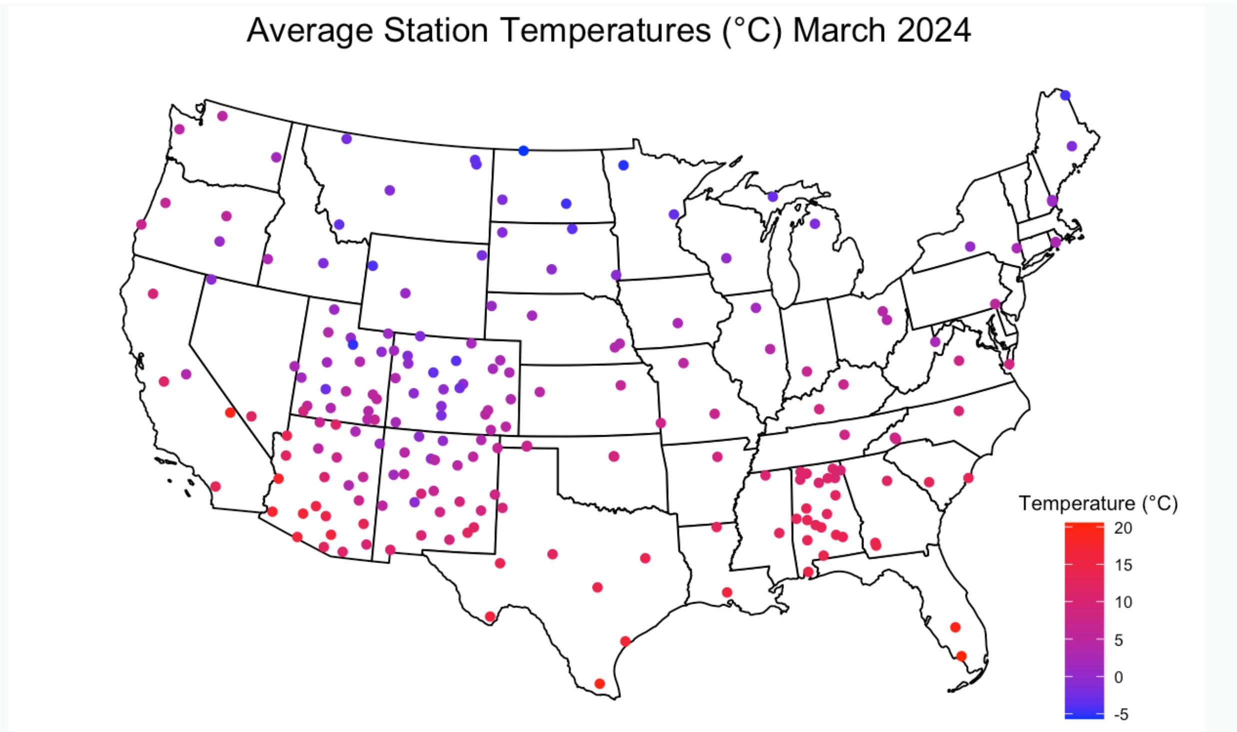
ClimateSCFP Vignette

```
library(ClimateSCFP)
```

Objective-1: Plot of Average Temperatures for March 2024

```
# extract station data
stat_ids <- station_data$station_id
stat_lons <- station_data$LONGITUDE
stat_lats <- station_data$LATITUDE
n <- length(stat_ids)

# make average temperature vector & setup dates
avg_temps <- rep(NA, n)
start_d <- as.Date("2024-03-01") |> lubridate::yday()
end_d <- as.Date("2024-03-31") |> lubridate::yday()
for(i in 1:n){
  wbanno <- stat_ids[i]
  station <- estimate_yearly_cycle(weather_data, wbanno)
  avg_temps[i] <- mean(station[,2][start_d:end_d], na.rm=T)
}

# plot

# make title for plot
t <- "Average Station Temperatures (°C) March 2024"

# make plotting data frame
g <- cbind.data.frame(stat_lons, stat_lats)
colnames(g) <- c("lon", "lat")
contig_usa <- maps::map("usa", "main", exact=T, plot=F)
pts_poly <- sp::point.in.polygon(g$lon, g$lat, contig_usa$x, contig_usa$y)
keep <- which(pts_poly == 1)
img.mat <- cbind.data.frame(g, avg_temps)
img.mat <- img.mat[keep, ]
img.mat <- na.omit(img.mat)
colnames(img.mat) <- c("lon", "lat", "temp")

# get United States data
usm.img.mat <- usmap::usmap_transform(img.mat)


# plot data
usmap::plot_usmap("states", exclude=c("AK", "HI")) +
  ggplot2::geom_sf(data=usm.img.mat, ggplot2::aes(colour=temp)) +
  ggplot2::coord_sf() + ggplot2::ggtitle(t)+
  ggplot2::scale_color_gradient(low="blue", high="red",
                                  name="Temperature (°C)") +
```

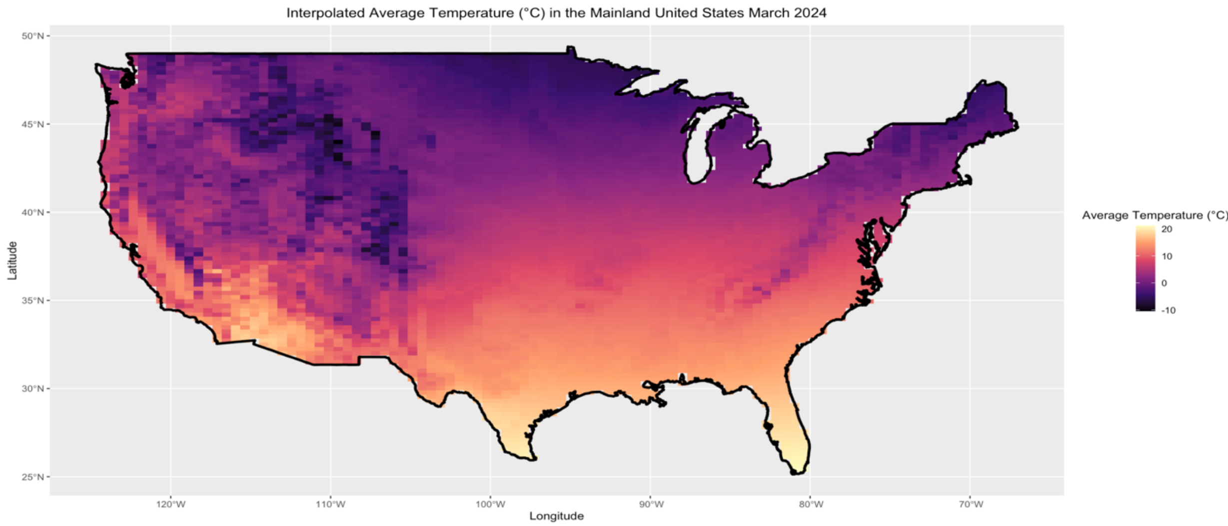
```
  ggplot2::theme(legend.title = ggplot2::element_text(hjust=0.5),
                 legend.position.inside=c(0.87,0.001),
                 plot.title=ggplot2::element_text(color="black", hjust=0.5,
                                                  size=15))
```



Objective-2: Interpolate Average Temperatures for March 2024

```
# make station data on contigous USA
HI_inds <- which(station_data$state == "HI") # Hawaii indexes
AK_inds <- which(station_data$state == "AK") # Alaska indexes
joint_inds <- c(AK_inds, HI_inds)
contig_data <- station_data[-joint_inds,]

# extract station data
stat_ids <- contig_data$station_id
stat_lons <- contig_data$LONGITUDE
stat_lats <- contig_data$LATITUDE
n <- length(stat_ids)

# make average temperature vector & setup dates
avg_temps <- rep(NA, n)
start_d <- as.Date("2024-03-01") |> lubridate::yday()
end_d <- as.Date("2024-03-31") |> lubridate::yday()
for(i in 1:n){
  wbanno <- stat_ids[i]
  station <- estimate_yearly_cycle(weather_data, wbanno)
  avg_temps[i] <- mean(station[,2][start_d:end_d], na.rm=T)
}
```

```r
# make data frame for temperature + spatial data
WD <- cbind.data.frame(lon=stat_lons, lat=stat_lats, avg_temp=avg_temps)
WD <- na.omit(WD) # remove NA

# make spatial data frame
spat.df <- data.frame(x=WD$lon, y=WD$lat)

# make response vector
Y <- WD$avg_temp

# get elevation data for model
prj_code <- 4326 # projection code
el.mod <- elevatr::get_elev_point(spat.df, prj=prj_code, src="aws")
#> Mosaicing & Projecting
#> Note: Elevation units are in meters
spat.df$elev <- el.mod$elevation

# make design matrix for model
X_mod <- model.matrix(~1+x+y+elev, data=spat.df)

# get grid points in contiguous USA + elevation data
g <- create_grid_points(resolution=100)
colnames(g) <- c("x", "y") # change lon to x and lat to y for elavatr
el.interp <- elevatr::get_elev_point(g, prj=prj_code, src="aws")
#> Mosaicing & Projecting
#> Note: Elevation units are in meters
g$elev <- el.interp$elevation

# make design matrix for interpolations
X_interp <- model.matrix(~1+x+y+elev, data=g)

# make interpolations
g <- g[,1:2]
interps <- interpolate_points(WD, Y, X_mod, g, X_interp)
#> Assuming columns 1 and 2 of locs are (longitude,latidue) in degrees

# make title for plot & plot
t <- paste0("Interpolated Average Temperature (°C)",
      " in the Mainland United States March 2024")

int.plot <- plot_gridded_interpolations(interps, g, t=t)

# view plot
int.plot
```



Objective-3: Warmest and Coldest Days for USCRN Stations

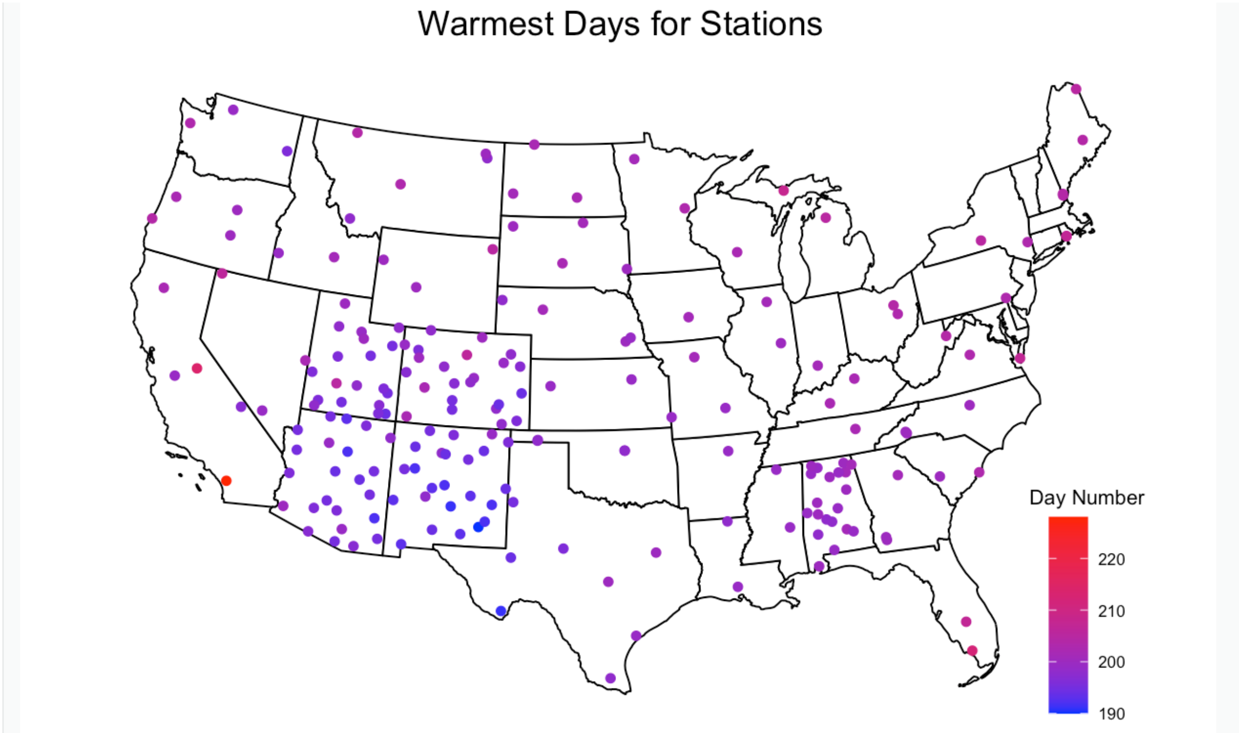
The model being assumed is depicted below:

$$T\_DAILY\_AVG_i = \beta_0 + \beta_1 cos(2\pi day_i/365.25) + \beta_2 sin(2\pi day_i/365.25) + \beta_3 cos^2(2\pi day_i/365.25) + \beta_4 cos\big((2\pi yday_i/365.25) + 1.27\big) + \epsilon_i$$

Where $\epsilon_i \sim Normal(0, \sigma^2)$, day_i is the number of days from the epoch of `2000-01-01` and $yday_i$ is the day number (1-366). The null hypothesis is H_0: β_i = 0,i.e., the days do not have any effect on the average temperature, and it will be assessed at a significance level of α = 0.05.

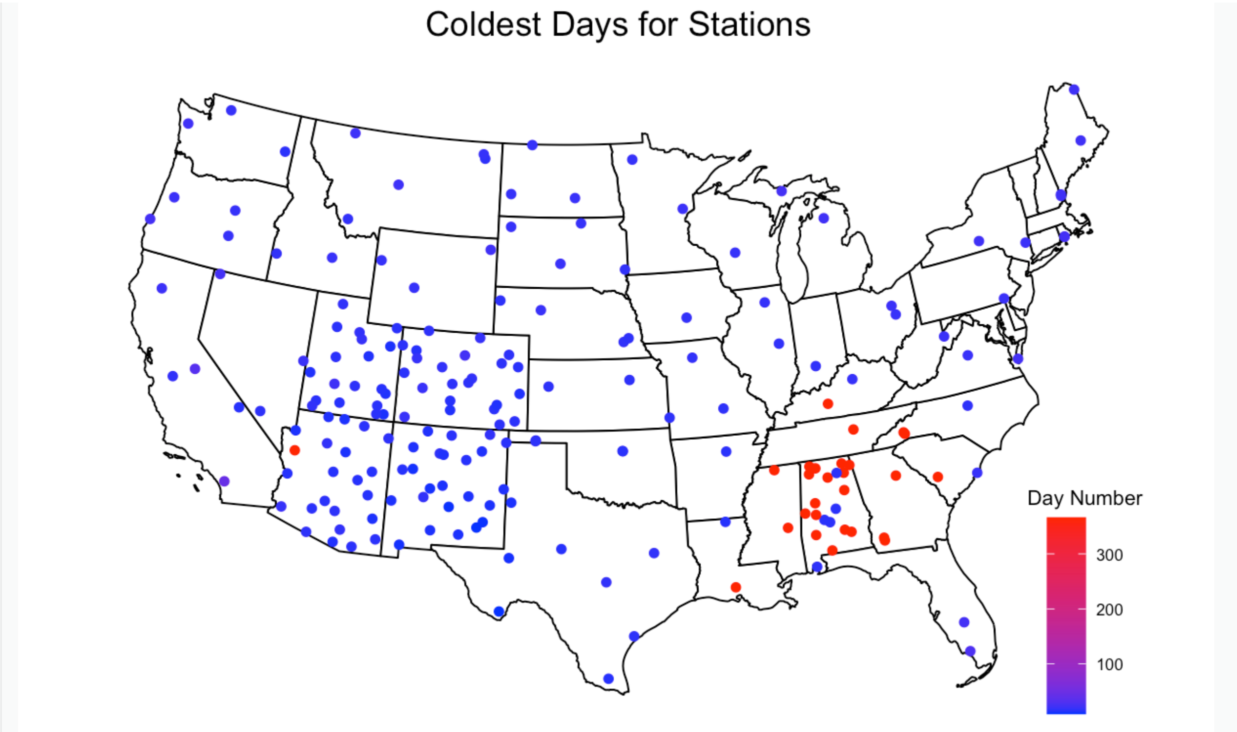
```
# make station data on contigous USA
HI_inds <- which(station_data$state == "HI") # Hawaii indexes
AK_inds <- which(station_data$state == "AK") # Alaska indexes
joint_inds <- c(AK_inds, HI_inds)
contig_data <- station_data[-joint_inds,]

# extract station data
stat_ids <- contig_data$station_id
stat_lons <- contig_data$LONGITUDE
stat_lats <- contig_data$LATITUDE
n <- length(stat_ids)

# make warm and cold day vectors
warm <- rep(NA, n)
cold <- rep(NA, n)

# get yearly cycles & population warm and cold vectors
for(i in 1:n){
```

```r
  wbanno <- stat_ids[i]
  station <- estimate_yearly_cycle(weather_data, wbanno)
  warm[i] <- which.max(station[,2])
  cold[i] <- which.min(station[,2])
}

# plot warm

# make title for plot
t <- "Warmest Days for Stations"

# make plotting data frame
g <- cbind.data.frame(stat_lons, stat_lats)
colnames(g) <- c("lon", "lat")
contig_usa <- maps::map("usa", "main", exact=T, plot=F)
pts_poly <- sp::point.in.polygon(g$lon, g$lat, contig_usa$x, contig_usa$y)
keep <- which(pts_poly == 1)
img.mat <- cbind.data.frame(g, warm)
img.mat <- img.mat[keep, ]
img.mat <- na.omit(img.mat)
colnames(img.mat) <- c("lon", "lat", "warm")

# get United States data
usm.img.mat <- usmap::usmap_transform(img.mat)


# plot data
usmap::plot_usmap("states", exclude=c("AK", "HI")) +
  ggplot2::geom_sf(data=usm.img.mat, ggplot2::aes(colour=warm)) +
  ggplot2::coord_sf() + ggplot2::ggtitle(t)+
  ggplot2::scale_color_gradient(low="blue", high="red",
                                name="Day Number") +
  ggplot2::theme(legend.title = ggplot2::element_text(hjust=0.5),
                 legend.position.inside=c(0.87,0.001),
                 plot.title=ggplot2::element_text(color="black", hjust=0.5,
                                                  size=15))
```



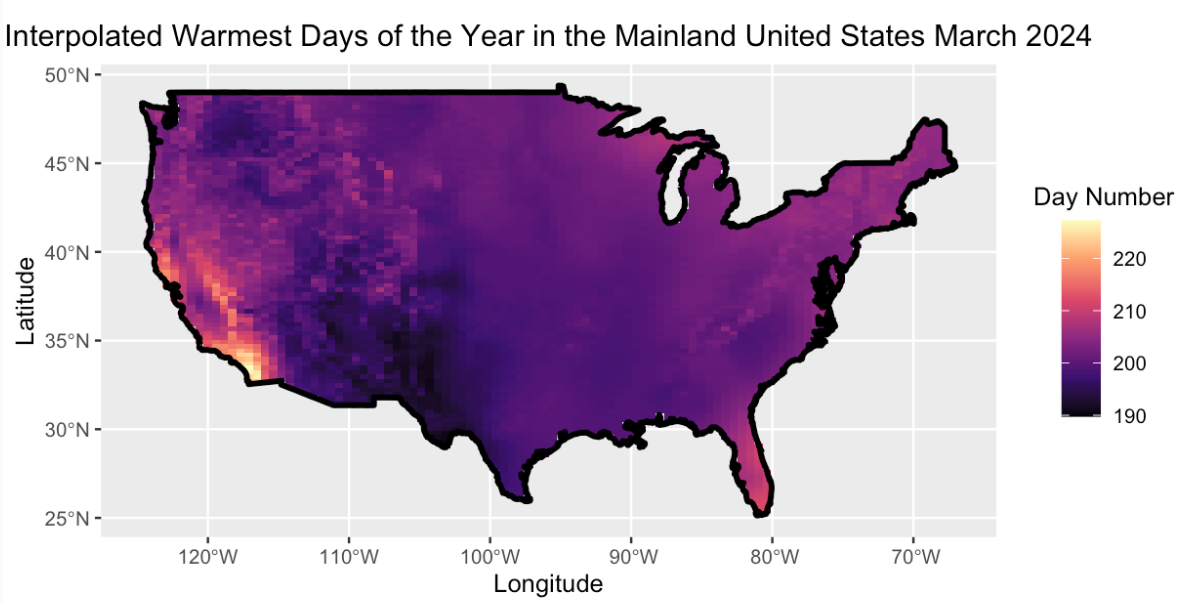

```
# plot cold

# make title for plot
t <- "Coldest Days for Stations"

# make plotting data frame
g <- cbind.data.frame(stat_lons, stat_lats)
colnames(g) <- c("lon", "lat")
contig_usa <- maps::map("usa", "main", exact=T, plot=F)
pts_poly <- sp::point.in.polygon(g$lon, g$lat, contig_usa$x, contig_usa$y)
keep <- which(pts_poly == 1)
img.mat <- cbind.data.frame(g, cold)
img.mat <- img.mat[keep, ]
img.mat <- na.omit(img.mat)
colnames(img.mat) <- c("lon", "lat", "cold")

# get United States data
usm.img.mat <- usmap::usmap_transform(img.mat)


# plot data
usmap::plot_usmap("states", exclude=c("AK", "HI")) +
  ggplot2::geom_sf(data=usm.img.mat, ggplot2::aes(colour=cold)) +
  ggplot2::coord_sf() + ggplot2::ggtitle(t)+
  ggplot2::scale_color_gradient(low="blue", high="red",
                                name="Day Number") +
  ggplot2::theme(legend.title = ggplot2::element_text(hjust=0.5),
```

```
                   legend.position.inside=c(0.87,0.001),
                   plot.title=ggplot2::element_text(color="black", hjust=0.5,
                                                    size=15))
```

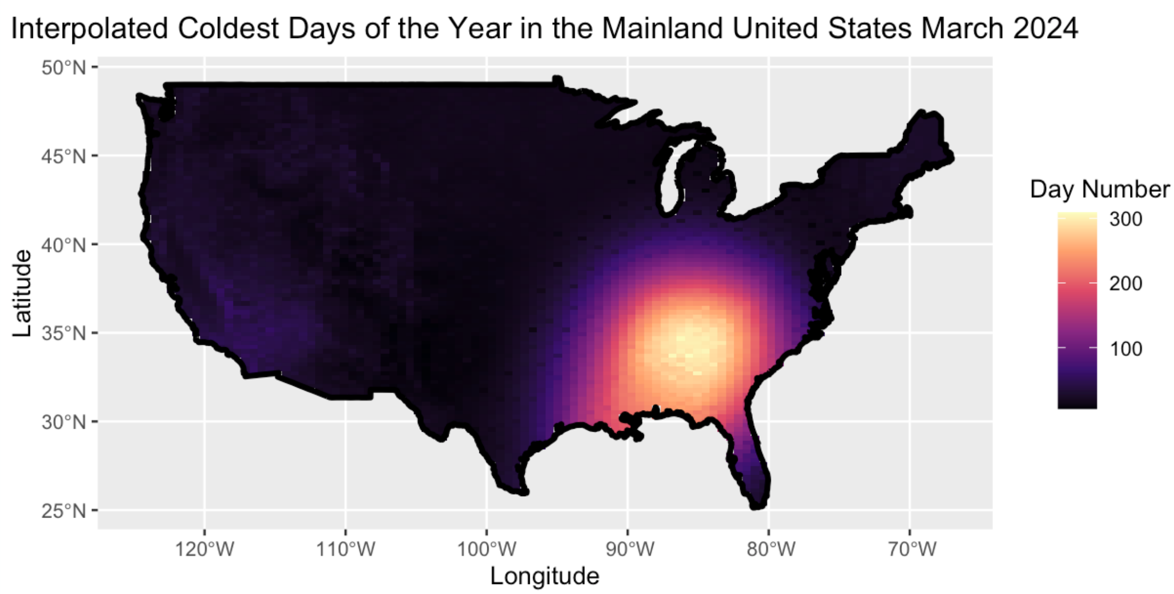



```
# interpolate & plot warm

# make data frame for day + spatial data
WD <- cbind.data.frame(lon=stat_lons, lat=stat_lats, warm=warm)
WD <- na.omit(WD) # remove NA

# make spatial data frame
spat.df <- data.frame(x=WD$lon, y=WD$lat)

# make response vector
Y <- WD$warm

# get elevation data for model
prj_code <- 4326 # projection code
el.mod <- elevatr::get_elev_point(spat.df, prj=prj_code, src="aws")
#> Mosaicing & Projecting
#> Note: Elevation units are in meters
spat.df$elev <- el.mod$elevation

# make design matrix for model
X_mod <- model.matrix(~1+x+y+elev, data=spat.df)
```

```r
# get grid points in contiguous USA + elevation data
g <- create_grid_points(resolution=100)
colnames(g) <- c("x", "y") # change lon to x and lat to y for elavatr
el.interp <- elevatr::get_elev_point(g, prj=prj_code, src="aws")
#> Mosaicing & Projecting
#> Note: Elevation units are in meters
g$elev <- el.interp$elevation

# make design matrix for interpolations
X_interp <- model.matrix(~1+x+y+elev, data=g)

# make interpolations
g <- g[,1:2]
interps <- interpolate_points(WD, Y, X_mod, g, X_interp)
#> Assuming columns 1 and 2 of locs are (longitude,latidue) in degrees

# make title for plot & plot
t <- paste0("Interpolated Warmest Days of the Year",
      " in the Mainland United States March 2024")
f <- "Day Number"

int.plot <- plot_gridded_interpolations(interps, g, t, f)

# view plot
int.plot
```



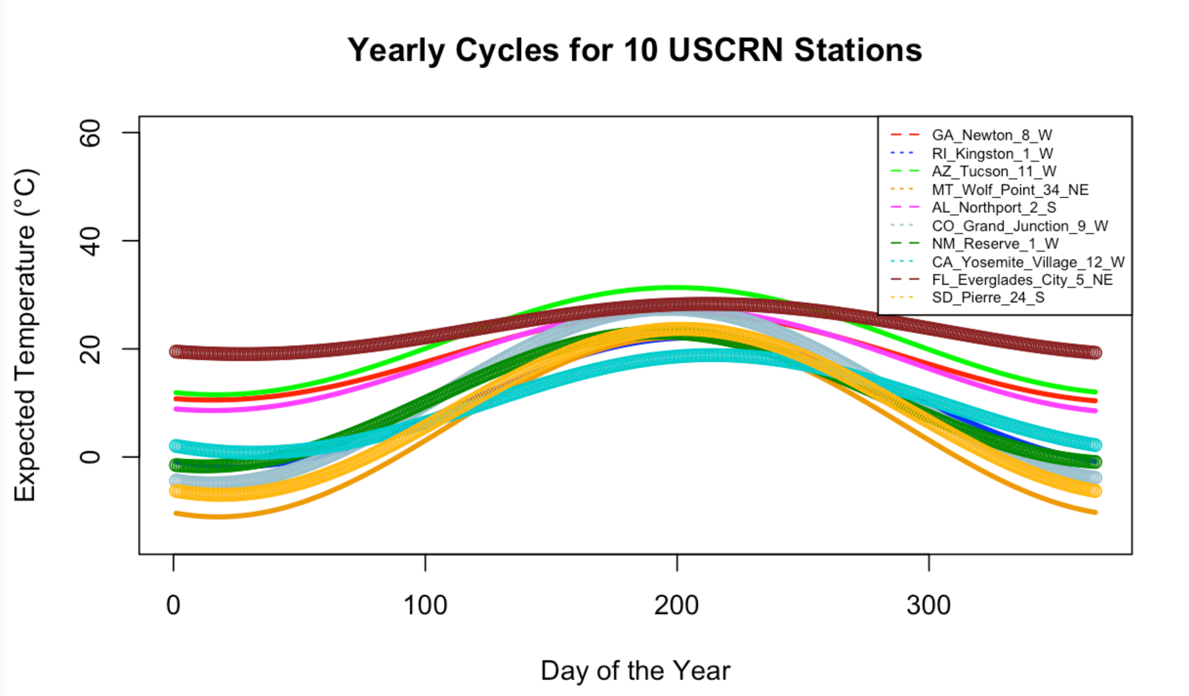
```r
# interpolate & plot cold

# make data frame for day + spatial data
WD <- cbind.data.frame(lon=stat_lons, lat=stat_lats, cold=cold)
WD <- na.omit(WD) # remove NA

# make spatial data frame
spat.df <- data.frame(x=WD$lon, y=WD$lat)

# make response vector
Y <- WD$cold

# get elevation data for model
prj_code <- 4326 # projection code
el.mod <- elevatr::get_elev_point(spat.df, prj=prj_code, src="aws")
#> Mosaicing & Projecting
#> Note: Elevation units are in meters
spat.df$elev <- el.mod$elevation

# make design matrix for model
X_mod <- model.matrix(~1+x+y+elev, data=spat.df)

# get grid points in contiguous USA + elevation data
g <- create_grid_points(resolution=100)
colnames(g) <- c("x", "y") # change lon to x and lat to y for elavatr
el.interp <- elevatr::get_elev_point(g, prj=prj_code, src="aws")
#> Mosaicing & Projecting
#> Note: Elevation units are in meters
g$elev <- el.interp$elevation

# make design matrix for interpolations
X_interp <- model.matrix(~1+x+y+elev, data=g)

# make interpolations
g <- g[,1:2]
interps <- interpolate_points(WD, Y, X_mod, g, X_interp)
#> Assuming columns 1 and 2 of locs are (longitude,latidue) in degrees

# make title for plot & plot
t <- paste0("Interpolated Coldest Days of the Year",
      " in the Mainland United States March 2024")
f <- "Day Number" # legend title

int.plot <- plot_gridded_interpolations(interps, g, t, f)

# view plot
int.plot
```



Objective-4: Yearly Cycles for 10 USCRN Stations

```
# get estimates for 10 diverse stations
GA_Newton_8_W <- estimate_yearly_cycle(weather_data, 63828)
RI_Kingston_1_W <- estimate_yearly_cycle(weather_data, 54797)
AZ_Tucson_11_W <- estimate_yearly_cycle(weather_data, 53131)
MT_Wolf_Point_34_NE <- estimate_yearly_cycle(weather_data, 94059)
AL_Northport_2_S <- estimate_yearly_cycle(weather_data, 73801)
CO_Grand_Junction_9_W <- estimate_yearly_cycle(weather_data, 3076)
NM_Reserve_1_W <- estimate_yearly_cycle(weather_data, 3080)
CA_Yosemite_Village_12_W <- estimate_yearly_cycle(weather_data, 53150)
FL_Everglades_City_5_NE <- estimate_yearly_cycle(weather_data, 92826)
SD_Pierre_24_S <- estimate_yearly_cycle(weather_data, 94085)

# plotting
x <- 1:366
plot(x, GA_Newton_8_W$EXPECTED_TEMP, type = "n",
     xlab="Day of the Year",
     ylab="Expected Temperature (°C)",
     main = "Yearly Cycles for 10 USCRN Stations",
     ylim=c(-15,60))
lines(x, GA_Newton_8_W$EXPECTED_TEMP, type = "l", col = "red", lw=3)
lines(x, RI_Kingston_1_W$EXPECTED_TEMP, type = "l", col = "blue", lw=3)
lines(x, AZ_Tucson_11_W$EXPECTED_TEMP, type = "l", col = "green", lw=3)
lines(x, MT_Wolf_Point_34_NE$EXPECTED_TEMP, type = "l", col = "orange2",
lw=3)
lines(x, AL_Northport_2_S$EXPECTED_TEMP, type = "l", col = "magenta", lw=3)
lines(x, CO_Grand_Junction_9_W$EXPECTED_TEMP, type = "b", col = "lightblue3")
```

```r
lines(x, NM_Reserve_1_W$EXPECTED_TEMP, type = "b", col = "green4")
lines(x, CA_Yosemite_Village_12_W$EXPECTED_TEMP, type = "b", col = "cyan3")
lines(x, FL_Everglades_City_5_NE$EXPECTED_TEMP, type = "b", col = "brown4")
lines(x, SD_Pierre_24_S$EXPECTED_TEMP, type = "b", col = "darkgoldenrod1")
par(xpd=TRUE)
legend("topright",legend = c("GA_Newton_8_W", "RI_Kingston_1_W",
                              "AZ_Tucson_11_W","MT_Wolf_Point_34_NE",
                              "AL_Northport_2_S","CO_Grand_Junction_9_W",
                              "NM_Reserve_1_W","CA_Yosemite_Village_12_W",
                              "FL_Everglades_City_5_NE","SD_Pierre_24_S"),
       col = c("red", "blue", "green","orange2","magenta",
               "lightblue3","green4","cyan3","brown4","darkgoldenrod1"),
       lty=c(2,3), cex=0.55)
```



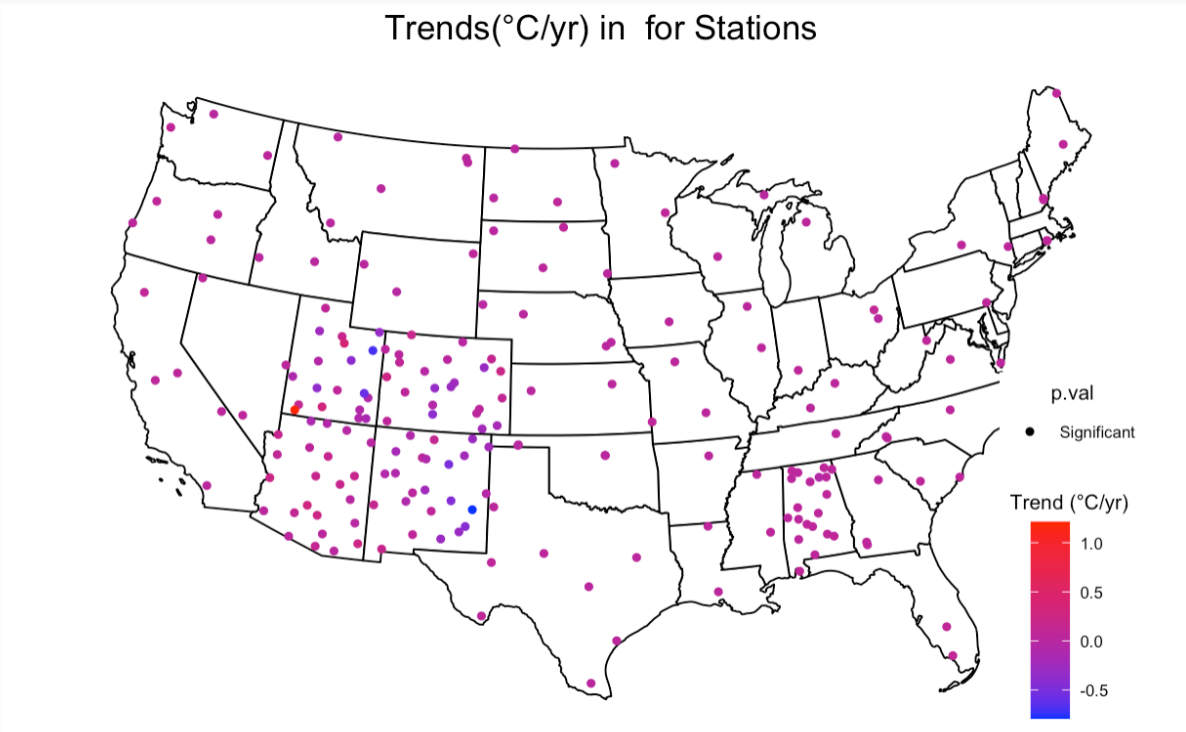
Objective-5: Estimating and Interpolating Station Trends

The model being assumed is depicted below: $T\_DAILY\_AVG_i = \beta_0 + \beta_1 day_i + \beta_2 cos(2\pi day_i/365.25) + \beta_3 sin(2\pi day_i/365.25) + \beta_4 cos^2(2\pi day_i/365.25) + \beta_5 cos\left((2\pi yday_i/365.25) + 1.27\right) + \beta_6 sin(2\pi yday_i/365.25) + \epsilon_i$ Where $\epsilon_i \sim Normal(0, \sigma^2)$, and day_i is the number of days from the epoch of `2000-01-01`, and $yday_i$ is the day number (1-366). The null hypothesis is H_0: $\beta_i = 0$, i.e., the days do not have any effect on the average temperature, and it will be assessed at a significance level of $\alpha = 0.05$.

```r
# make station data on contigous USA
HI_inds <- which(station_data$state == "HI") # Hawaii indexes
AK_inds <- which(station_data$state == "AK") # Alaska indexes
joint_inds <- c(AK_inds, HI_inds)
```

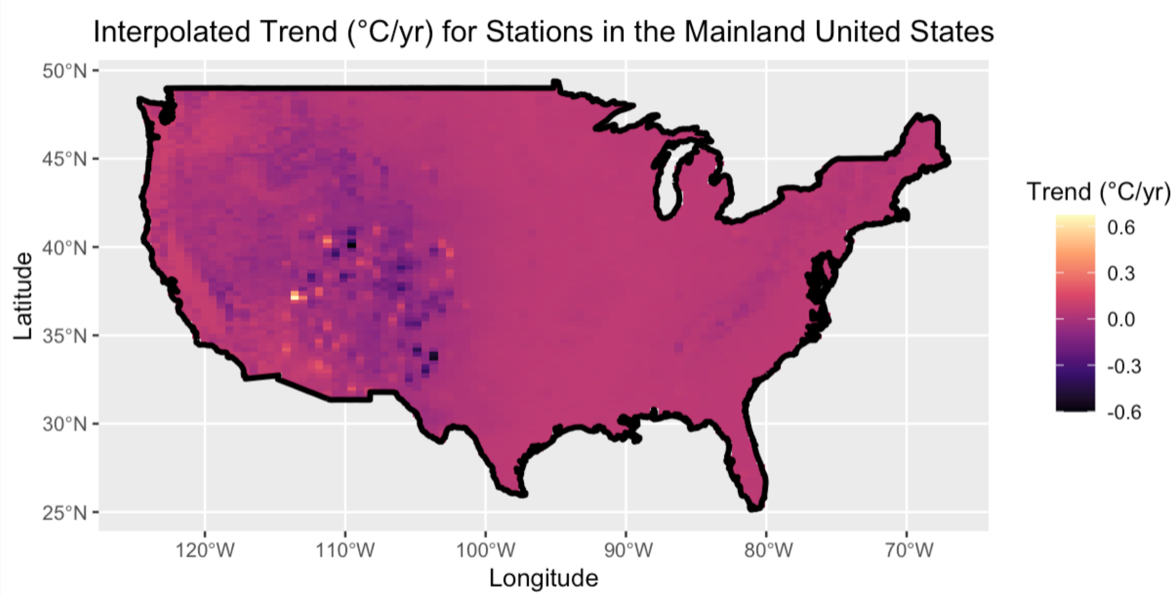
```r
contig_data <- station_data[-joint_inds,]

# extract station data
stat_ids <- contig_data$station_id
stat_lons <- contig_data$LONGITUDE
stat_lats <- contig_data$LATITUDE
n <- length(stat_ids)

# make trend vectors
trend <- rep(NA, n)
trend_pv <- rep(NA, n)
trend_se <- rep(NA, n)

# get yearly cycles & population trend vectors
for(i in 1:n){
  wbanno <- stat_ids[i]
  dat3 <- estimate_trends(weather_data, wbanno)
  trend[i] <- dat3[1]
  trend_pv[i] <- dat3[2]
  trend_se[i] <- dat3[3]
}

trend <- 365.25*trend # to make from day to year

# plot

# make title for plot
t <- "Trends(°C/yr) in  for Stations"

# get statistical significance
trend_pv <- ifelse(trend_pv < 0.05, "Significant", "Insignificant")

# make plotting data frame
g <- cbind.data.frame(stat_lons, stat_lats)
colnames(g) <- c("lon", "lat")
contig_usa <- maps::map("usa", "main", exact=T, plot=F)
pts_poly <- sp::point.in.polygon(g$lon, g$lat, contig_usa$x, contig_usa$y)
keep <- which(pts_poly == 1)
img.mat <- cbind.data.frame(g, trend, trend_pv, trend_se)
img.mat <- img.mat[keep, ]
img.mat <- na.omit(img.mat)
colnames(img.mat) <- c("lon", "lat", "beta", "p.val", "se")

# get United States data
usm.img.mat <- usmap::usmap_transform(img.mat)


# plot data
usmap::plot_usmap("states", exclude=c("AK", "HI")) +
```

```
  ggplot2::geom_sf(data=usm.img.mat, ggplot2::aes(colour=beta,shape=p.val)) +
  ggplot2::coord_sf() + ggplot2::ggtitle(t)+
  ggplot2::scale_color_gradient(low="blue", high="red",
                                name="Trend (°C/yr)") +
  ggplot2::theme(legend.title = ggplot2::element_text(hjust=0.5),
                 legend.position.inside=c(0.87,0.001),
                 plot.title=ggplot2::element_text(color="black", hjust=0.5,
                                                  size=15))
```




```
# interpolate & plot

# keep small se trends
keep.i <- which(trend_se < 1)
trend <- trend[keep.i]
stat_lons <- stat_lons[keep.i]
stat_lats <- stat_lats[keep.i]

# make data frame for day + spatial data
WD <- cbind.data.frame(lon=stat_lons, lat=stat_lats, tr=trend)
WD <- na.omit(WD) # remove NA

# make spatial data frame
spat.df <- data.frame(x=WD$lon, y=WD$lat)

# make response vector
Y <- WD$tr
```

```r
# get elevation data for model
prj_code <- 4326 # projection code
el.mod <- elevatr::get_elev_point(spat.df, prj=prj_code, src="aws")
#> Mosaicing & Projecting
#> Note: Elevation units are in meters
spat.df$elev <- el.mod$elevation

# make design matrix for model
X_mod <- model.matrix(~1+x+y+elev, data=spat.df)

# get grid points in contiguous USA + elevation data
g <- create_grid_points(resolution=100)
colnames(g) <- c("x", "y") # change lon to x and lat to y for elavatr
el.interp <- elevatr::get_elev_point(g, prj=prj_code, src="aws")
#> Mosaicing & Projecting
#> Note: Elevation units are in meters
g$elev <- el.interp$elevation

# make design matrix for interpolations
X_interp <- model.matrix(~1+x+y+elev, data=g)

# make interpolations
g <- g[,1:2]
interps <- interpolate_points(WD, Y, X_mod, g, X_interp)
#> Assuming columns 1 and 2 of locs are (longitude,latidue) in degrees

# make title for plot & plot
t <- paste0("Interpolated Trend (°C/yr) for Stations",
      " in the Mainland United States")
f <- "Trend (°C/yr)"

int.plot <- plot_gridded_interpolations(interps, g, t, f)

# view plot
int.plot
```



Objective-6: Validation with Reputable Sources

I found a research paper by Gil-Alana et al. 2022 which can be found at this https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9616705/ link. The authors claim that the overall temperature trend in the United States is approximately +1.60ºF/100 year which is approximately +0.71ºC/100 year. I obtained answers similar to this range ranging from -0.6ºC/year to +0.6ºC/year, however, the units are off since mine is estimated per year and theirs is the average estimated temperature trend over the past 100 years. My work is slightly off and is probably due to my models not accounting for the data correctly or being slightly insufficient.