# Data Prep - Summer Temperatures

## Libraries

```
library(sf)
library(tidyverse)
library(spmodel)
library(data.table)
library(ggplot2)
library(StreamCatTools)
library(tigris)
library(prism)
library(terra)
```

## Data Development

### Stream Temperature (st) Observations

- These data represent raw daily mean values from USGS loggers/stations
- We applied QA/QC process to flag and remove records that can represent a variety of issues (see QA documentation in SI)
- This code does the following:

1. Reads raw flagged data; removes those that are flagged to be removed
2. Calculates mean monthly values for July and August for sites with >20 days of record for those months
3. Converts data to simple feature spatial object

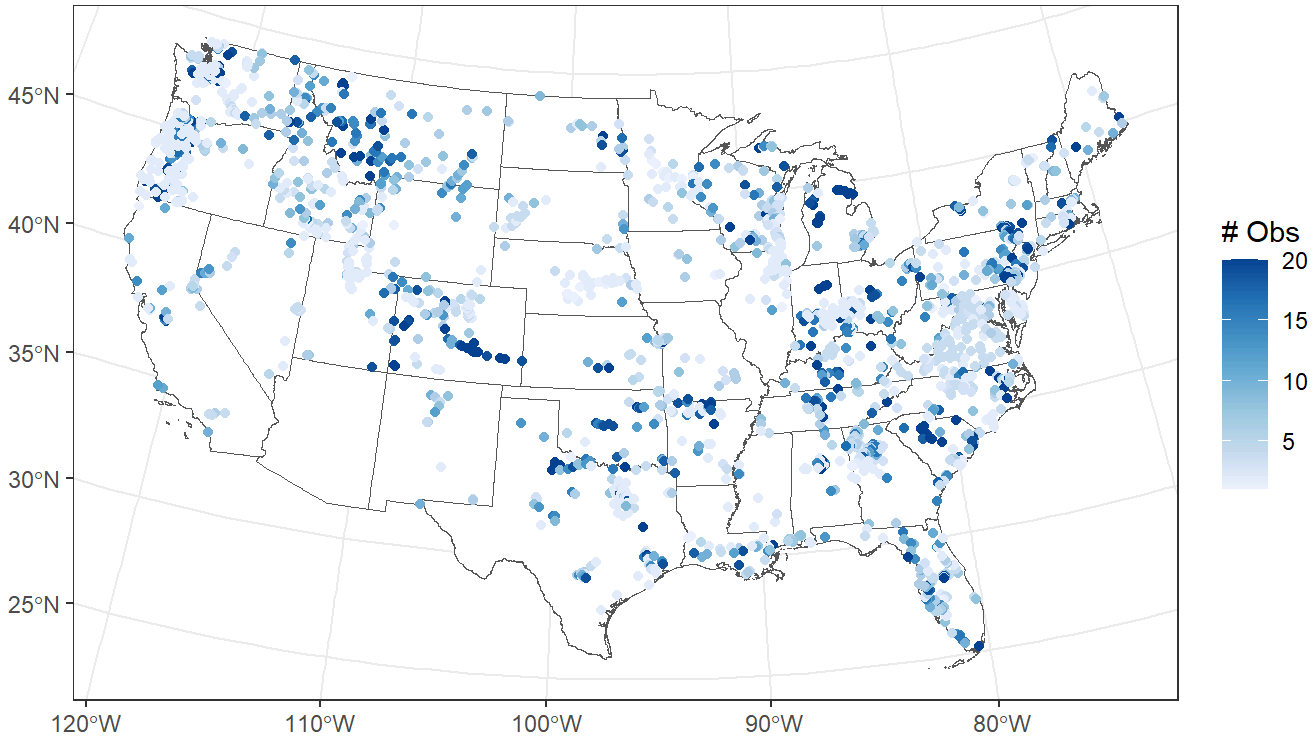
```
pts <- read_rds('../data/pts_sf.rds')

st <- read_rds('../data/stream_temperatures_raw_flagged.rds') %>%

  # Filter anything with remove flag
  filter(Flag_Remove == "") %>%

  # Add year and month
  mutate(date = lubridate::ymd(YYYYMMDD),
         year = year(date),
         month = month(date)) %>%

  # Select just necessary columns (DV_VALUE = daily temperatures)
  dplyr::select(SITECODE, DV_VALUE, year, month) %>%

  # Calc mean for year/month by sites
  group_by(SITECODE, year, month) %>%
```

```
  summarise(wtmp_mo = mean(DV_VALUE, na.rm = TRUE),
            count = n()) %>%

  # Filter to just July/August w/ greater than/equal 20 days
  filter(month == 7 | month == 8,
         count >= 20) %>%

  # Join to locations by ID
  left_join(pts, join_by(SITECODE)) %>%

  st_as_sf() %>%

  st_transform(crs = 5070)
```

## Map observed values

Map temperature sites and color by number of observations (months with data)

```
states <- tigris::states(cb = TRUE, progress_bar = FALSE)  %>%
  filter(!STUSPS %in% c('HI', 'PR', 'AK', 'MP', 'GU', 'AS', 'VI'))  %>%
  st_transform(crs = 5070)

ggplot() +
  geom_sf(data = states,
          fill = 'white') +
  geom_sf(data = st %>%
            group_by(SITECODE) %>%
            summarise(n = n()),
          aes(color = n)) +
  scale_color_distiller(name = '# Obs',
                        palette = 'Blues',
                        direction = 2) +
  theme_bw()
```




```
ggsave(file = '../figures/number_summer_temperature_obs.png',
       width = 8,
       height = 5,
       units = 'in',
       dpi = 600)
```

## Summary of model data table

```
# Number of monthly observations across all sites
nrow(st)
```

```
[1] 16157
```

```
# Number of records for July and August
table(st$month)
```

```
   7    8 
8051 8106 
```

```
# Number of records for each year
```

```
table(st$year)
```

```
1999 2000 2001 2002 2003 2004 2005 2006 2007 2008
1177 1322 1322 1499 1739 1766 1661 1716 2041 1914
```

```
# Summary of data
summary(st)
```

```
  SITECODE              year          month          wtmp_mo
Length:16157       Min.   :1999   Min.   :7.000   Min.   : 3.60
Class :character   1st Qu.:2002   1st Qu.:7.000   1st Qu.:18.31
Mode  :character   Median :2004   Median :8.000   Median :22.26
                   Mean   :2004   Mean   :7.502   Mean   :21.72
                   3rd Qu.:2006   3rd Qu.:8.000   3rd Qu.:25.63
                   Max.   :2008   Max.   :8.000   Max.   :33.92
    count                geometry
Min.   :20.00   POINT        :16157
1st Qu.:31.00   epsg:5070    :    0
Median :31.00   +proj=aea ...:    0
Mean   :30.24
3rd Qu.:31.00
Max.   :31.00
```

## USGS flow metrics

- Modeled monthly (July and August) flow estimates for each site (source: USGS).
- Data not easily accessible for new sites.
- We used table to filter stations with data issues that were identified by USGS.

```
flow <- fread('../data/comid_matches_daren_usgs_flow.csv') %>%
  dplyr::select(SITECODE, COMID, Comment, July.Q.mn,
                August.Q.mn, July.Q.md, August.Q.md)
```

## NHDPlus flow metrics

Modeled monthly (July and August) flow estimates from NHDPlus

- Data available for calibration sites and USGS/EPA fish sites.
- Flow values are very correlated with USGS estimates of flow from above.
- USGS values included some very large values, but inspection of streams in Google Maps suggested that NHDPlus flow estimates of river size were more accurate.

```
nhd_dir <- 'C:/Users/RHill04/WorkFolder/GIS/NHDPlusV21/NHDPlusNationalData/NHDPlusV21_National_Sea

nhd_flow <-
  st_read(dsn = paste0(nhd_dir),
          layer = 'NHDFlowline_Network') %>%
```

```
  st_drop_geometry() %>%
  dplyr::select(COMID, QE_07, QE_08) %>%
  pivot_longer(!COMID, names_to = 'tmpcol', values_to = 'nhdflow') %>%
  mutate(month = str_replace(tmpcol, 'QE_0', '') %>%
           as.integer()) %>%
  dplyr::select(-tmpcol)
```

```
Reading layer `NHDFlowline_Network' from data source

`C:\Users\RHill04\WorkFolder\GIS\NHDPlusV21\NHDPlusNationalData\NHDPlusV21_National_Seamless_Flat
tened_Lower48.gdb'
  using driver `OpenFileGDB'
Simple feature collection with 2691339 features and 137 fields
Geometry type: MULTILINESTRING
Dimension:     XYZM
Bounding box:  xmin: -124.7332 ymin: 24.63052 xmax: -66.94983 ymax: 49.37661
z_range:       zmin: 0 zmax: 0
m_range:       mmin: -2.35e-05 mmax: 100
Geodetic CRS:  NAD83
```

## StreamCat (sc) static metrics

Static watershed/local catchment metrics:

- Elevation (Cat)
- Calcium oxide content of underlying lithology (Ws)
- Base flow index (Ws)
- Water table depth (Ws)
- Watershed area (Ws)
- Runoff (ws)
- Clay soil content (Ws)
- Sand soil content (Ws)
- Topographic wetness index (Ws)
- National Anthropogenic Barriers dam density (screened dams of NID) (Ws)
- Hydrologic conductivity (HydrlCond) (Ws)

```
comids <- flow$COMID %>%
  na.omit() %>%
  unique()

#Pull in static watershed metrics
sc <-
  sc_get_data(metric = 'HydrlCond,Runoff,Clay,Sand,WtDep,WetIndex,NABD_Dens,NABD_NRMSTOR,BFI,PRECI
              aoi = 'catchment,watershed',
              comid = comids) %>%
  dplyr::select(COMID, ELEVCAT, CAOWS, BFIWS, WTDEPWS,
                WSAREASQKM, RUNOFFWS, CLAYWS, SANDWS, WETINDEXWS,
                NABD_DENSCAT, NABD_DENSWS, NABD_NRMSTORWS,
                PRECIP8110WS, HYDRLCONDWS) %>%
```

```
  mutate(dam_prescat = ifelse(NABD_DENSCAT > 0, 1, 0),
         dam_presws = ifelse(NABD_DENSWS > 0, 1, 0))
```

## StreamCat Year-Specific NLCD data

### Riparian forest cover (catchment)

1. Extracts yrs. 2001-2008 NLCD from StreamCat for riparian (~100m buffer) watersheds.
2. Filters data to just CONIF, DECID, or MXFST types.
3. Pivots table to include year of NLCD and % riparian forest column.

```
riparian_forest <-
  sc_nlcd(year = '2001, 2004, 2006, 2008',
          aoi = 'riparian_watershed',
          comid = comids) %>%
  dplyr::select(COMID,
                grep('CONIF|DECID|MXFST', names(.))) %>%
  pivot_longer(!COMID, names_to = 'tmpcol', values_to = 'PCTFSTXXXXWSRP100') %>%
  mutate(year = as.integer(
    str_replace_all(tmpcol, 'PCTMXFST|PCTDECID|PCTCONIF|WSRP100', ''))) %>%
  group_by(COMID, year) %>%
  summarise(PCTFSTXXXXWSRP100 = sum(PCTFSTXXXXWSRP100))
```

### Crop cover (watershed)

Same process as riparian forest cover, but for NLCD type CROP.

```
crop <-
  sc_nlcd(year = '2001, 2004, 2006, 2008',
          aoi = 'watershed',
          comid = comids) %>%
  dplyr::select(COMID,
                grep('CROP', names(.))) %>%
  pivot_longer(!COMID, names_to = 'tmpcol', values_to = 'PCTCROPXXXXWS') %>%
  mutate(year = as.integer(
    str_replace_all(tmpcol, 'PCTCROP|WS', ''))) %>%
  dplyr::select(-tmpcol)
```

### Urban cover (watershed)

Same process as riparian forest cover, but for NLCD type PCTURBLO, PCTURBMD, or PCTURBHI.

```
urban <-
  sc_nlcd(year = '2001, 2004, 2006, 2008',
          aoi = 'watershed',
          comid = comids) %>%
  dplyr::select(COMID,
                grep('PCTURBLO|PCTURBMD|PCTURBHI', names(.))) %>%
  pivot_longer(!COMID, names_to = 'tmpcol', values_to = 'PCTURBXXXXWS') %>%
```

```
  mutate(year = as.integer(
    str_replace_all(tmpcol, 'PCTURBLO|PCTURBMD|PCTURBHI|WS', ''))) %>%
  group_by(COMID, year) %>%
  summarise(PCTURBXXXXWS = sum(PCTURBXXXXWS))
```

### Lake/Reservoir (open water) in watershed (watershed)

Same process as riparian forest cover, but for NLCD type PCTOW.

Variable added to interact with dam presence/absence to account for stations that occur below natural lakes or man made reservoirs.

```
water <-
  sc_nlcd(year = '2001, 2004, 2006, 2008',
          aoi = 'watershed',
          comid = comids) %>%
  dplyr::select(COMID,
                grep('PCTOW', names(.))) %>%
  pivot_longer(!COMID, names_to = 'tmpcol', values_to = 'PCTOWXXXXWS') %>%
  mutate(year = as.integer(
    str_replace_all(tmpcol, 'PCTOW|WS', ''))) %>%
  group_by(COMID, year) %>%
  summarise(PCTOWXXXXWS = sum(PCTOWXXXXWS))
```

## PRISM Climate Data

### Air temperature

```
years <- 1999:2008

# Set the PRISM directory (creates directory in not present)
prism_set_dl_dir("../data/prism_data", create = TRUE)

# Download monthly PRISM rasters (tmean)
get_prism_monthlys('tmean',
                   years = years,
                   mon = 7:8,
                   keepZip = FALSE)
```

```
  |
  |                                                                      |   0%
  |
  |====                                                                  |   5%
  |
  |=======                                                               |  10%
  |
  |==========                                                            |  15%
  |
```

```
  |==============                                                        |  20%
  |
  |==================                                                    |  25%
  |
  |======================                                                |  30%
  |
  |========================                                              |  35%
  |
  |============================                                          |  40%
  |
  |================================                                      |  45%
  |
  |==================================                                    |  50%
  |
  |====================================                                  |  55%
  |
  |========================================                              |  60%
  |
  |============================================                          |  65%
  |
  |==============================================                        |  70%
  |
  |================================================                      |  75%
  |
  |==================================================                    |  80%
  |
  |======================================================                |  85%
  |
  |========================================================              |  90%
  |
  |============================================================          |  95%
  |
  |======================================================================| 100%
```

```r
# Create stack of PRISM climate rasters to extract values
tmn <- pd_stack((prism_archive_subset("tmean","monthly",
                                      years = years,
                                      mon = 7:8)))

# Extract tmean at sample points and massage data
tmn <- terra::extract(tmn,
                      # Transform pts to CRS of PRISM on the fly
                      pts %>%
                        st_transform(crs = st_crs(tmn))) %>%

  # Add site IDs to extracted values
  data.frame(SITECODE = pts$SITECODE, .) %>%

  # Remove front and back text from PRISM year/month in names
  rename_with( ~ stringr::str_replace_all(., 'PRISM_tmean_stable_4kmM3_|_bil', '')) %>%
```

```
  # Pivot to long table and calle column tmeanPRISM
  pivot_longer(!SITECODE, names_to = 'year_month',
               values_to = 'tmeanPRISM') %>%

  # Create new column of year
  mutate(year = year(ym(year_month)),
         month = month(ym(year_month))) %>%

  dplyr::select(-year_month)
```

## Precipitation

```
get_prism_monthlys('ppt',
                   years = years,
                   mon = 7:8,
                   keepZip = FALSE)
```

```
  |
  |                                                                      |   0%
  |
  |====                                                                  |   5%
  |
  |=======                                                               |  10%
  |
  |==========                                                            |  15%
  |
  |==============                                                        |  20%
  |
  |==================                                                    |  25%
  |
  |======================                                                |  30%
  |
  |=========================                                             |  35%
  |
  |=============================                                         |  40%
  |
  |================================                                      |  45%
  |
  |====================================                                  |  50%
  |
  |========================================                              |  55%
  |
  |============================================                          |  60%
  |
  |================================================                      |  65%
  |
  |====================================================                  |  70%
```

```
  |
  |=================================================                     |  75%
  |
  |=====================================================                 |  80%
  |
  |=========================================================             |  85%
  |
  |=============================================================         |  90%
  |
  |=================================================================     |  95%
  |
  |=====================================================================| 100%
```

```
ppt <- pd_stack((prism_archive_subset("ppt","monthly",
                                      years = years,
                                      mon = 7:8)))

ppt <- terra::extract(ppt,
                      pts %>%
                        st_transform(crs = st_crs(ppt))) %>%
  data.frame(SITECODE = pts$SITECODE, .) %>%
  rename_with( ~ stringr::str_replace_all(., 'PRISM_ppt_stable_4kmM3_|_bil', '')) %>%
  pivot_longer(!SITECODE, names_to = 'year_month',
               values_to = 'pptPRISM') %>%
  mutate(year = year(ym(year_month)),
         month = month(ym(year_month))) %>%

  dplyr::select(-year_month)
```

## Combine data for modeling

- Code creates crosswalk that matches the closest temperature years and NLCD years.
- All geospatial metrics are then joined to location (COMID)/month/year combinations of observed water temperatures.

```
nlcd_years <-
  data.table(year = c(2001, 2004, 2006, 2008)) %>%
  mutate(merge = year) %>%
  setkeyv('merge')

st_years <-
  data.table(year = 1999:2008) %>%
  mutate(merge = year) %>%
  setkeyv('merge')

nearest <-
  nlcd_years[st_years, roll = 'nearest'] %>%
  dplyr::select(year, i.year) %>%
  rename(nlcd_year = year,
         year = i.year)
```

```r
st <- st %>%
  left_join(nearest, join_by(year)) %>%
  left_join(tmn,
            join_by(SITECODE, year, month)) %>%
  left_join(ppt,
            join_by(SITECODE, year, month)) %>%
  left_join(flow, join_by(SITECODE)) %>%
  left_join(sc, join_by(COMID)) %>%
  left_join(riparian_forest,
            join_by(COMID == COMID,
                    nlcd_year == year)) %>%
  left_join(crop,
            join_by(COMID == COMID,
                    nlcd_year == year)) %>%
  left_join(urban,
            join_by(COMID == COMID,
                    nlcd_year == year)) %>%
  left_join(water,
            join_by(COMID == COMID,
                    nlcd_year == year)) %>%
  left_join(nhd_flow,
            join_by(COMID == COMID,
                    month == month)) %>%
  mutate(q_mn = ifelse(month == 7,
                       July.Q.mn,
                       August.Q.mn),
         q_md = ifelse(month == 7,
                       July.Q.md,
                       August.Q.md)) %>%
  dplyr::select(-July.Q.mn:-August.Q.md)

# Write output file for modeling
write_rds(st,
          file = '../data/summer_data.2024.08.08.rds',
          compress = "xz")
```