

Preparing Regional Weather Data Analysis Project

Marc Los Huertos

January 30, 2024 (ver. 0.4)

1 Background

1.1 Project Goals

1.2 Project Stages

1. Data Collection
2. Data Processing
3. Data Analysis
4. Data Visualization
5. Communicating Project

1.3 Global Weather Station Data

1.4 Download Station Inventory

```
library(here)

## here() starts at /home/mwl04747/RTricks

# Get Stations Data (Inventory)
inventory = read.table("https://www.ncei.noaa.gov/pub/data/ghcn/daily/ghcnd-inventory.txt")

# Define Variable Names
inventory_names = c("ID", # 1-11 Character
                    "LATITUDE", # 13-20 Real
                    "LONGITUDE", # 22-30 Real
                    "ELEMENT", # 32-35 Character
                    "FIRSTYEAR", # 37-40 Integer
                    "LASTYEAR") # 42-45 Integer
```

```

# Assign Variable Names
names(inventory) = inventory_names

# Check the structure of the data
str(inventory)

## 'data.frame': 747094 obs. of 6 variables:
## $ ID : chr "ACW00011604" "ACW00011604" "ACW00011604" "ACW00011604" ...
## $ LATITUDE : num 17.1 17.1 17.1 17.1 17.1 ...
## $ LONGITUDE: num -61.8 -61.8 -61.8 -61.8 -61.8 ...
## $ ELEMENT : chr "TMAX" "TMIN" "PRCP" "SNOW" ...
## $ FIRSTYEAR: int 1949 1949 1949 1949 1949 1949 1949 1949 1949 1949 ...
## $ LASTYEAR : int 1949 1949 1949 1949 1949 1949 1949 1949 1949 1949 ...

```

```

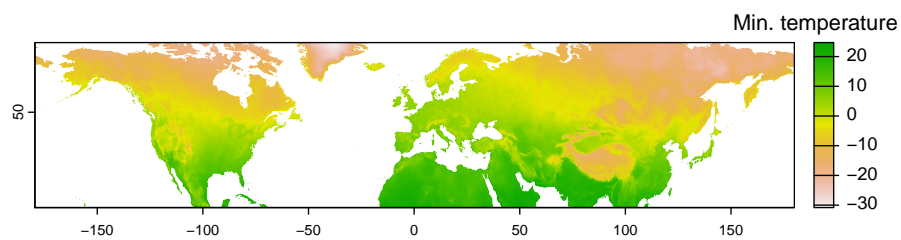
library(geodata)

## Loading required package: terra
## terra 1.7.55
##
## Attaching package: 'terra'
## The following object is masked from 'package:knitr':
##
##      spin

d <- worldclim_country(country = "USA", var = "tmin",
                       path = tempdir())
terra::plot(mean(d), plg = list(title = "Min. temperature (C)"))

##
|-----|-----|-----|-----|
=====

```



1.5 Visualizing of Active Weather Stations with Maximum Daily Temperature Readings

```
# Subset data for TMAX (Max Temperature) Element
inventory.TMAX = subset(inventory, subset=ELEMENT=="TMAX")

str(inventory.TMAX)

## 'data.frame': 40395 obs. of 6 variables:
## $ ID      : chr  "ACW00011604" "ACW00011647" "AE000041196" "AEM00041194" ...
## $ LATITUDE: num   17.1 17.1 25.3 25.3 24.4 ...
## $ LONGITUDE: num  -61.8 -61.8 55.5 55.4 54.7 ...
## $ ELEMENT  : chr   "TMAX" "TMAX" "TMAX" "TMAX" ...
## $ FIRSTYEAR: int   1949 1961 1944 1983 1983 1994 1973 1973 1966 1973 ...
```

```
## $ LASTYEAR : int 1949 1961 2024 2024 2024 2024 1992 2020 2021 2020 ...

#plot(inventory.TMAX$LONGITUDE, inventory.TMAX$LATITUDE)

#plot(inventory.TMAX$LONGITUDE, inventory.TMAX$LATITUDE, pch=20, cex=.4)

# Selective ~Active Stations

inventory.TMAX = subset(inventory.TMAX, subset=LASTYEAR>=2022); str(inventory.TMAX)

## 'data.frame': 12745 obs. of 6 variables:
## $ ID : chr "AE000041196" "AEM00041194" "AEM00041217" "AEM00041218" ...
## $ LATITUDE : num 25.3 25.3 24.4 24.3 36.7 ...
## $ LONGITUDE: num 55.52 55.36 54.65 55.61 3.25 ...
## $ ELEMENT : chr "TMAX" "TMAX" "TMAX" "TMAX" ...
## $ FIRSTYEAR: int 1944 1983 1983 1994 1940 1940 1958 1886 1878 1880 ...
## $ LASTYEAR : int 2024 2024 2024 2024 2024 2024 2024 2023 2024 2024 ...

#plot(inventory.TMAX$LONGITUDE, inventory.TMAX$LATITUDE, pch=20, cex=.4, xlab="Long", ylab=
#par(mfrow=c(2,2))
```

2 Creating Up-to-Date Weather Datasets

To prepare for the project, we need to accomplish two things:

1. Select a region, i.e. State, of interest
2. Read in the most recent EPA information on the state.

3 Updated Station Selection Dataset

3.1 Download Recent

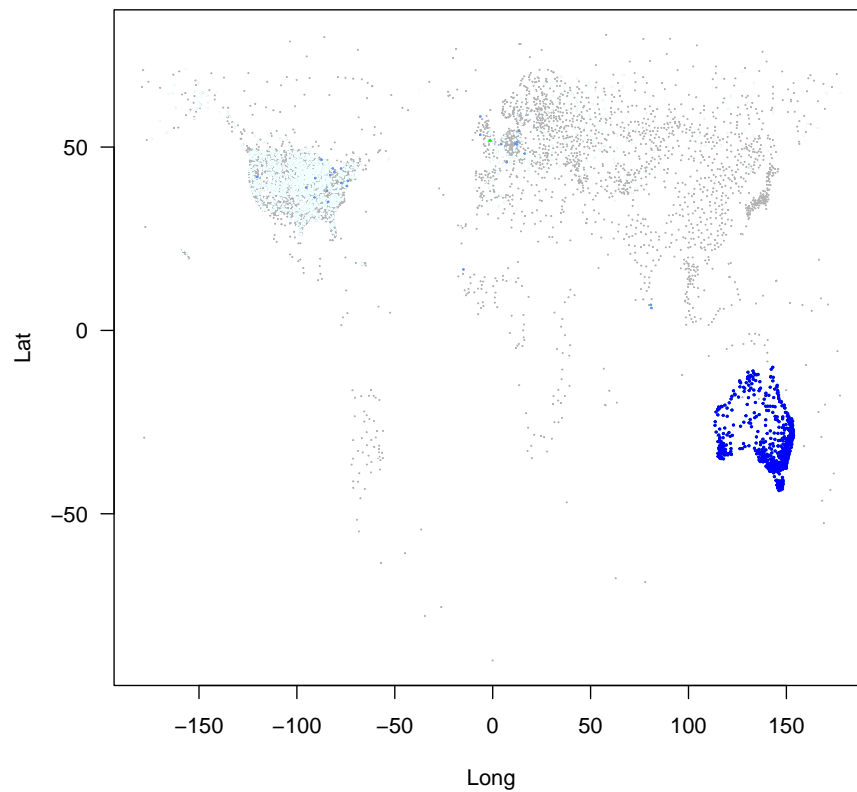
3.2 States in GHCND-station Dataset

The inventory has a list of stations and map coordinates (latitude and longitude). However, it's not easy to select a region, like a state, from the inventory. Thus, we need to merge the inventory with a dataset that includes state names.

It's a bit strange, but the dataset includes US states and Canadian Provinces, plus various territories of the US.

```
station_names = c("ID",           # 1-11   Character 11
                  "LATITUDE",     # 13-20  Real      8
                  "LONGITUDE",    # 22-30  Real      9
```

Figure 1: A plot of the global weather stations. Note the increase in stations over time and spatial distribution. Australia has most of it's stations with 1750 start dates, but I suspect most of these stations have lots of missing data.




```
stations.USCan = subset(temp.TMAX, subset=(STATE!=" ")) # Remove Stations that STATE = blank
```

3.3 Select Active Stations

How many stations are in the state? `'r nrow(stations.USCan)'`!

```
stations.active = subset(stations.USCan, subset=LASTYEAR>=2022)
str(stations.active)

## 'data.frame': 7751 obs. of 8 variables:
## $ ID      : chr  "AQW00061705" "CA001011500" "CA001012055" "CA001012475" ...
## $ LATITUDE : num  -14.3 48.9 48.8 48.4 48.4 ...
## $ LONGITUDE : num  -171 -124 -124 -123 -123 ...
## $ ELEMENT  : chr  "TMAX" "TMAX" "TMAX" "TMAX" ...
## $ FIRSTYEAR : int  1966 1979 1960 1997 1991 1991 2007 1972 1970 1996 ...
## $ LASTYEAR  : int  2024 2024 2023 2024 2024 2024 2024 2024 2022 2024 ...
## $ STATE     : chr  "AS" "BC" "BC" "BC" ...
## $ STATE_NAME: chr  "AMERICAN SAMOA" "BRITISH COLUMBIA" "BRITISH COLUMBIA" "BRITISH COLUMBIA" ...

nrow(stations.active)

## [1] 7751
```

3.4 Select 5 Stations for Each State

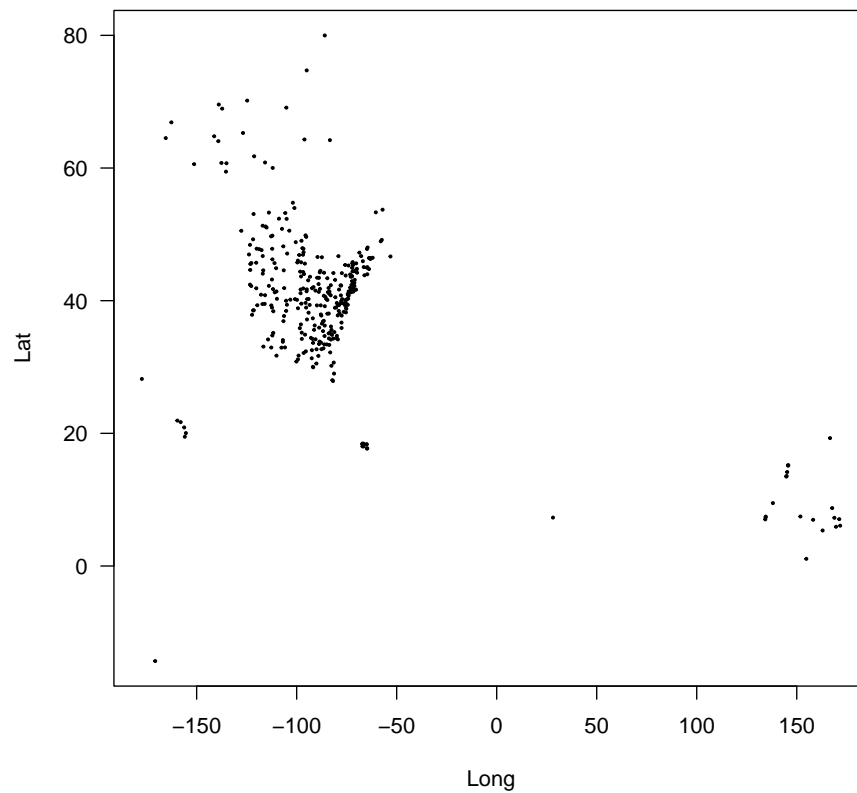
To accomplish this, I need to do a loop to select the first 5 stations for each state.

```
# Loop to select 5 stations for each state
#stations.active.oldest = subset(stations.active, subset=FIRSTYEAR==min(FIRSTYEAR))

for(i in 1:nrow(States)) {
  state.df = subset(stations.active, subset=STATE==States$STATE[i])
  if(nrow(state.df) > 5) {
    state.df = state.df[order(state.df$FIRSTYEAR),][1:5,]
  }
  if(i==1) {
    stations.active.oldest = state.df
  } else {
    stations.active.oldest = rbind(stations.active.oldest, state.df)
  }
}
```

4 Plot Results

```
plot(stations.active.oldest$LONGITUDE, stations.active.oldest$LATITUDE, pch=20, cex=.4, xlab=
```



4.1 Next Steps

```
# export file to csv  
write.csv(stations.active.oldest, here("04_Regional_Climate_Trends", "stations.active.oldest
```