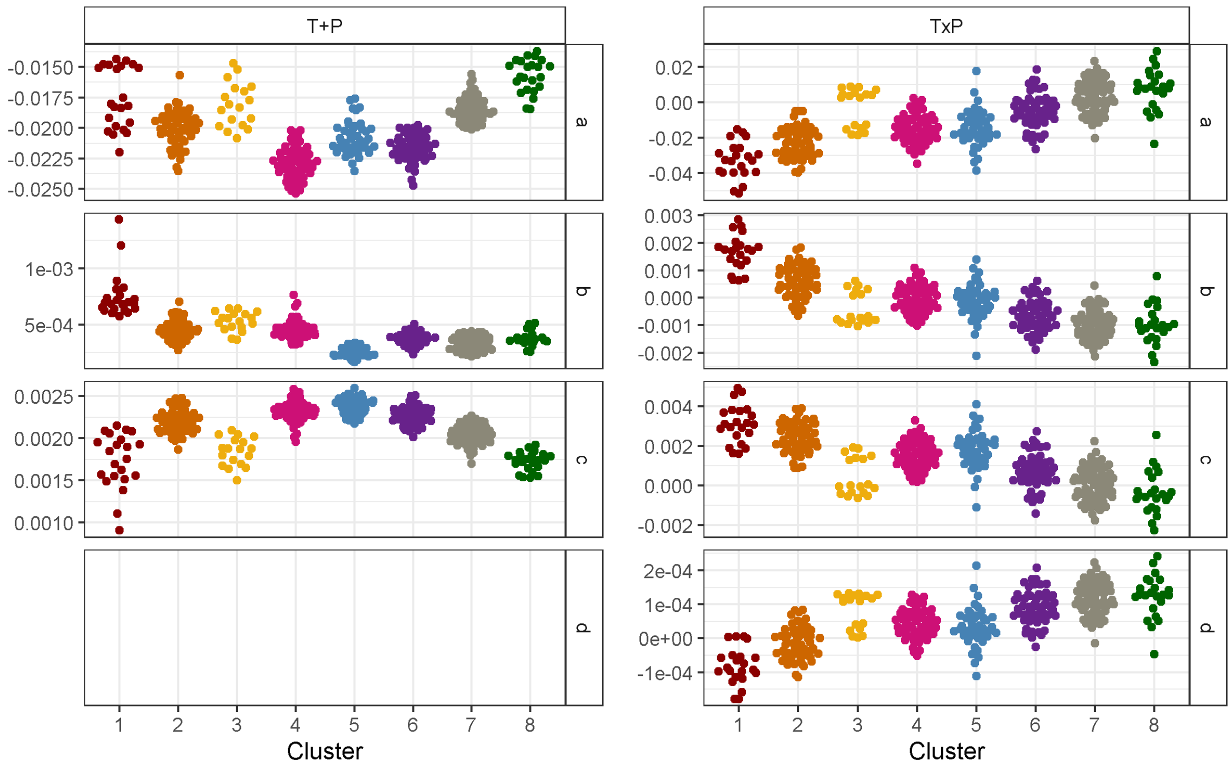
Supplemental Figure 5 Model T + P vs T x P

```r
# Prep data
xx <- read.csv("data/model_T+P_d.csv") %>% mutate(Expt = factor(Expt, levels = names_Expt))
```

```r
# Plot Observed vs Predicted
mp1 <- gg_model_1(xx, title = expression(paste("A) ", italic("1 / f = a + bT + cP"))))
# Prep data
xx <- read.csv("data/model_TxP_d.csv") %>% mutate(Expt = factor(Expt, levels = names_Expt))
# Plot Observed vs Predicted
mp2 <- gg_model_1(xx, title = expression(paste("B) ", italic("1 / f = a + bT + cP + dTP"))))
# Append
mp <- ggarrange(mp1, mp2, ncol = 2, common.legend = T, legend = "right")
ggsave("Supplemental_Figure_05_Models.png", mp, width = 12, height = 5)
```

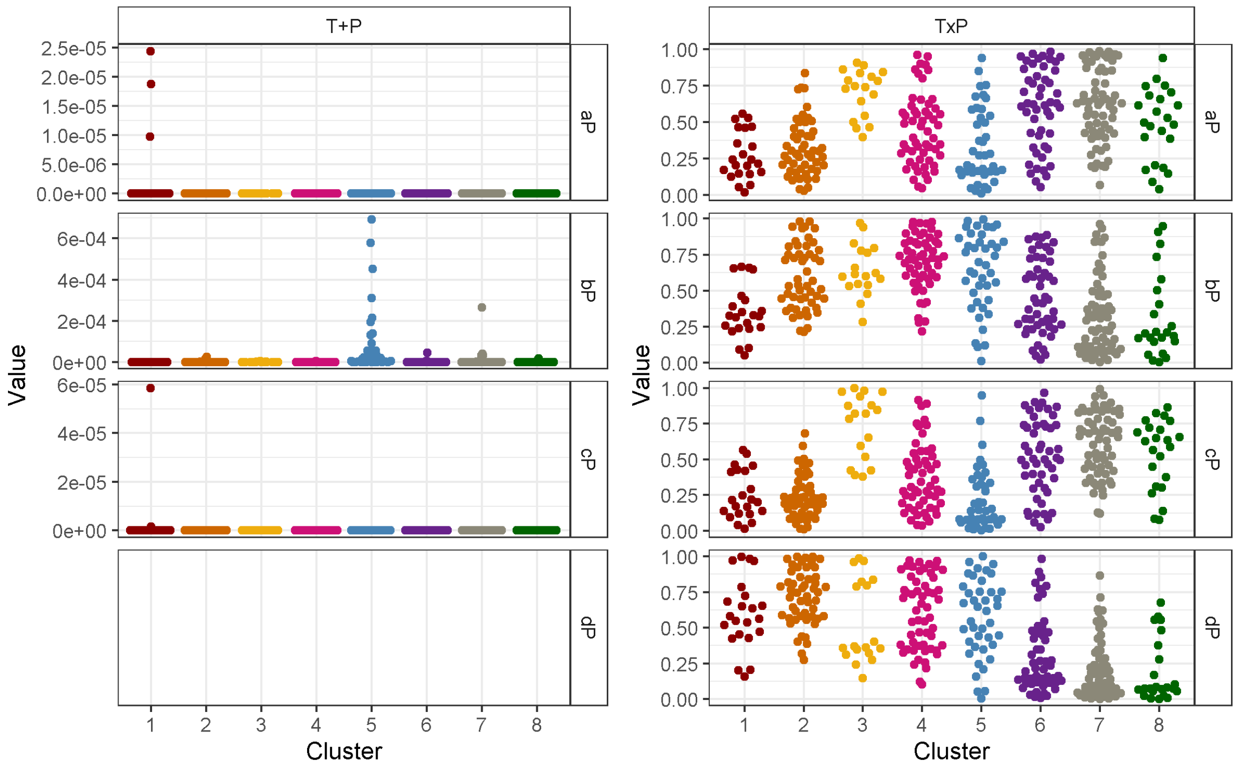
Additional Figure 10 Constants

```r
pca <- read.csv("data/data_PCA_Results.csv") %>% mutate(Cluster = factor(Cluster))
x1 <- read.csv("data/model_T+P_coefs.csv") %>% mutate(Model = "T+P")
x2 <- read.csv("data/model_TxP_coefs.csv") %>% mutate(Model = "TxP")
xx <- bind_rows(x1, x2) %>%
  gather(Coef, Value, a,b,c,d) %>%
  left_join(pca, by = "Entry") %>% mutate(Cluster = factor(Cluster))
mp1 <- ggplot(xx%>%filter(Model=="T+P"), aes(x = Cluster, y = Value, color = Cluster)) +
  geom_quasirandom() + theme_AGL +
  facet_grid(Coef~Model, scales = "free_y") +
  scale_color_manual(values = colors) + labs(y = NULL)
mp2 <- ggplot(xx%>%filter(Model=="TxP"), aes(x = Cluster, y = Value, color = Cluster)) +
  geom_quasirandom() + theme_AGL +
  facet_grid(Coef~Model, scales = "free_y") +
  scale_color_manual(values = colors) + labs(y = NULL)
mp <- ggarrange(mp1, mp2, ncol = 2, legend = "none")
ggsave("Additional/Additional_Figure_10_Coefs.png", mp, width = 8, height = 5)
```



Additional Figure 11 Coefficient p-values

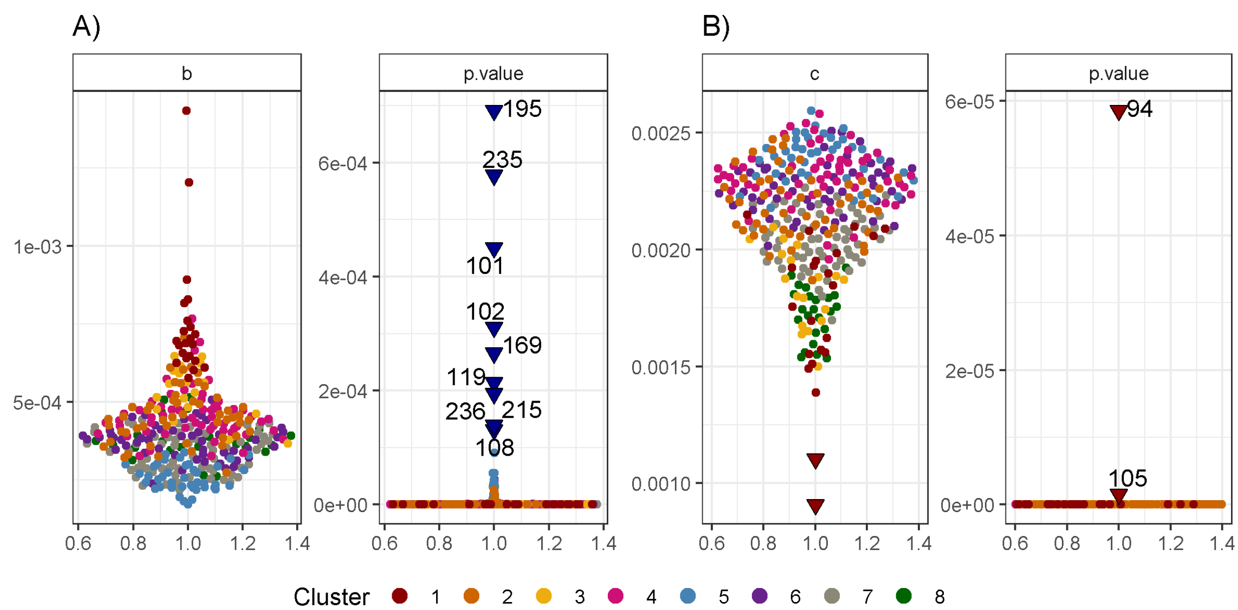
```
pca <- read.csv("data/data_PCA_Results.csv") %>% mutate(Cluster = factor(Cluster))
x1 <- read.csv("data/model_T+P_coefs.csv") %>% mutate(Model = "T+P")
x2 <- read.csv("data/model_TxP_coefs.csv") %>% mutate(Model = "TxP")
xx <- bind_rows(x1, x2) %>% gather(Coef, Value, aP,bP,cP,dP) %>%
  left_join(pca, by = "Entry") %>% mutate(Cluster = factor(Cluster))
mp1 <- ggplot(xx%>%filter(Model=="T+P"), aes(x = Cluster, y = Value, color = Cluster)) +
  geom_quasirandom() + theme_AGL +
  facet_grid(Coef~Model, scales = "free_y") +
  scale_color_manual(values = colors)
mp2 <- ggplot(xx%>%filter(Model=="TxP"), aes(x = Cluster, y = Value, color = Cluster)) +
  geom_quasirandom() + theme_AGL +
  facet_grid(Coef~Model, scales = "free_y") +
  scale_color_manual(values = colors)
mp <- ggarrange(mp1,mp2,ncol=2, legend = "none")
ggsave("Additional/Additional_Figure_11_CoefsP.png", mp, width = 8, height = 5)
```



Additional Figure 12 p-values b c

```r
# Prep A)
pca <- read.csv("data/data_PCA_Results.csv") %>% mutate(Cluster = factor(Cluster))
xx <- read.csv("data/model_T+P_coefs.csv") %>%
  mutate(Sig = factor(ifelse(bP > 0.0001, "Sig","Less Sig"))) %>%
  select(Entry, Sig, p.value=bP, b) %>%
  gather(Trait, Value, p.value, b) %>%
  left_join(pca, by = "Entry")
x1 <- xx %>% filter(Sig == "Sig", Trait == "p.value")
mp1 <- ggplot(xx, aes(x = 1, y = Value)) +
  geom_quasirandom(aes(color = Cluster)) +
    geom_point(data = x1, size = 2.5, pch = 25, color = "black", fill = "darkblue") +
  geom_text_repel(data = x1, aes(label = Entry)) +
  facet_wrap(.~Trait, scales = "free_y") +
  theme_AGL +
  scale_color_manual(values = colors) +
  guides(colour = guide_legend(nrow = 1, override.aes = list(size = 3))) +
  labs(title = "A)", x = NULL, y = NULL)
# Prep B)
xx <- read.csv("data/model_T+P_coefs.csv") %>%
  mutate(Sig = factor(ifelse(cP > 0.000001, "Sig","Less Sig"))) %>%
  select(Entry, Sig, p.value=cP, c) %>%
```

```
  gather(Trait, Value, p.value, c) %>%
  left_join(pca, by = "Entry")
x1 <- xx %>% filter(Sig == "Sig")
mp2 <- ggplot(xx, aes(x = 1, y = Value)) +
  geom_quasirandom(aes(color = Cluster)) +
  geom_point(data = x1, size = 2.5, pch = 25, color = "black", fill = "darkred") +
  geom_text_repel(data = x1 %>% filter(Trait =="p.value"), aes(label = Entry)) +
  facet_wrap(.~Trait, scales = "free_y") +
  theme_AGL +
  scale_color_manual(values = colors) +
  guides(colour = guide_legend(nrow = 1, override.aes = list(size = 3))) +
  labs(title = "B)", x = NULL, y = NULL)
mp <- ggarrange(mp1, mp2, ncol = 2, common.legend = T, legend = "bottom")
ggsave("Additional/Additional_Figure_12_bP.png", mp, width = 8, height = 4)
```



Modeling DTF (T + P) - Location Out

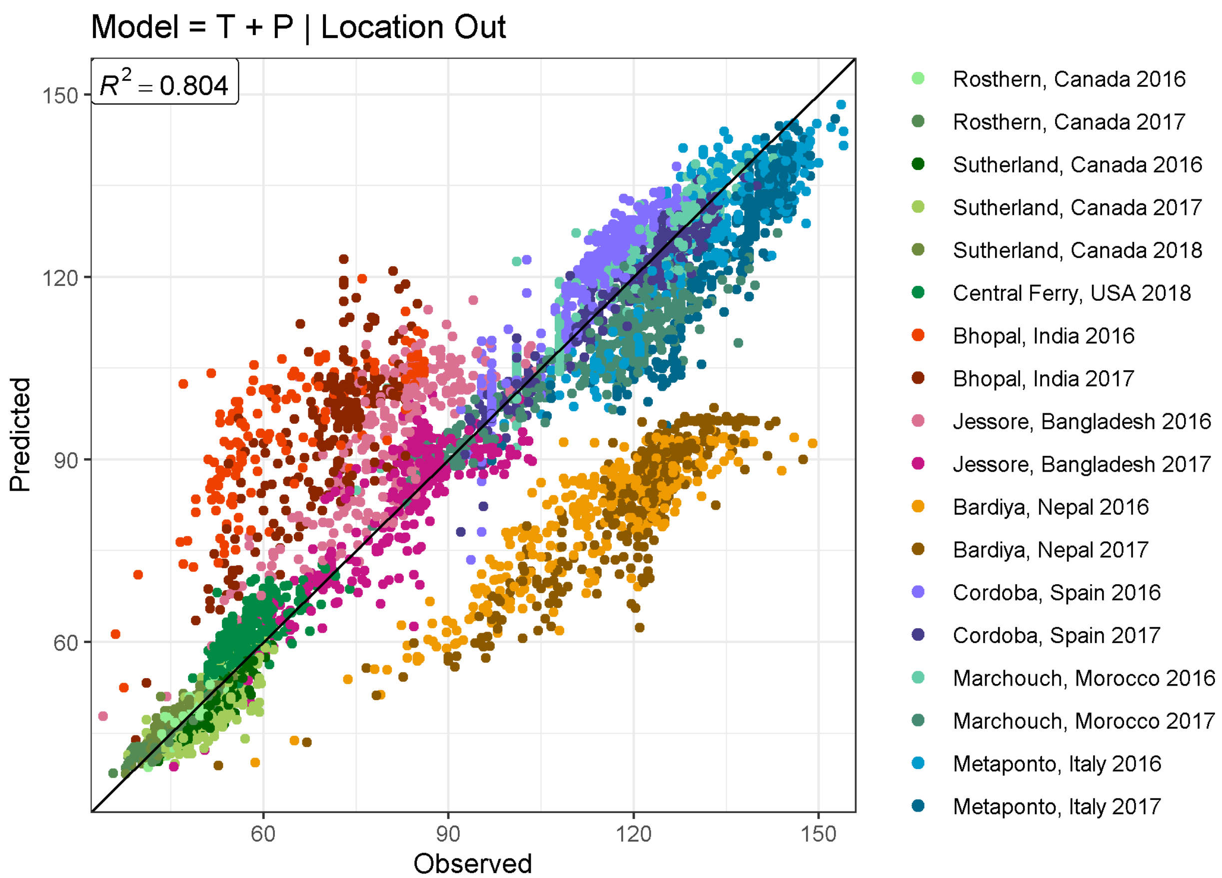
Train the model without the location used for prediction

```
######################################
# 1/f = a + bT + cP (Location Out) #
######################################
# Prep data
xx <- rr %>% filter(!is.na(RDTF)) %>%
  left_join(select(ff, Expt, T_mean, P_mean), by = "Expt")
mr <- NULL; md <- NULL
# Model - For each Location, the model is re-trained without that locations data
for(i in 1:324) {
  for(k in unique(xx$Location)) {
```

```
    # Prep data
    xi1 <- xx %>% filter(Entry == i, Location != k)
    xi2 <- xx %>% filter(Entry == i, Location == k)
    xd2 <- xi2 %>% group_by(Entry, Name, Expt, ExptShort) %>%
      summarise_at(vars(DTF, RDTF, T_mean, P_mean), funs(mean), na.rm = T) %>%
      ungroup()
    # Train model
    mi <- lm(RDTF ~ T_mean * P_mean, data = xi1)
    # Predict DTF
    xi2 <- xi2 %>% mutate(Predicted_DTF = 1 / predict(mi, newdata = xi2))
    xd2 <- xd2 %>% mutate(Predicted_DTF = 1 / predict(mi, newdata = xd2))
    # Save to table
    mr <- bind_rows(mr, xi2)
    md <- bind_rows(md, xd2)
  }
}
# Save Results
write.csv(mr, "data/model_Test.csv", row.names = F)
write.csv(md, "data/model_Test_d.csv", row.names = F)
```

Figure 4 Test Model

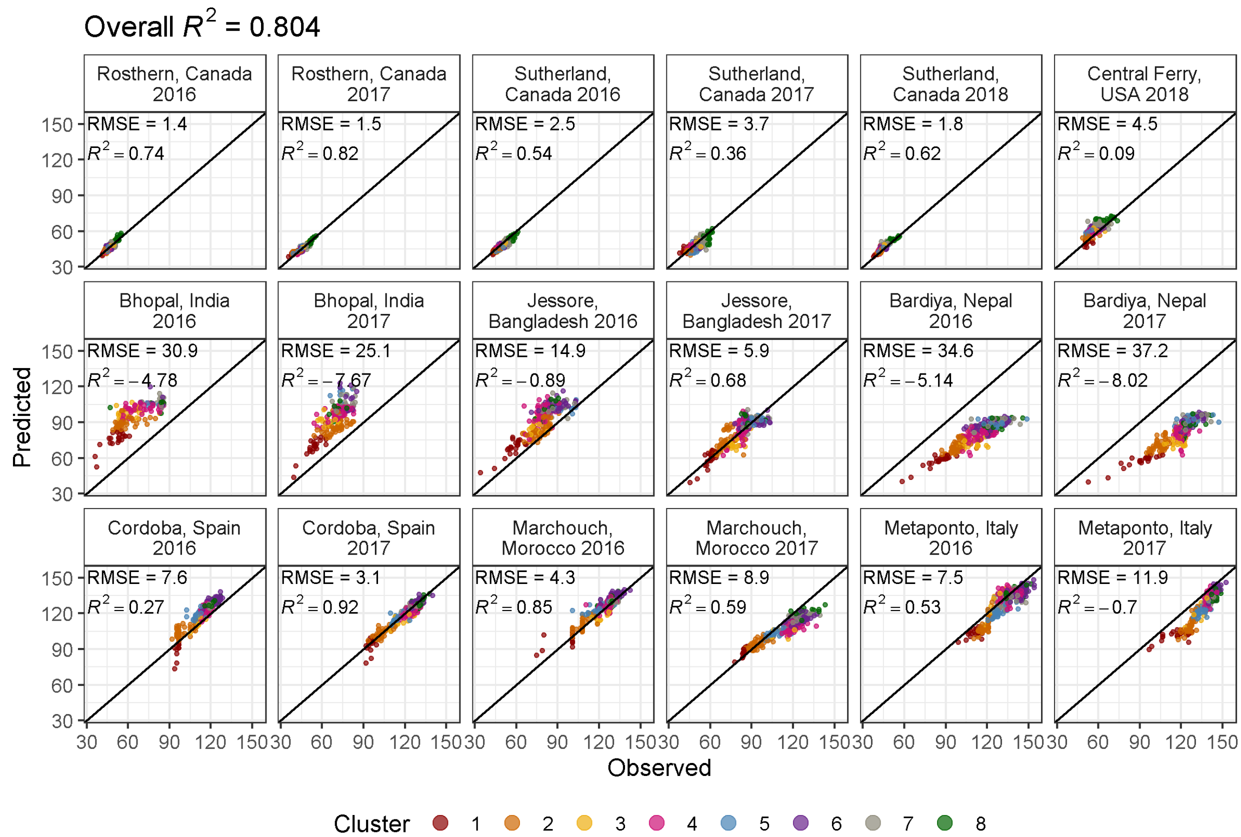
```
# Prep data
xx <- read.csv("data/model_Test_d.csv") %>% mutate(Expt = factor(Expt, levels = names_Expt))
# Plot Observed vs Predicted
mp <- gg_model_1(xx, title = "Model = T + P | Location Out")
ggsave("Additional/Model/Model_1_3.png", mp, width = 7, height = 5)
```




```r
modelR2(xx$DTF, xx$Predicted_DTF)
```

```
[1] 0.8035367
```

```r
# Plot A)
mp <- gg_model_2(xx, title = expression(paste("Overall ",italic("R")^2, " = 0.804")))
ggsave("Figure_04_ModelPrediction.png", mp, width = 8, height = 5.5)
mp <- gg_model_2(xx, title = "Model = T + P | Location Out")
ggsave("Additional/Model/Model_2_3.png", mp, width = 8, height = 5.5)
```



Supplemental Table 4 Test Model

```r
######################################
# 1/f = a + bT + cP (3 Locations) #
######################################
# Prep data
xx <- rr %>% #filter(!is.na(DTF)) %>%
  left_join(select(ff, Expt, T_mean, P_mean), by = "Expt")
mt <- data.frame(Temperate_Location     = rep(names_ExptShort[1:6],  each = 36),
                 SouthAsian_Location    = rep(names_ExptShort[7:12], times = 36)) %>%
  arrange(SouthAsian_Location) %>%
  mutate(Mediterranean_Location = rep(names_ExptShort[13:18], 36),
                 RR = NA, Genotypes = NA)
sum(duplicated(paste(mt$Temperate_Location,mt$SouthAsian_Location,mt$Mediterranean_Location)))
#
#t<-names_ExptShort[1]
#s<-names_ExptShort[7]
#m<-names_ExptShort[13]
for(t in names_ExptShort[1:6]) { # Temperate site-years
  for(s in names_ExptShort[7:12]) { # South asian site-years
    for(m in names_ExptShort[13:18]) { # Mediterranean site-years
```

```r
      mr <- NULL; md <- NULL
      for(i in 1:324) {
        # Prep data
        xi1 <- xx %>% filter(Entry == i, ExptShort %in% c(t, s, m))
        xi2 <- xx %>% filter(Entry == i)
        xd2 <- xi2 %>% group_by(Entry, Name, Expt, ExptShort) %>%
          summarise_at(vars(DTF, RDTF, T_mean, P_mean), funs(mean), na.rm = T) %>%
          ungroup()
        # Train model
        mi <- lm(RDTF ~ T_mean + P_mean, data = xi1)
        # Predict DTF
        xi2 <- xi2 %>% mutate(Predicted_DTF = 1 / predict(mi, newdata = xi2))
        xd2 <- xd2 %>% mutate(Predicted_DTF = 1 / predict(mi, newdata = xd2))
        # Save to table
        mr <- bind_rows(mr, xi2)
        md <- bind_rows(md, xd2)
      }
      remEntries <- unique(md$Entry[is.na(md$DTF) & md$ExptShort %in% c(t, s, m)])
      md2 <- md %>% filter(!Entry %in% remEntries, !md$ExptShort %in% c(t, s, m))
      myrow <- mt$Temperate_Location==t & mt$SouthAsian_Location==s & mt$Mediterranean_Location==m
      mt[myrow,"RR"] <- modelR2(md2$DTF, md2$Predicted_DTF)
      mt[myrow,"Genotypes"] <- length(unique(md2$Entry))
    }
  }
}
# Save
write.csv(mt %>% arrange(RR), "Supplemental_Table_04.csv", row.names = F)
#
xx <- mt %>%
  mutate(Model = paste(Temperate_Location, SouthAsian_Location, Mediterranean_Location, sep = "-")) %>%
  arrange(RR) %>%
  mutate(Model = factor(Model, levels = unique(Model)))
mp <- ggplot(xx, aes(x = Model, y = RR)) +
  geom_bar(stat = "identity") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
ggsave("Additional/Model_3_Locations.png", mp, width = 30, height = 5)
```

Modeling DTF (T + P) - 3 Best

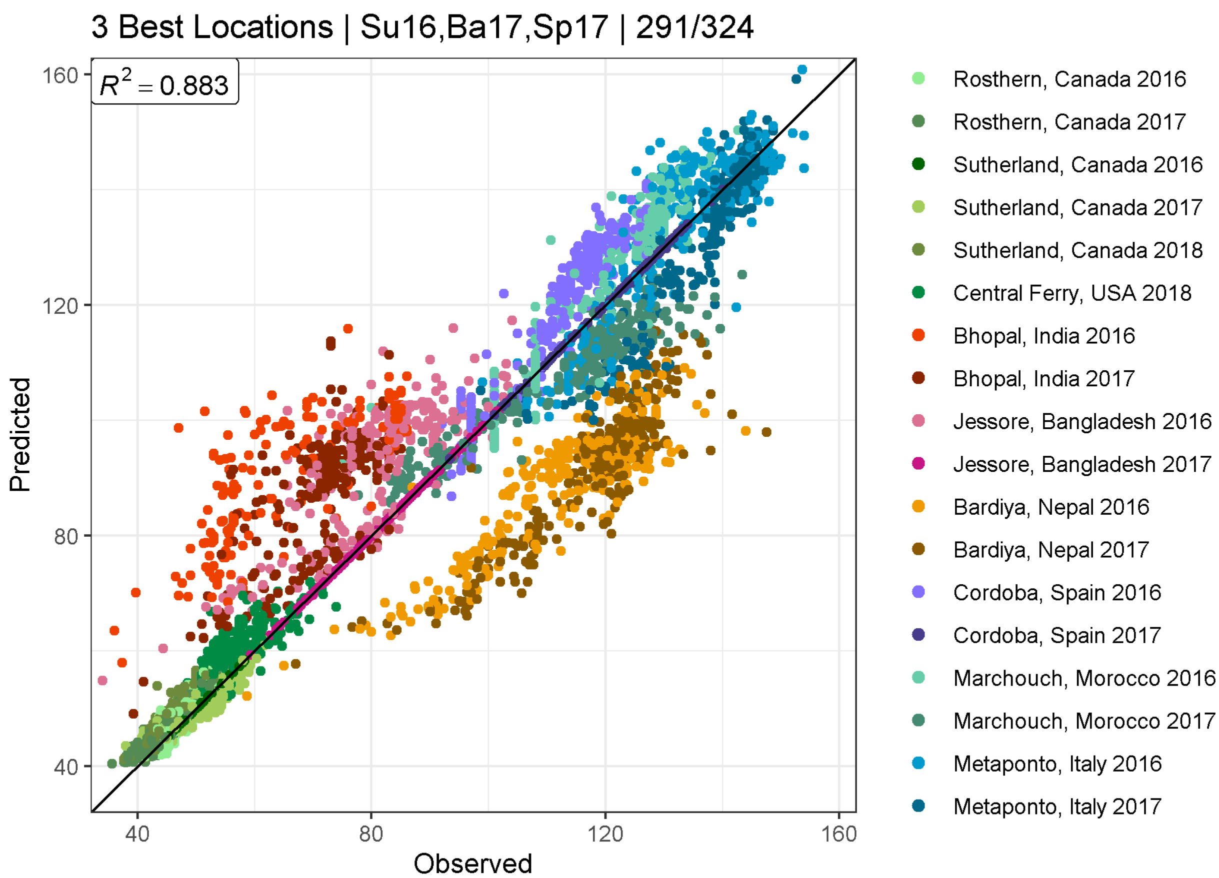
```r
####################################
# 1/f = a + bT + cP (3 Locations) #
####################################
# Prep data
xx <- rr %>% filter(!is.na(RDTF)) %>%
  left_join(select(ff, Expt, T_mean, P_mean), by = "Expt")
mr <- NULL; md <- NULL
mc <- select(ldp, Entry, Name) %>%
  mutate(a = NA, b = NA, c = NA, RR = NA, Environments = NA )
k <- c("Sutherland, Canada 2016", "Jessore, Bangladesh 2017", "Cordoba, Spain 2017")
```

```r
# Model - only the ^above^ three site-years are used to train the model
for(i in 1:324) {
  # Prep data
  xi1 <- xx %>% filter(Entry == i, Expt %in% k)
  xi2 <- xx %>% filter(Entry == i)
  xd2 <- xi2 %>% group_by(Entry, Name, Expt, ExptShort) %>%
    summarise_at(vars(DTF, RDTF, T_mean, P_mean), funs(mean), na.rm = T) %>%
    ungroup()
  # Train model
  mi <- lm(RDTF ~ T_mean * P_mean, data = xi1)
  # Predict DTF
  xi2 <- xi2 %>% mutate(Predicted_DTF = 1 / predict(mi, newdata = xi2))
  xd2 <- xd2 %>% mutate(Predicted_DTF = 1 / predict(mi, newdata = xd2))
  # Save to table
  mr <- bind_rows(mr, xi2)
  md <- bind_rows(md, xd2)
  # Save coefficients
  mc[i,c(3:5)] <- mi$coefficients
  # Calculate rr and # of environments used
  mc[i,6] <- 1 - sum((xi2$DTF - xi2$Predicted_DTF)^2) /
    sum((xi2$Predicted_DTF - mean(xi2$DTF))^2)
  mc[i,7] <- length(unique(xi2$Expt))
}
ents <- xx %>% filter(ExptShort %in% c("Su16", "Ba17", "Sp17"), is.na(DTF)) %>%
  pull(Entry) %>% unique()
mr <- mr %>% filter(!Entry %in% ents)
md <- md %>% filter(!Entry %in% ents)
mc <- mc %>% filter(!Entry %in% ents)
# Save Results
write.csv(mr, "data/model_3best.csv",        row.names = F)
write.csv(md, "data/model_3best_d.csv",      row.names = F)
write.csv(mc, "data/model_3best_coefs.csv",  row.names = F)
```

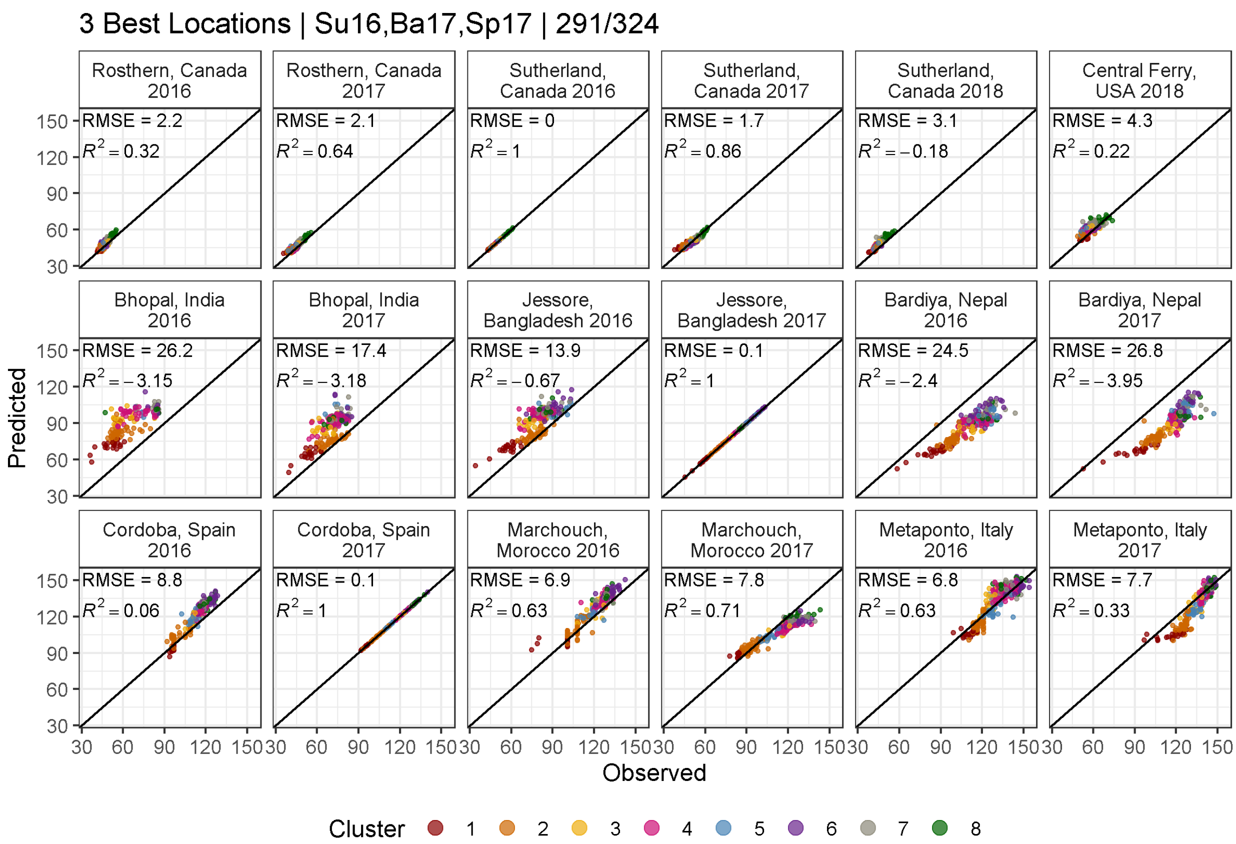
```r
# Prep data
xx <- read.csv("data/model_3best_d.csv") %>% mutate(Expt = factor(Expt, levels = names_Expt))
324 - length(unique(xx$Entry))
```

```
[1] 33
```

```r
# Plot Observed vs Predicted
mp <- gg_model_1(xx, title = "3 Best Locations | Su16,Ba17,Sp17 | 291/324")
ggsave("Additional/Model/Model_1_4.png", mp, width = 7, height = 5)
```




```r
# Plot B)
mp <- gg_model_2(xx, title = "3 Best Locations | Su16,Ba17,Sp17 | 291/324")
ggsave("Additional/Model/Model_2_4.png", mp, width = 8, height = 5.5)
```



Modeling DTF (T + P) - 3 Worst

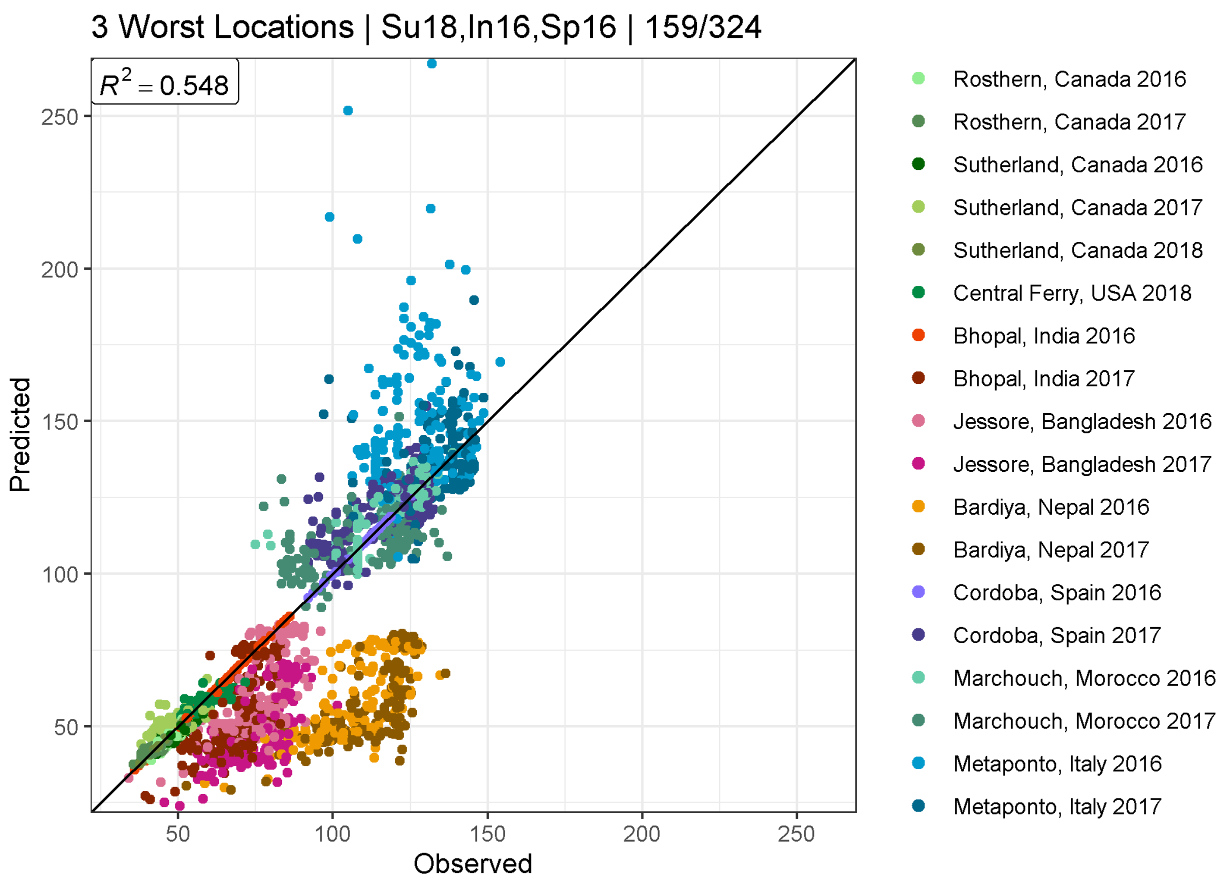
```
#####################################
# 1/f = a + bT + cP (3 Locations)  #
#####################################
# Prep data
xx <- rr %>% filter(!is.na(RDTF)) %>%
  left_join(select(ff, Expt, T_mean, P_mean), by = "Expt")
mr <- NULL; md <- NULL
mc <- select(ldp, Entry, Name) %>%
  mutate(a = NA, b = NA, c = NA, RR = NA, Environments = NA )
k <- c("Sutherland, Canada 2018", "Bhopal, India 2016", "Cordoba, Spain 2016")
# Model - only the ~above~ three site-years are used to train the model
for(i in 1:324) {
  # Prep data
  xi1 <- xx %>% filter(Entry == i, Expt %in% k)
  xi2 <- xx %>% filter(Entry == i)
  xd2 <- xi2 %>% group_by(Entry, Name, Expt, ExptShort) %>%
    summarise_at(vars(DTF, RDTF, T_mean, P_mean), funs(mean), na.rm = T) %>%
    ungroup()
  # Train model
```

```r
  mi <- lm(RDTF ~ T_mean * P_mean, data = xi1)
  # Predict DTF
  xi2 <- xi2 %>% mutate(Predicted_DTF = 1 / predict(mi, newdata = xi2))
  xd2 <- xd2 %>% mutate(Predicted_DTF = 1 / predict(mi, newdata = xd2))
  # Save to table
  mr <- bind_rows(mr, xi2)
  md <- bind_rows(md, xd2)
  # Save coefficients
  mc[i,c(3:5)] <- mi$coefficients
  # Calculate rr and # of environments used
  mc[i,6] <- 1 - sum((xi2$DTF - xi2$Predicted_DTF)^2) /
    sum((xi2$Predicted_DTF - mean(xi2$DTF))^2)
  mc[i,7] <- length(unique(xi2$Expt))
}
ents <- xx %>% filter(ExptShort %in% c("Su18", "In16", "Sp16"), is.na(DTF)) %>%
  pull(Entry) %>% unique()
mr <- mr %>% filter(!Entry %in% ents)
md <- md %>% filter(!Entry %in% ents)
mc <- mc %>% filter(!Entry %in% ents)
# Save Results
write.csv(mr, "data/model_3Worst.csv",        row.names = F)
write.csv(md, "data/model_3Worst_d.csv",      row.names = F)
write.csv(mc, "data/model_3Worst_coefs.csv",  row.names = F)
```

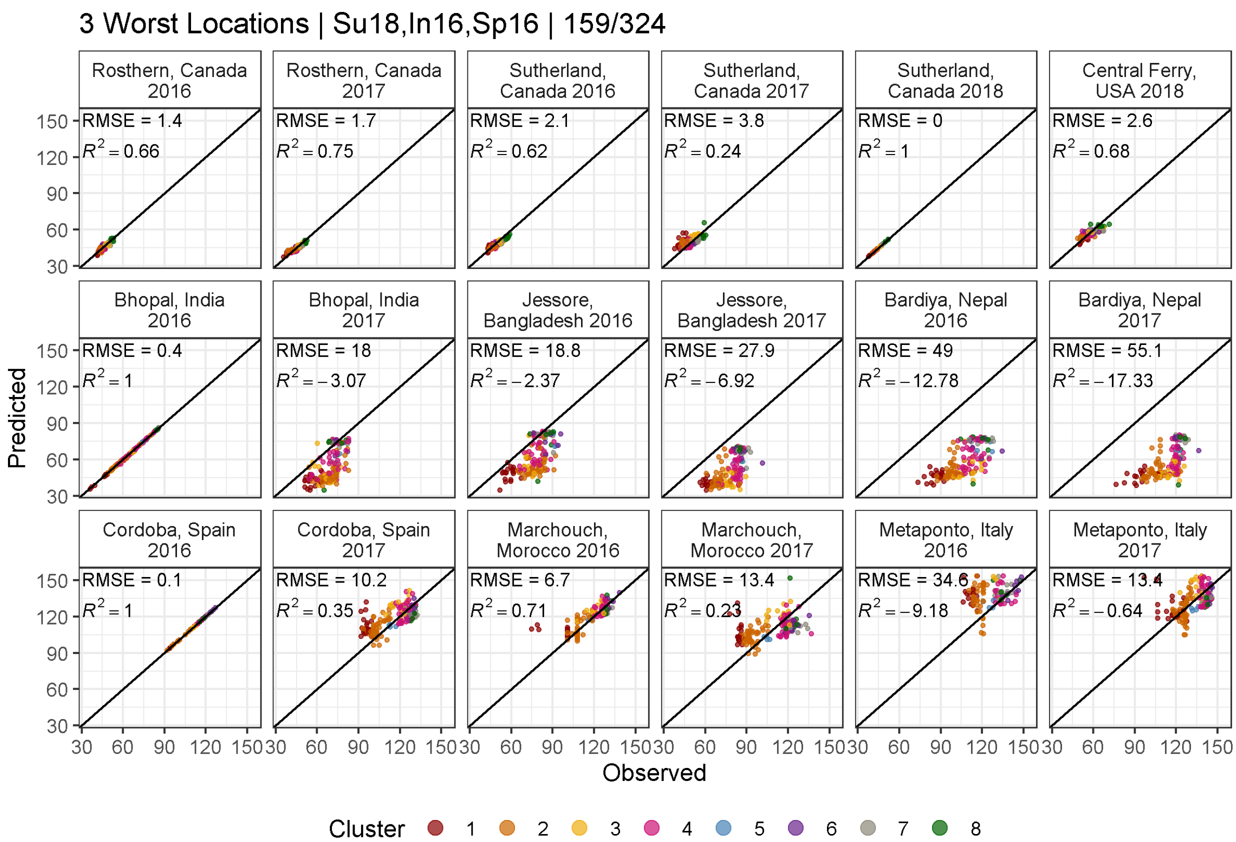
```r
# Prep data
xx <- read.csv("data/model_3Worst_d.csv") %>% mutate(Expt = factor(Expt, levels = names_Expt))
324 - length(unique(xx$Entry))
```

```
[1] 165
```

```r
# Plot Observed vs Predicted
mp <- gg_model_1(xx, title = "3 Worst Locations | Su18,In16,Sp16 | 159/324")
ggsave("Additional/Model/Model_1_5.png", mp, width = 7, height = 5)
```

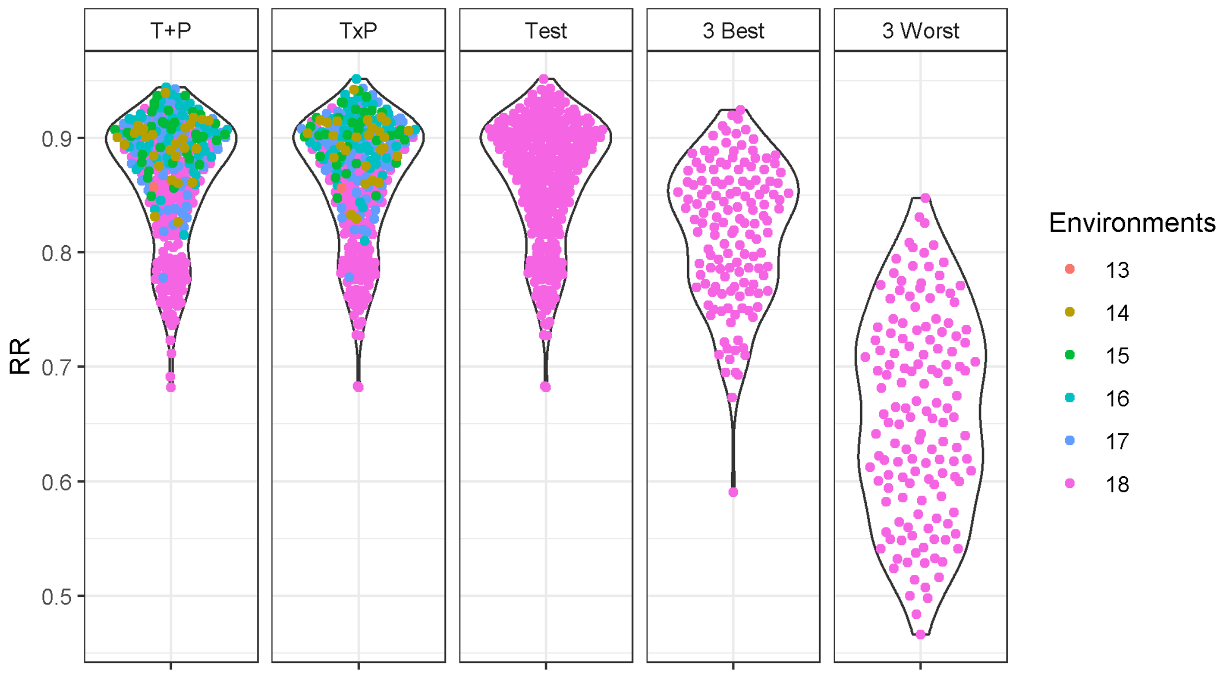



```
# Plot B)
mp <- gg_model_2(xx, title = "3 Worst Locations | Su18,In16,Sp16 | 159/324")
ggsave("Additional/Model/Model_2_5.png", mp, width = 8, height = 5.5)
```



DTF Model correlation coefficients

```
# Prep data
x1 <- read.csv("data/model_T+P_coefs.csv") %>% mutate(Model = "T+P")
x2 <- read.csv("data/model_TxP_coefs.csv") %>% mutate(Model = "TxP")
x3 <- read.csv("data/model_Test_coefs.csv") %>% mutate(Model = "Test")
x4 <- read.csv("data/model_3Best_coefs.csv")  %>% mutate(Model = "3 Best")
x5 <- read.csv("data/model_3Worst_coefs.csv")  %>% mutate(Model = "3 Worst")
xx <- bind_rows(x1, x2, x3, x4, x5) %>%
  mutate(Model = factor(Model, levels = unique(Model)),
         Environments = factor(Environments))
# Plot RR
mp <- ggplot(xx, aes(x = "", y = RR)) +
  geom_violin() + geom_quasirandom(aes(color = Environments)) +
  facet_grid(. ~ Model) +
  theme_AGL + labs(x = NULL)
ggsave("Additional/Model/Model_pvalues.png", mp, width = 7, height = 4)
```

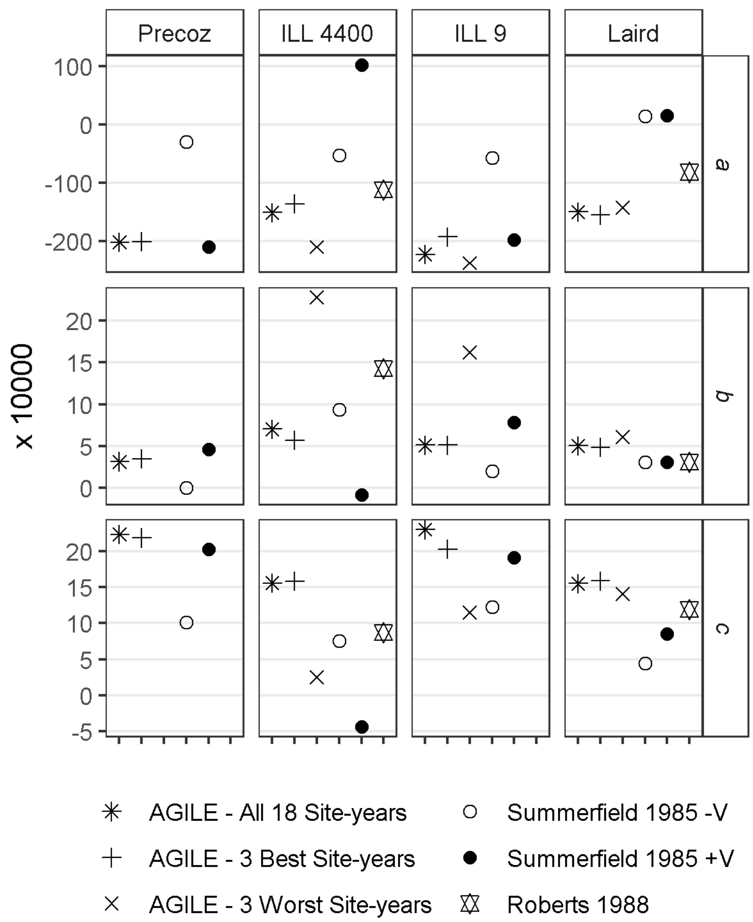


Supplemental Figure 6 Compare Constants Entry

- Entry 76 = ILL 4400 (Syrian Local Large)
- Entry 77 = ILL 4605 (Precoz)
- Entry 118 = ILL 9
- Entry 128 = Laird

```
# Prep data
x1 <- read.csv("data/model_T+P_coefs.csv") %>%
  filter(Entry %in% c(76, 77, 118, 128)) %>%
  mutate(Expt = "AGILE - All 18 Site-years")
x2.1 <- read.csv("data/model_3Best_coefs.csv") %>%
  filter(Entry %in% c(76, 77, 118, 128)) %>%
  mutate(Expt = "AGILE - 3 Best Site-years")
x2.2 <- read.csv("data/model_3Worst_coefs.csv") %>%
  filter(Entry %in% c(76, 77, 118, 128)) %>%
  mutate(Expt = "AGILE - 3 Worst Site-years")
# Summerfield et al., 1985
x3 <- x1 %>% mutate(Expt = "Summerfield 1985 -V")
x3[x3$Entry == 76,  c("a","b","c")] <- c(-0.002918,  0,          0.0010093)
x3[x3$Entry == 77,  c("a","b","c")] <- c(-0.0052226, 0.00093643, 0.00075104)
x3[x3$Entry == 118, c("a","b","c")] <- c(-0.0057408, 0.00020113, 0.0012292)
x3[x3$Entry == 128, c("a","b","c")] <- c( 0.0014689, 0.00030622, 0.00044640)
x4 <- x1 %>% mutate(Expt = "Summerfield 1985 +V")
x4[x4$Entry == 76,  c("a","b","c")] <- c(-0.020910,   0.00045813,  0.0020210)
x4[x4$Entry == 77,  c("a","b","c")] <- c( 0.0101590, -0.00008401, -0.00044067)
```

```r
x4[x4$Entry == 118, c("a","b","c")] <- c(-0.0196948,  0.00078441,  0.0019110)
x4[x4$Entry == 128, c("a","b","c")] <- c( 0.0015094,  0.00030622,  0.00085502)
# Roberts et al., 1988
x5 <- x1 %>% filter(Entry %in% c(77, 128)) %>% mutate(Expt = "Roberts 1988")
x5[x5$Entry == 77,  c("a","b","c")] <- c(-0.0112,   0.001427, 0.000871)
x5[x5$Entry == 128, c("a","b","c")] <- c(-0.008172, 0.000309, 0.001187)
#
xx <- bind_rows(x1, x2.1, x2.2, x3, x4, x5) %>%
  gather(Constant, Value, a, b, c) %>%
  mutate(Entry = factor(Entry),
         Name = plyr::mapvalues(Entry, c(76,77,118,128),
                  c("Precoz","ILL 4400","ILL 9","Laird")),
         Expt = factor(Expt, levels = c("AGILE - All 18 Site-years",
                  "AGILE - 3 Best Site-years", "AGILE - 3 Worst Site-years",
                  "Summerfield 1985 -V", "Summerfield 1985 +V", "Roberts 1988")))
# Plot a, b and c
mp <- ggplot(xx, aes(x = Expt, y = Value * 10000, shape = Expt)) +
  geom_quasirandom(size = 2, width = 0.2) +
  facet_grid(Constant ~ Name, scales = "free_y") +
  scale_shape_manual(name = NULL, values = c(8,3,4,1,16,11)) +
  guides(shape=guide_legend(nrow = 3, byrow = F)) +
  theme_AGL +
  theme(legend.position = "bottom",
        strip.text.y = element_text(face = "italic"),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        axis.text.x = element_blank()) +
  labs(x = NULL, y = "x 10000")
ggsave("Supplemental_Figure_06_CompareConstants.png", mp, width = 4, height = 5)
```

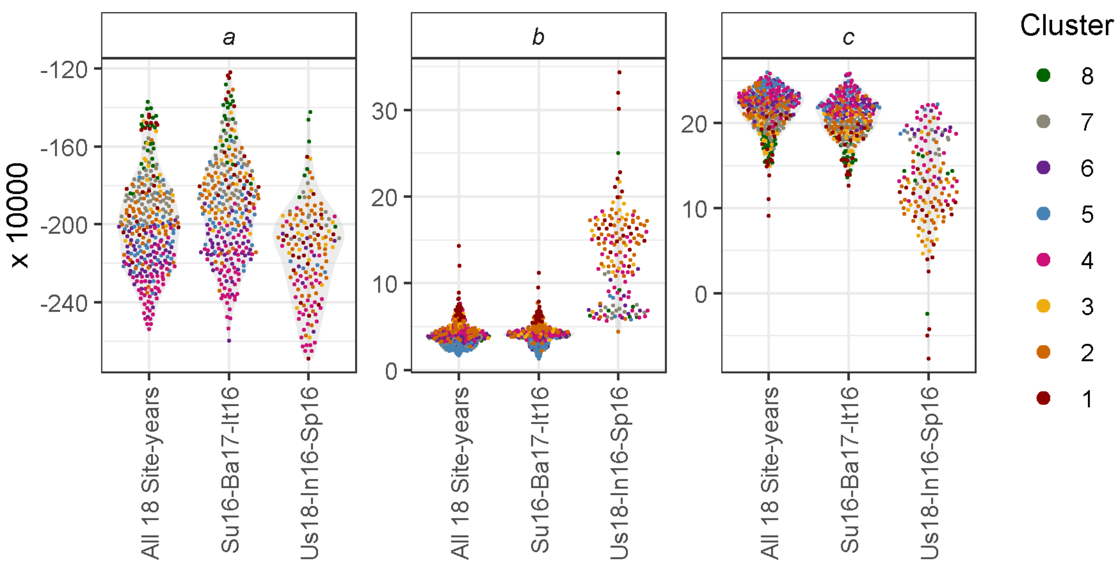


Supplemental Figure 7 Compare Constants All

```
# Prep data
levs <- c("All 18 Site-years", "Su16-Ba17-It16", "Us18-In16-Sp16")
pca <- read.csv("data/data_PCA_Results.csv") %>% select(Entry, Cluster) %>%
  mutate(Cluster = factor(Cluster))
x1 <- read.csv("data/model_T+P_coefs.csv") %>%
  mutate(Expt = levs[1]) %>% select(-RR)
x2 <- read.csv("data/model_3Best_coefs.csv") %>%
  mutate(Expt = levs[2])
x3 <- read.csv("data/model_3Worst_coefs.csv") %>%
  mutate(Expt = levs[3])

xx <- bind_rows(x1, x2,x3) %>%
  left_join(pca, by = "Entry") %>%
  gather(Trait, Value, a, b, c) %>%
  mutate(Expt = factor(Expt,levels = levs))
# Plot a, b, and c
```

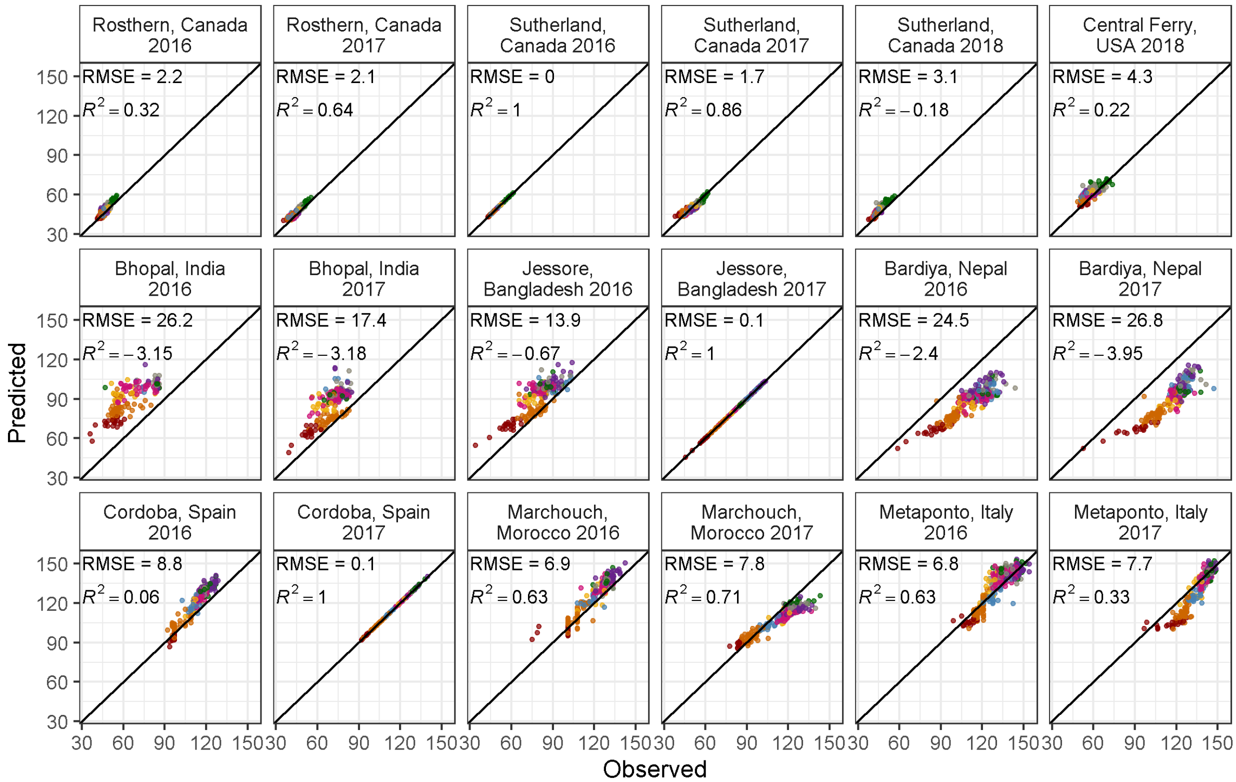
```
mp <- ggplot(xx, aes(x = Expt, y = Value * 10000 )) +
  geom_violin(alpha = 0.3, color = NA, fill = "grey") +
  geom_quasirandom(aes(color = Cluster), size = 0.3) +
  facet_wrap(Trait ~ ., scales = "free") +
  scale_color_manual(values = colors, breaks = 8:1) +
  theme_AGL +
  theme(strip.text = element_text(face = "italic"),
        axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1)) +
  guides(colour = guide_legend(override.aes = list(size = 2))) +
  labs(x = NULL, y = "x 10000")
ggsave("Supplemental_Figure_07_ConstantsCompare.png", mp, width = 6, height = 3)
```



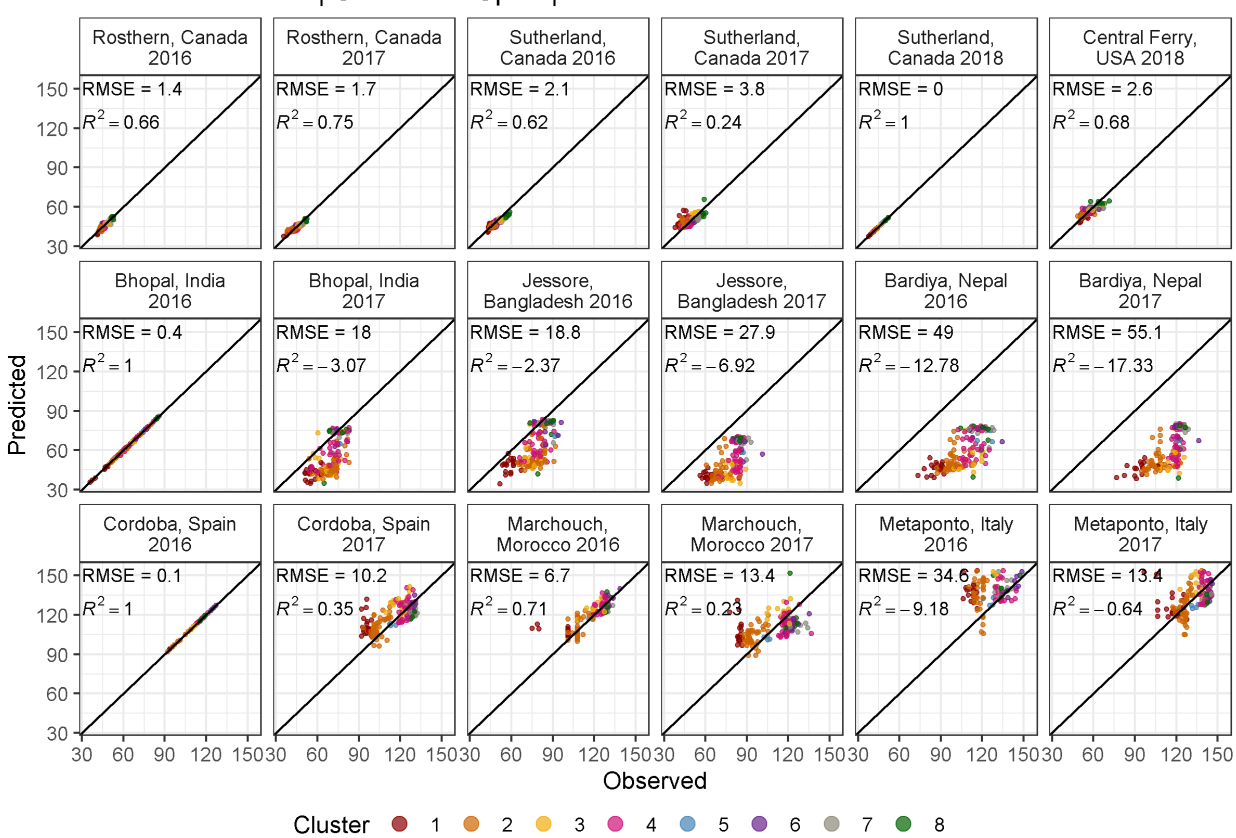
Supplemental Figure 8 3 best 3 worst

```
# Plot A)
xx <- read.csv("data/model_3best_d.csv") %>% mutate(Expt = factor(Expt, levels = names_Expt))
mp1 <- gg_model_2(xx, title = "3 Best Locations | Su16-Ba17-It16 | 291/324")
# Plot B)
xx <- read.csv("data/model_3Worst_d.csv") %>% mutate(Expt = factor(Expt, levels = names_Expt))
mp2 <- gg_model_2(xx, title = "3 Worst Locations | Us18-In16-Sp16 | 159/324")
# Append A) and B)
mp <- ggarrange(mp1, mp2, ncol = 1, common.legend = T, legend = "bottom")
ggsave("Supplemental_Figure_08_TestModel.png", mp, width = 8, height = 11)
```

3 Best Locations | Su16-Ba17-It16 | 291/324



3 Worst Locations | Us18-In16-Sp16 | 159/324



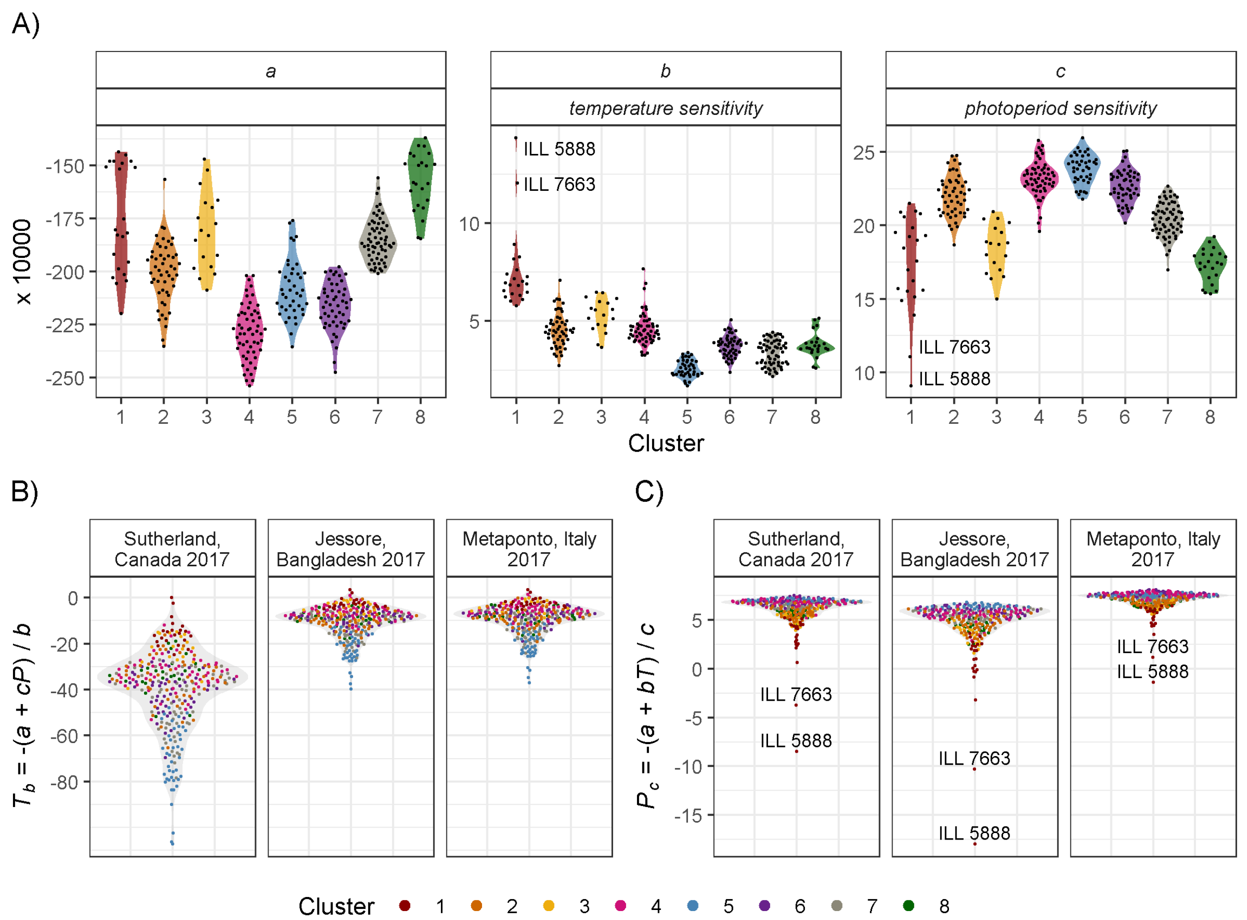
Base Temperature & Critical Photoperiod

```
# Calculate Tf and Pf
xx <- read.csv("data/model_T+P_coefs.csv") %>% select(-Name) %>%
  mutate(predicted_Tf = 1/b, predicted_Pf = 1/c )
xx <- rr %>% left_join(xx, by = "Entry") %>%
  left_join(select(ff, Expt, T_mean, P_mean), by = "Expt") %>%
  mutate(Tb = -(a + c * P_mean) / b,
         Pc = -(a + b * T_mean) / c,
         Tf_0 = NA, Tf_5 = NA, Tf = NA, Pf = NA, PTT = NA)
for(i in 1:nrow(xx)) {
  e1 <- ee %>% filter(Expt == xx$Expt[i])
  for(k in 1:nrow(e1)) {
    e1$Tfsum[k] <- sum(e1$Temp_mean[1:k] - xx$Tb[i])
    e1$Pfsum[k] <- sum(e1$DayLength[1:k] - xx$Pc[i])
  }
  ei <- e1 %>%
    filter(Date <= xx$PlantingDate[i] + xx$DTF2[i], !is.na(Temp_mean))
  if(nrow(ei) > 0) {
    xx$Tf_0[i]  <- round(sum(ei$Temp_mean), 1)
    xx$Tf_5[i]  <- round(sum(ei$Temp_mean - 5), 1)
    xx$Tf[i]    <- round(sum(ei$Temp_mean - xx$Tb[i]), 1)
    xx$Pf_0[i]  <- round(sum(ei$DayLength), 1)
    xx$Pf_7[i]  <- round(sum(ei$DayLength - 7), 1)
    xx$Pf[i]    <- round(sum(ei$DayLength - xx$Pc[i]), 1)
    xx$PTT_0[i] <- round(sum(ei$Temp_mean * ei$DayLength), 1)
    xx$PTT[i]   <- round(sum((ei$Temp_mean - xx$Tb[i]) * (ei$DayLength - xx$Pc[i])), 1)
    eTf <- e1 %>% filter(Tfsum > xx$predicted_Tf[i])
    ePf <- e1 %>% filter(Pfsum > xx$predicted_Pf[i])
    xx$predicted_DTF_Tf[i] <- eTf$DaysAfterPlanting[1]
    xx$predicted_DTF_Pf[i] <- ePf$DaysAfterPlanting[1]
  }
}
xx <- xx %>% left_join(select(ff, Expt, MacroEnv), by = "Expt") %>%
  group_by(Entry, Name, Expt, ExptShort, MacroEnv) %>%
  summarise_at(vars(DTF, Tb, Pc, Tf_0, Tf_5, Tf, Pf_0, Pf_7, Pf, PTT, PTT_0,
                    predicted_DTF_Tf, predicted_DTF_Pf,
                    predicted_Pf, predicted_Tf), funs(mean), na.rm = T) %>%
  ungroup()
# Save
write.csv(xx,"data/data_Tf_Pf.csv", row.names = F)
```

Figure 5 Tb and Pc

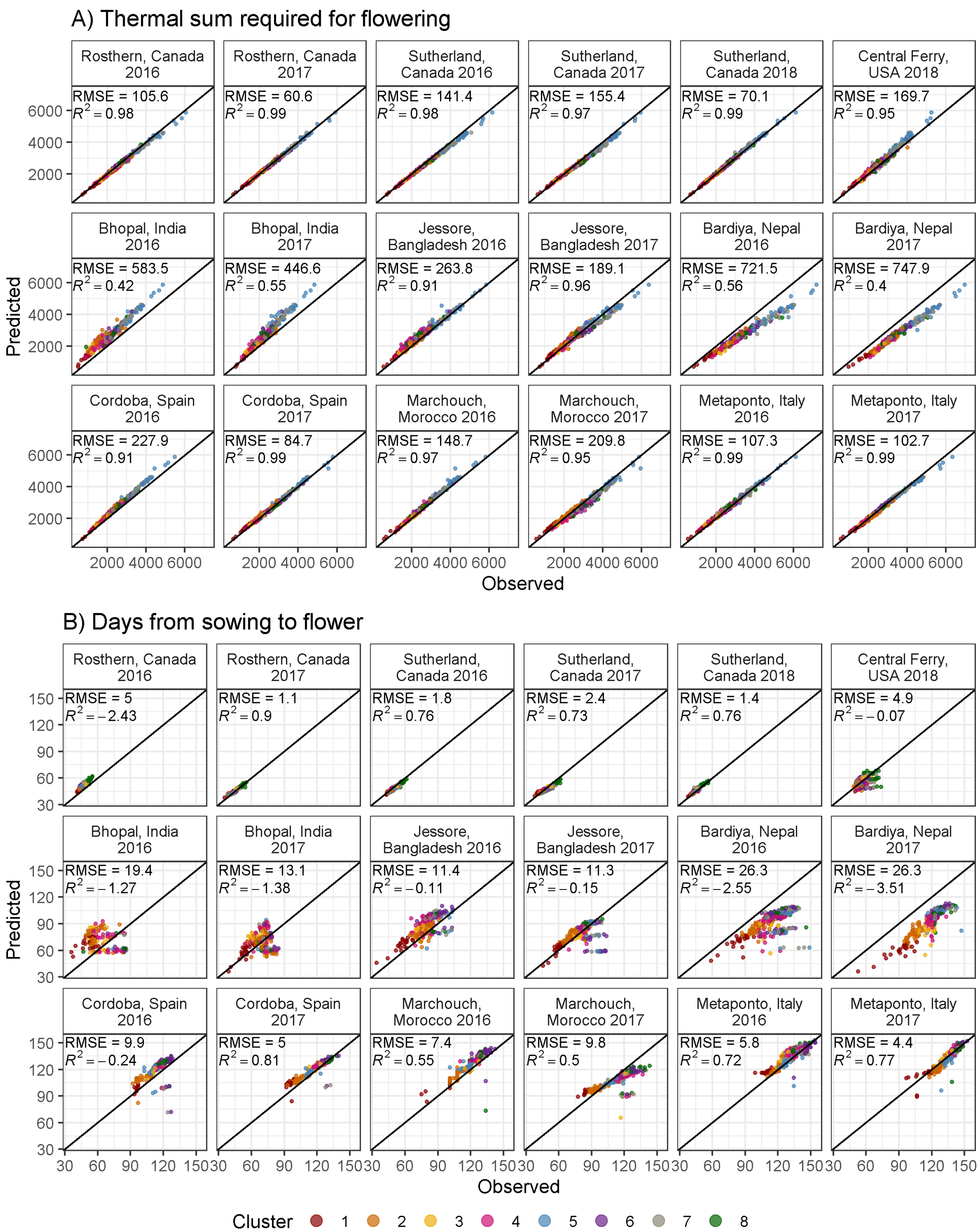
```
# Prep data for A) a, b and c
pca <- read.csv("data/data_PCA_Results.csv") %>% select(Entry, Cluster) %>%
  mutate(Cluster = factor(Cluster))
xx <- read.csv("data/model_T+P_coefs.csv") %>%
  left_join(pca, by = "Entry") %>%
  select(Entry, Name, Cluster, a, b, c) %>%
  gather(Constant, Value, 4:ncol(.)) %>%
  mutate(Meaning = plyr::mapvalues(Constant, c("a","b","c"),
           c("", "temperature sensitivity", "photoperiod sensitivity")))
#genotypes <- c("ILL 5888", "ILL 4400", "Laird")
#x4 <- xx %>% filter(Entry %in% c(76,94,128)) %>%
#  mutate(Name = gsub(" AGL", "", Name),
#         Name = factor(Name, levels = genotypes))
x1 <- xx %>% filter(Entry %in% c(94,105), Constant != "a") %>%
  mutate(Name = gsub(" AGL", "", Name))
# Plot A) a, b and c
mp1 <- ggplot(xx, aes(x = Cluster, y = Value * 10000) ) +
  geom_violin(aes(fill = Cluster), color = NA, alpha = 0.7) +
  geom_quasirandom(size = 0.3) +
  geom_text_repel(data = x1, aes(label = Name), size = 3, nudge_x = 0.5) +
  facet_wrap(Constant+Meaning ~ ., nrow = 1, scales = "free") +
  theme_AGL +
  theme(strip.text = element_text(face = "italic"),
        legend.margin     = unit(c(0,0,0,0), "cm"),
        plot.title = element_text(hjust = -0.075),
        legend.position = "none") +
  scale_fill_manual(name = NULL, values = colors) +
  guides(fill = F) +
  labs(title = "A)", y = "x 10000")
#ggsave("Additional/Temp/Temp_F06_1.png", mp1, width = 6, height = 2.5)
# Prep data
xx <- read.csv("data/data_Tf_Pf.csv") %>%
  left_join(pca, by = "Entry") %>%
  mutate(Expt = factor(Expt, levels = names_Expt)) %>%
  select(Entry, Name, Expt, ExptShort, Cluster, Tb, Pc, predicted_Tf, predicted_Pf)
x1 <- xx %>%
  filter(ExptShort %in% c("Su17", "Ba17", "It17")) %>%
  group_by(Entry, Name, Expt, ExptShort, Cluster) %>%
  summarise_at(vars(Tb, Pc), funs(mean), na.rm = T)
# Plot A) Tb
mp2.1 <- ggplot(x1, aes(x = 1, y = Tb)) +
  geom_violin(fill = "grey", alpha = 0.3, color = NA) +
  geom_quasirandom(aes(color = Cluster), size = 0.3) +
  facet_grid(. ~ Expt, labeller = label_wrap_gen(width = 17)) +
  scale_y_continuous(breaks = seq(-80,0,20), minor_breaks = seq(-110,0,10)) +
  scale_color_manual(values = colors) +
  theme_AGL +
  theme(axis.text.x        = element_blank(),
        axis.ticks.x       = element_blank(),
        panel.grid.minor.x = element_blank(),
        plot.title = element_text(hjust = -0.16)) +
  guides(colour = guide_legend(nrow = 1, override.aes = list(size=2))) +
  labs(title = "B)", x = NULL,
```

```r
      y = expression(paste(italic("T")[italic("b")], " = -(", italic("a"), " + ",
                             italic("cP"), ") / ", italic("b"))))
# Plot B) Pc
x2 <- x1 %>% filter(Entry %in% c(94,105)) %>%
  ungroup() %>% mutate(Name = gsub(" AGL", "", Name))
mp2.2 <- ggplot(x1, aes(x = 1, y = Pc)) +
  geom_violin(fill = "grey", alpha = 0.3, color = NA) +
  geom_quasirandom(aes(color = Cluster), size = 0.3) +
  facet_grid(. ~ Expt, labeller = label_wrap_gen(width = 17)) +
  geom_text(data = x2, aes(label = Name), size = 3, nudge_y = 1.2) +
  scale_y_continuous(breaks = c(-20,-15,-10,-5,0,5)) +
  scale_color_manual(values = colors) +
  theme_AGL +
  theme(axis.text.x        = element_blank(),
        axis.ticks.x       = element_blank(),
        panel.grid.minor.x = element_blank(),
        plot.title = element_text(hjust = -0.16)) +
  guides(colour = guide_legend(nrow = 1, override.aes = list(size=2))) +
  labs(title = "C)", x = NULL,
      y = expression(paste(italic("P")[italic("c")], " = -(", italic("a"), " + ",
                             italic("bT"), ") / ", italic("c"))))
mp2 <- ggarrange(mp2.1, mp2.2, nrow = 1, ncol = 2, common.legend = T, legend = "bottom")
#ggsave("Additional/Temp/Temp_F06_2.png", mp2, width = 7, height = 5)
# Append ALL
mp <- ggarrange(mp1, mp2, nrow = 2, ncol = 1, align = "hv") #heights = c(1,1.75),
#mp1 <- ggarrange(mp1, mp2, nrow = 2, ncol = 1, heights = c(1,1.75), align = "v")
#mp2 <- ggarrange(mp3, mp4, nrow = 1, ncol = 2, common.legend = T, legend = "bottom")
#mp <- ggarrange(mp1, mp2, nrow = 2, ncol = 1, heights = c(2,1.25))
ggsave("Figure_05_TbPc.png", mp, width = 8, height = 6)
```



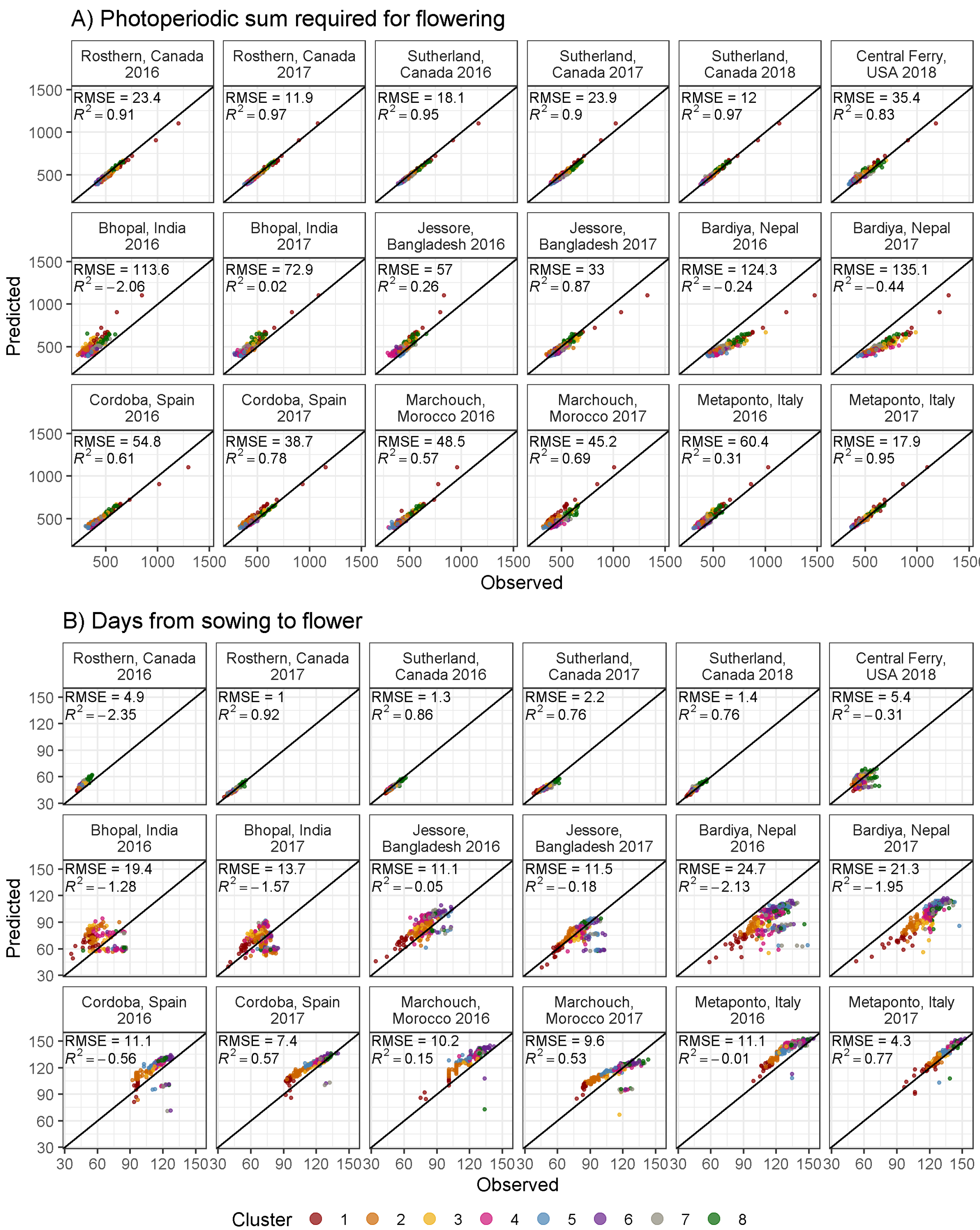
Supplemental Figure 9 Thermal Sums

```r
# Prep data
xx <- read.csv("data/data_Tf_Pf.csv") %>% #select(-MacroEnv) %>%
  mutate(Expt = factor(Expt, levels = names_Expt))
# Plot A)
mp1 <- gg_model_2(xx, "Tf", "predicted_Tf", "A) Thermal sum required for flowering",
                  200, 200, 6600, 5500)
# Plot B)
mp2 <- gg_model_2(xx, "DTF", "predicted_DTF_Tf", "B) Days from sowing to flower",
                  30, 30, 145, 125)
# Append A) and B)
mp <- ggarrange(mp1, mp2, nrow = 2, ncol = 1, common.legend = T, legend = "bottom")
ggsave("Supplemental_Figure_09_Tf.png", mp, width = 8, height = 10)
```


A) Thermal sum required for flowering
Rosthern, Canada 2016
RMSE = 105.6
$R^2 = 0.98$
Rosthern, Canada 2017
RMSE = 60.6
$R^2 = 0.99$
Sutherland, Canada 2016
RMSE = 141.4
$R^2 = 0.98$
Sutherland, Canada 2017
RMSE = 155.4
$R^2 = 0.97$
Sutherland, Canada 2018
RMSE = 70.1
$R^2 = 0.99$
Central Ferry, USA 2018
RMSE = 169.7
$R^2 = 0.95$
Bhopal, India 2016
RMSE = 583.5
$R^2 = 0.42$
Bhopal, India 2017
RMSE = 446.6
$R^2 = 0.55$
Jessore, Bangladesh 2016
RMSE = 263.8
$R^2 = 0.91$
Jessore, Bangladesh 2017
RMSE = 189.1
$R^2 = 0.96$
Bardiya, Nepal 2016
RMSE = 721.5
$R^2 = 0.56$
Bardiya, Nepal 2017
RMSE = 747.9
$R^2 = 0.4$
Cordoba, Spain 2016
RMSE = 227.9
$R^2 = 0.91$
Cordoba, Spain 2017
RMSE = 84.7
$R^2 = 0.99$
Marchouch, Morocco 2016
RMSE = 148.7
$R^2 = 0.97$
Marchouch, Morocco 2017
RMSE = 209.8
$R^2 = 0.95$
Metaponto, Italy 2016
RMSE = 107.3
$R^2 = 0.99$
Metaponto, Italy 2017
RMSE = 102.7
$R^2 = 0.99$
Predicted
2000 4000 6000
Observed
B) Days from sowing to flower
Rosthern, Canada 2016
RMSE = 5
$R^2 = -2.43$
Rosthern, Canada 2017
RMSE = 1.1
$R^2 = 0.9$
Sutherland, Canada 2016
RMSE = 1.8
$R^2 = 0.76$
Sutherland, Canada 2017
RMSE = 2.4
$R^2 = 0.73$
Sutherland, Canada 2018
RMSE = 1.4
$R^2 = 0.76$
Central Ferry, USA 2018
RMSE = 4.9
$R^2 = -0.07$
Bhopal, India 2016
RMSE = 19.4
$R^2 = -1.27$
Bhopal, India 2017
RMSE = 13.1
$R^2 = -1.38$
Jessore, Bangladesh 2016
RMSE = 11.4
$R^2 = -0.11$
Jessore, Bangladesh 2017
RMSE = 11.3
$R^2 = -0.15$
Bardiya, Nepal 2016
RMSE = 26.3
$R^2 = -2.55$
Bardiya, Nepal 2017
RMSE = 26.3
$R^2 = -3.51$
Cordoba, Spain 2016
RMSE = 9.9
$R^2 = -0.24$
Cordoba, Spain 2017
RMSE = 5
$R^2 = 0.81$
Marchouch, Morocco 2016
RMSE = 7.4
$R^2 = 0.55$
Marchouch, Morocco 2017
RMSE = 9.8
$R^2 = 0.5$
Metaponto, Italy 2016
RMSE = 5.8
$R^2 = 0.72$
Metaponto, Italy 2017
RMSE = 4.4
$R^2 = 0.77$
Predicted
30 60 90 120 150
Observed
Cluster 1 2 3 4 5 6 7 8

Supplemental Figure 10 Photoperiodic Sums

```r
# Prep data
xx <- read.csv("data/data_Tf_Pf.csv") %>% #select(-MacroEnv) %>%
  mutate(Expt = factor(Expt, levels = names_Expt))
# Plot A)
mp1 <- gg_model_2(xx, "Pf", "predicted_Pf", "A) Photoperiodic sum required for flowering",
                  190, 190, 1350, 1150)
# Plot B)
mp2 <- gg_model_2(xx, "DTF", "predicted_DTF_Pf", "B) Days from sowing to flower",
                  30, 30, 145, 125)
# Append A) and B)
mp <- ggarrange(mp1, mp2, nrow = 2, ncol = 1, common.legend = T, legend = "bottom")
ggsave("Supplemental_Figure_10_Pf.png", mp, width = 8, height = 10)
```



A) Photoperiodic sum required for flowering
Rosthern, Canada 2016
RMSE = 23.4
$R^2 = 0.91$
Rosthern, Canada 2017
RMSE = 11.9
$R^2 = 0.97$
Sutherland, Canada 2016
RMSE = 18.1
$R^2 = 0.95$
Sutherland, Canada 2017
RMSE = 23.9
$R^2 = 0.9$
Sutherland, Canada 2018
RMSE = 12
$R^2 = 0.97$
Central Ferry, USA 2018
RMSE = 35.4
$R^2 = 0.83$
Bhopal, India 2016
RMSE = 113.6
$R^2 = -2.06$
Bhopal, India 2017
RMSE = 72.9
$R^2 = 0.02$
Jessore, Bangladesh 2016
RMSE = 57
$R^2 = 0.26$
Jessore, Bangladesh 2017
RMSE = 33
$R^2 = 0.87$
Bardiya, Nepal 2016
RMSE = 124.3
$R^2 = -0.24$
Bardiya, Nepal 2017
RMSE = 135.1
$R^2 = -0.44$
Cordoba, Spain 2016
RMSE = 54.8
$R^2 = 0.61$
Cordoba, Spain 2017
RMSE = 38.7
$R^2 = 0.78$
Marchouch, Morocco 2016
RMSE = 48.5
$R^2 = 0.57$
Marchouch, Morocco 2017
RMSE = 45.2
$R^2 = 0.69$
Metaponto, Italy 2016
RMSE = 60.4
$R^2 = 0.31$
Metaponto, Italy 2017
RMSE = 17.9
$R^2 = 0.95$
Predicted
Observed
B) Days from sowing to flower
Rosthern, Canada 2016
RMSE = 4.9
$R^2 = -2.35$
Rosthern, Canada 2017
RMSE = 1
$R^2 = 0.92$
Sutherland, Canada 2016
RMSE = 1.3
$R^2 = 0.86$
Sutherland, Canada 2017
RMSE = 2.2
$R^2 = 0.76$
Sutherland, Canada 2018
RMSE = 1.4
$R^2 = 0.76$
Central Ferry, USA 2018
RMSE = 5.4
$R^2 = -0.31$
Bhopal, India 2016
RMSE = 19.4
$R^2 = -1.28$
Bhopal, India 2017
RMSE = 13.7
$R^2 = -1.57$
Jessore, Bangladesh 2016
RMSE = 11.1
$R^2 = -0.05$
Jessore, Bangladesh 2017
RMSE = 11.5
$R^2 = -0.18$
Bardiya, Nepal 2016
RMSE = 24.7
$R^2 = -2.13$
Bardiya, Nepal 2017
RMSE = 21.3
$R^2 = -1.95$
Cordoba, Spain 2016
RMSE = 11.1
$R^2 = -0.56$
Cordoba, Spain 2017
RMSE = 7.4
$R^2 = 0.57$
Marchouch, Morocco 2016
RMSE = 10.2
$R^2 = 0.15$
Marchouch, Morocco 2017
RMSE = 9.6
$R^2 = 0.53$
Metaponto, Italy 2016
RMSE = 11.1
$R^2 = -0.01$
Metaponto, Italy 2017
RMSE = 4.3
$R^2 = 0.77$
Predicted
Observed
Cluster 1 2 3 4 5 6 7 8

Supplemental Figure 11 Pc Tf PTT

```
# Prep data for A) Tf
pca <- read.csv("data/data_PCA_Results.csv") %>% select(Entry, Cluster) %>%
  mutate(Cluster = factor(Cluster))
xx <- read.csv("data/data_Tf_Pf.csv") %>%
  left_join(pca, by = "Entry") %>%
  mutate(MacroEnv = factor(MacroEnv, levels = macroEnvs))
x1 <- xx %>%
  select(Entry, Name, Expt, ExptShort, MacroEnv, Cluster, Tf_0, Tf_5, Tf) %>%
  gather(Trait, Value, Tf_0, Tf_5, Tf) %>%
  mutate(Trait = factor(Trait, levels = c("Tf_0", "Tf_5", "Tf")))
new.lab <- as_labeller(c(
  Tf_0 = "italic(T)[italic(b)]==0", Tf_5 = "italic(T)[italic(b)]==5",
  Tf = "italic(T)[italic(b)]==-(italic(a)+italic(Pc))/italic(b)",
  Mediterranean = "Mediterranean", Temperate = "Temperate",
  `South Asia` = "South~Asia"), label_parsed)
# Plot A) Tf
mp1 <- ggplot(x1, aes(x = ExptShort, y = Value)) +
  geom_violin(fill = "grey", alpha = 0.3, color = NA) +
  geom_quasirandom(aes(color = Cluster), size = 0.1, alpha = 0.7) +
  facet_grid(Trait ~ MacroEnv, scales = "free", labeller = new.lab) +
  scale_color_manual(values = colors) +
  theme_AGL +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  guides(colour = guide_legend(nrow = 1, override.aes = list(size = 3))) +
  labs(y = expression(italic("T")[italic("f")]), x = NULL)
# Prep data for B) Pf
x1 <- xx %>%
  select(Entry, Expt, ExptShort, MacroEnv, Cluster, Pf_0, Pf_7, Pf) %>%
  gather(Trait, Value, Pf_0, Pf_7, Pf) %>%
  mutate(Trait = factor(Trait, levels = c("Pf_0", "Pf_7", "Pf")))
new.lab <- as_labeller(c(
  Pf_0 = "italic(P)[italic(c)]==0", Pf_7 = "italic(P)[italic(c)]==5",
  Pf = "italic(P)[italic(c)]==-(italic(a)+italic(Tb))/italic(c)",
  Mediterranean = "Mediterranean", Temperate = "Temperate",
  `South Asia` = "South~Asia"), label_parsed)
# Plot B) Pf
mp2 <- ggplot(x1, aes(x = ExptShort, y = Value)) +
  geom_violin(fill = "grey", alpha = 0.3, color = NA) +
  geom_quasirandom( aes(color = Cluster), size = 0.1, alpha = 0.7) +
  facet_grid(Trait ~ MacroEnv, scales = "free", labeller = new.lab) +
  scale_color_manual(values = colors) +
  theme_AGL +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  guides(colour = guide_legend(nrow = 1, override.aes = list(size = 3))) +
  labs(y = expression(italic("P")[italic("f")]), x = NULL)
# Prep data for C) PTT
x1 <- xx %>%
  select(Entry, Expt, ExptShort, MacroEnv, Cluster, PTT_0, PTT) %>%
  gather(Trait, Value, PTT_0, PTT) %>%
```

```r
  mutate(Trait = factor(Trait, levels = c("PTT_0", "PTT")))
new.lab <- as_labeller(c(PTT_0 = "Sigma(italic(T))(italic(P))",
  PTT = "Sigma(italic(T)-italic(T)[italic(b)])(italic(P)-italic(P)[italic(c)])",
  Mediterranean = "Mediterranean", Temperate = "Temperate",
  `South Asia` = "South~Asia"), label_parsed)
# Plot C) PTT
mp3 <- ggplot(x1, aes(x = ExptShort, y = Value)) +
  geom_violin(fill = "grey", alpha = 0.3, color = NA) +
  geom_quasirandom(aes(color = Cluster), size = 0.1, alpha = 0.7) +
  facet_grid(Trait ~ MacroEnv, scales = "free", labeller = new.lab) +
  scale_color_manual(values = colors) +
  theme_AGL +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  guides(colour = guide_legend(nrow = 1, override.aes = list(size = 3))) +
  labs(y = expression(italic("PTT")), x = NULL)
# Append A) B) C)
mp <- ggarrange(mp1, mp2, mp3, ncol = 1, heights = c(3,3,2.2), align = "v",
                common.legend = T, legend = "bottom")
# Save
ggsave("Additional/Temp/Temp_SF11_1.png", mp1, width = 10, height = 13 * 3 / 8.2)
ggsave("Additional/Temp/Temp_SF11_2.png", mp2, width = 10, height = 13 * 3 / 8.2)
ggsave("Additional/Temp/Temp_SF11_3.png", mp3, width = 10, height = 13 * 3 / 8.2)
#
im1 <- image_read("Additional/Temp/Temp_SF11_1.png") %>%
  image_annotate("A)", size = 60, location = "+0+0")
im2 <- image_read("Additional/Temp/Temp_SF11_2.png") %>%
  image_annotate("B)", size = 60, location = "+0+0")
im3 <- image_read("Additional/Temp/Temp_SF11_3.png") %>%
  image_annotate("C)", size = 60, location = "+0+0")
im <- image_append(c(im1, im2, im3), stack = T)
image_write(im, "Supplemental_Figure_11_TfPcPTT.png")
#ggsave("Supplemental_Figure_09_TfPcPTT.png", mp, width = 10, height = 13)
```

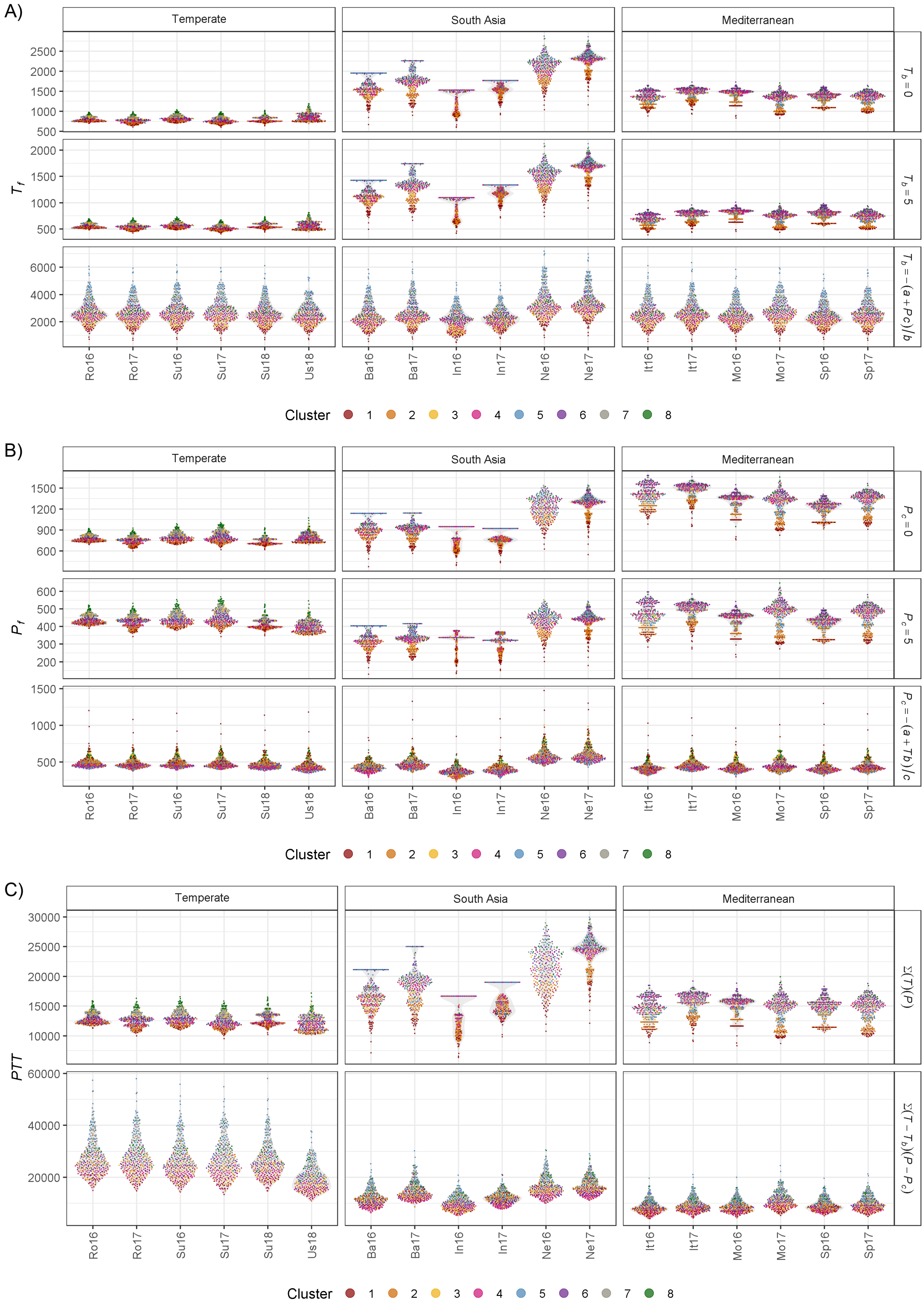

A)
Temperate
South Asia
Mediterranean
T_f
$T_b = 0$
$T_b = 5$
$T_b = -(a + Pc)/b$
Ro16
Ro17
Su16
Su17
Su18
Us18
Ba16
Ba17
In16
In17
Ne16
Ne17
It16
It17
Mo16
Mo17
Sp16
Sp17
Cluster 1 2 3 4 5 6 7 8
B)
P_f
$P_c = 0$
$P_c = 5$
$P_c = -(a + Tb)/c$
C)
PTT
$\Sigma(T)(P)$
$\Sigma(T - T_b)(P - P_c)$

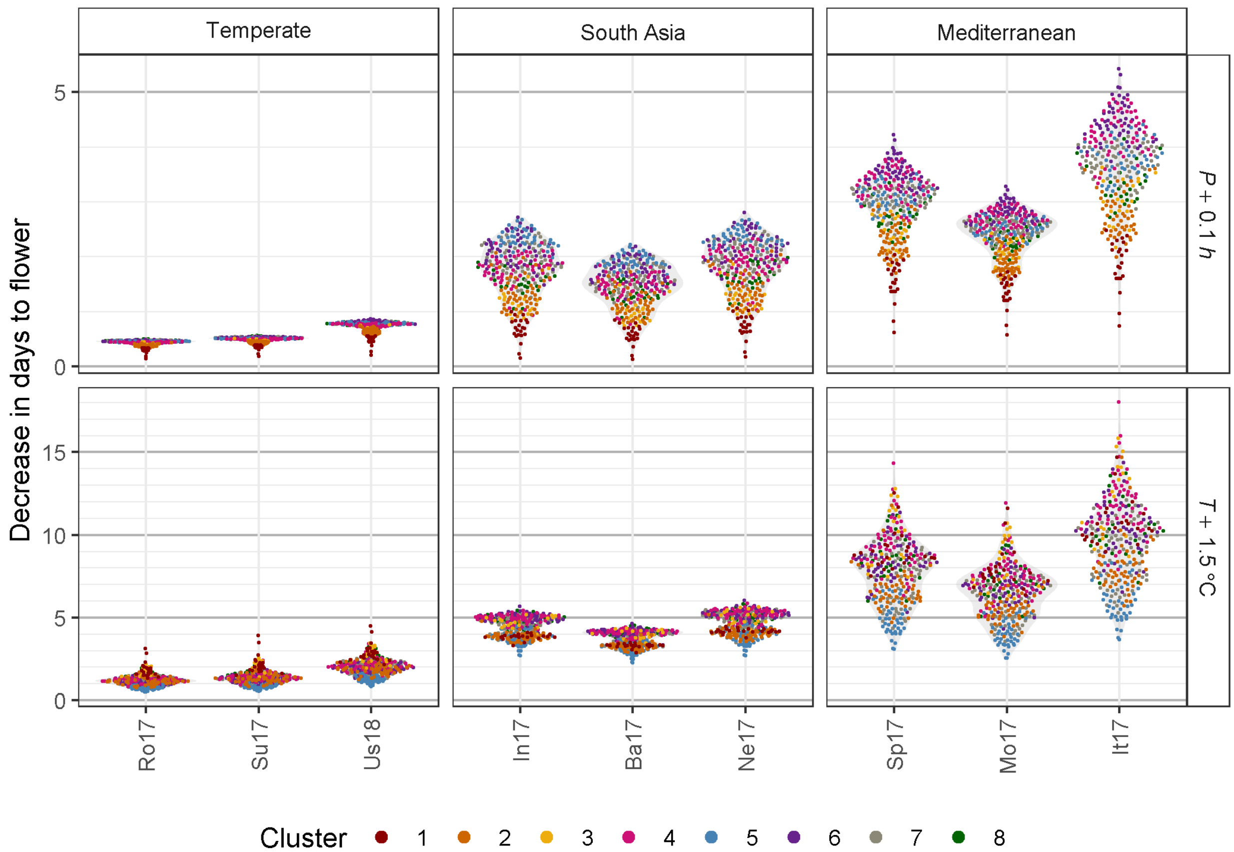
Figure 6 Temperature Increase By MacroEnv

```r
# Prep data
yy <- c("Ro17", "Su17", "Us18", "In17", "Ba17", "Ne17", "Sp17", "Mo17", "It17")
coefs <- read.csv("data/model_T+P_coefs.csv")
pca <- read.csv("data/data_PCA_Results.csv") %>% select(Entry, Cluster) %>%
  mutate(Cluster = factor(Cluster))
xx <- dd %>%
  select(Entry, Expt, ExptShort, DTF) %>%
  left_join(coefs, by = "Entry") %>%
  left_join(pca, by = "Entry") %>%
  left_join(select(ff, Expt, MacroEnv, T_mean, P_mean), by = "Expt")
# Temp +1
x1 <- xx %>%
  mutate(T_mean2 = T_mean + 1.5,
         DTF_1 = 1 / (a + b * T_mean2 + c * P_mean),
         DTF_0  = 1 / (a + b * T_mean + c * P_mean),
         Diff = DTF_0 - DTF_1) %>%
  filter(ExptShort %in% yy)
x2 <- xx %>%
  mutate(P_mean2 = P_mean + 0.1,
         DTF_1 = 1 / (a + b * T_mean + c * P_mean2),
         DTF_0  = 1 / (a + b * T_mean + c * P_mean),
         Diff = DTF_0 - DTF_1) %>%
  filter(ExptShort %in% yy)
x1 <- x1 %>% mutate(Increase = "T")
x2 <- x2 %>% mutate(Increase = "P")
knitr::kable(x2 %>% group_by(MacroEnv) %>%
               summarise(Min = round(min(Diff),2), Max = round(max(Diff),2)) )
```

MacroEnv	Min	Max
Temperate	0.14	0.87
South Asia	0.13	2.80
Mediterranean	0.57	5.42

```r
xx <- bind_rows(x1, x2)
write.csv(xx, "data/data_Temp_Increase.csv", row.names = F)
new.lab <- as_labeller(c(T = "italic(T)~+~1.5~degree*C", P = "italic(P)~+~0.1~italic(h)",
    Mediterranean = "Mediterranean", Temperate = "Temperate",
    `South Asia` = "South~Asia"), label_parsed)
# Plot
mp <- ggplot(xx, aes(x = ExptShort, y = Diff)) +
  geom_violin(fill = "grey", alpha = 0.3, color = NA) +
  geom_quasirandom(aes(color = Cluster), size = 0.3) +
  facet_grid(Increase ~ MacroEnv, scales = "free", labeller = new.lab) +
  scale_y_continuous(minor_breaks = seq(1,30,1), breaks = seq(0,30,5)) +
  scale_color_manual(values = colors) +
```

```r
  theme_AGL +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5),
        panel.grid.major.y = element_line(colour = "grey70", size = 0.5)) +
  guides(colour = guide_legend(nrow = 1, override.aes = list(size=2))) +
  labs(y = "Decrease in days to flower", x = NULL)
ggsave("Figure_06_TempIncrease.png", mp, width = 7, height = 5)
```



Additional Figure 12 Temperature Increase By Cluster

```r
x1 <- read.csv("data/data_Temp_Increase.csv")
x2 <- select(read.csv("data/data_PCA_Results.csv"), Entry, Cluster)
xx <- left_join(x1, x2, by = "Entry") %>%
  left_join(pca, by = "Entry") %>%
  mutate(Cluster = factor(Cluster)) %>%
  filter(ExptShort %in% c("Su17", "It17", "Ba17"))
mp <- ggplot(xx, aes(x = Cluster, y = Diff, fill = Cluster)) +
  geom_violin(color = NA, alpha = 0.6) +
  geom_quasirandom(color = "Black", alpha = 0.6) +
```

```r
  facet_grid(Expt ~ ., scales = "free") +
  theme_AGL +
  theme(legend.position = "none") +
  scale_fill_manual(name = NULL, values = colors) +
  labs(y = "Decrease in days to flower")
ggsave("Additional/Additional_Figure_12_TempIncreaseByCluster.png", mp, width = 7, height = 6)
```

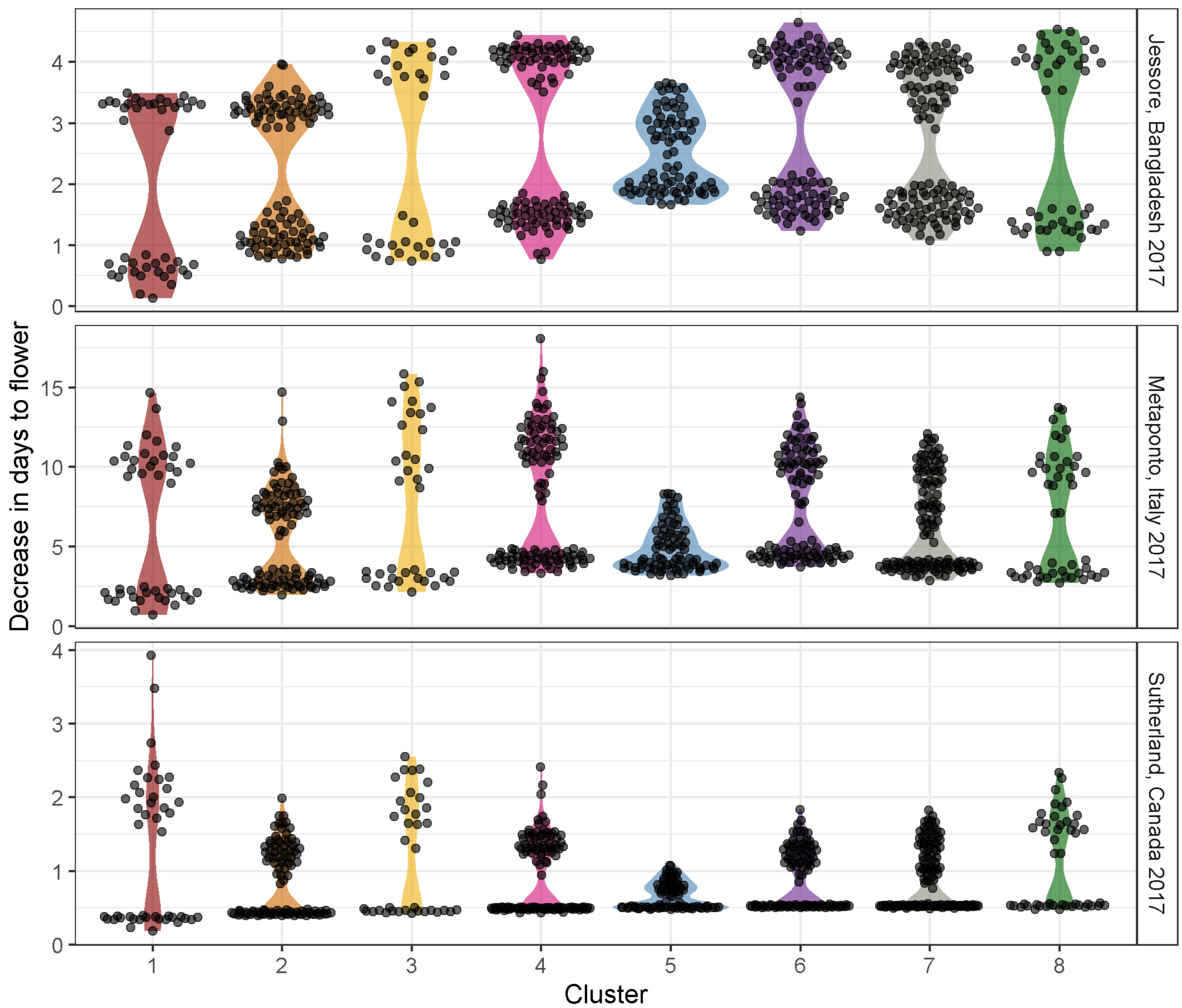
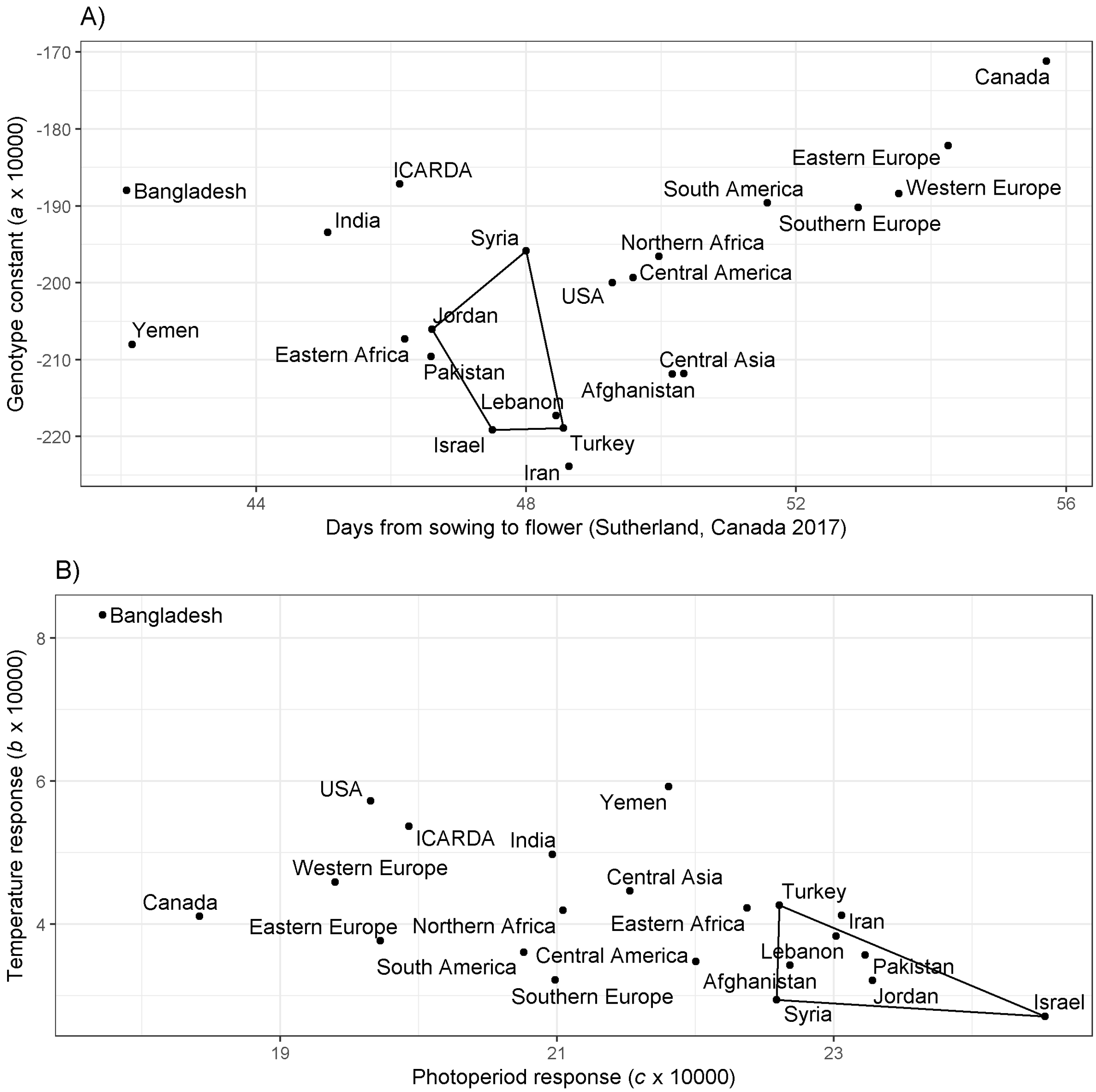


Figure 7 Origin Constants

```r
# Prep data
myctts <- c("Canada", "USA", "Iran", "Yemen",
            "India", "Pakistan", "Bangladesh", "Afghanistan",
            "Syria", "Jordan", "Turkey", "Lebanon", "Israel")
xx <- read.csv("data/model_T+P_coefs.csv") %>%
```

```r
  left_join(select(ldp, Entry, Origin), by = "Entry") %>%
  left_join(select(ct, Origin=Country, SubRegion), by = "Origin") %>%
  mutate(SubRegion = ifelse(Origin == "ICARDA", "ICARDA", as.character(SubRegion)),
         Origin = ifelse(Origin %in% mycts, Origin, as.character(SubRegion)))
x1 <- xx %>%
  left_join(dd %>% filter(ExptShort == "Su17") %>% select(Entry, DTF), by = "Entry") %>%
  group_by(Origin) %>%
  summarise_at(vars(DTF, a, b, c), funs(mean, sd)) %>%
  filter(Origin != "Unknown")
x2 <- x1 %>% mutate(CO = 1) %>%
  filter(Origin %in% c("Syria", "Jordan", "Turkey", "Lebanon", "Israel"))
# Plot A) a vs DTF
find_hull <- function(df) df[chull(df[,"DTF_mean"], df[,"a_mean"]), ]
polys <- plyr::ddply(x2, "CO", find_hull)
mp1 <- ggplot(x1, aes(x = DTF_mean, y = a_mean * 10000)) +
  geom_polygon(data = polys, fill = NA, color = "black") +
  geom_point() + geom_text_repel(aes(label = Origin)) +
  theme_AGL +
  labs(title = "A)",
       y = expression(paste("Genotype constant (", italic(a)," x 10000)")),
       x = "Days from sowing to flower (Sutherland, Canada 2017)")
# Plot B) b vs c
find_hull <- function(df) df[chull(df[,"c_mean"], df[,"b_mean"]), ]
polys <- plyr::ddply(x2, "CO", find_hull)
mp2 <- ggplot(x1, aes(x = c_mean * 10000, y = b_mean * 10000)) +
  geom_polygon(data = polys, fill = NA, color = "black") +
  geom_point() + geom_text_repel(aes(label = Origin)) +
  theme_AGL +
  labs(title = "B)",
       y = expression(paste("Temperature response (", italic(b), " x 10000)")),
       x = expression(paste("Photoperiod response (", italic(c), " x 10000)")))
# Append A) and B)
mp <- ggarrange(mp1, mp2, ncol = 1, nrow = 2)
ggsave("Figure_07_OriginCoefficients.png", mp, width = 8, height = 8)
ggsave("Additional/Temp/Temp_F07_1.png", mp1, width = 8, height = 4)
ggsave("Additional/Temp/Temp_F07_2.png", mp2, width = 8, height = 4)
```



A)
Genotype constant (a x 10000)
Days from sowing to flower (Sutherland, Canada 2017)
Canada
Eastern Europe
Western Europe
South America
Southern Europe
ICARDA
Bangladesh
India
Syria
Northern Africa
Central America
USA
Jordan
Yemen
Eastern Africa
Pakistan
Central Asia
Afghanistan
Lebanon
Israel
Turkey
Iran
B)
Temperature response (b x 10000)
Photoperiod response (c x 10000)
Bangladesh
USA
Yemen
ICARDA
India
Western Europe
Central Asia
Canada
Turkey
Eastern Europe
Northern Africa
Eastern Africa
Iran
Lebanon
Central America
Pakistan
South America
Afghanistan
Southern Europe
Jordan
Israel
Syria