

rsnodas vignette

Logan Schneider

July 12, 2022

Contents

1	Introduction	2
1.1	Installation	2
2	Downloading Download SNODAS, SNOTEL and PRISM	2
2.1	SNODAS	2
2.2	Download SNOTEL	3
2.3	Download PRISM climate data	3
3	GAM to Raster	4
3.1	Station Density	5
3.2	Ensemble SNODAS and GAM predictions	6

1 Introduction

This `rsnodas` package allows users to access, clean, visualize, and analyze data from the following sources:

- [Snow Data Assimilation System](#) (SNODAS)
- [University of Arizona](#) (UA)
- [Parameter-elevation Regressions on Independent Slopes Model](#) (PRISM)
- [Historical Climatology Network daily](#) (GHCND) which provides access to Snow Telemetry (SNOTEL) stations

SNODAS, UA, and PRISM provide data in the form of gridded rasters for the contiguous United States. Each source can have different spatial resolutions like 800m, 1km, or 4km and multiple temporal scales of daily, monthly, annual, and 30 year normals. GHCND provides in-situ measurements, spatial point data, with different temporal scales. This package focuses on downloading the daily measurements. Please refer to their webpages for more information about each data product.

1.1 Installation

`rsnodas` is available on GitHub and can be installed with devtools:

```
# install.packages("devtools")
library(devtools)
install_github("lschneider93/rsnodas")
```

The outline will be Downloading SNODAS, SNOTEL, and PRISM. After downloading all of those sources of data, the process of creating a raster of predictions and calculating station density will be shown. The blending of SNODAS and the Generalized Additive model can then be explored.

2 Downloading Download SNODAS, SNOTEL and PRISM

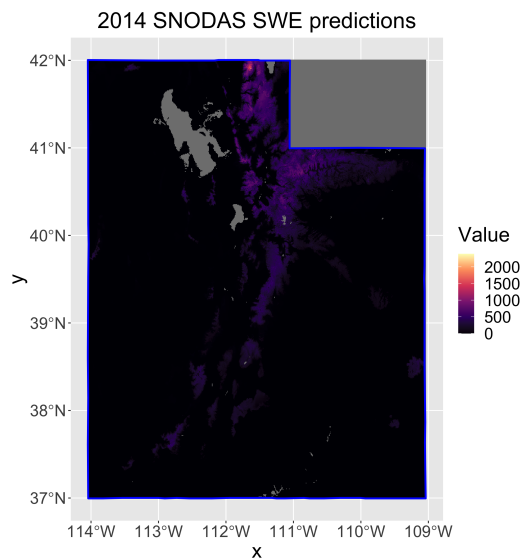
2.1 SNODAS

SNODAS provides maps of Snow water equivalent (SWE), Snow Depth (SnD), Snow melt runoff (SM), Sublimation from the Snow Pack (SSP), Sublimation of Blowing Snow (SBS), Solid and liquid precipitation (PPT), and Snow pack average temperature (SPTave). `download_snodas` allows us to choose the variables we want to download. Note that the dates need to be in 'YYYY-MM-DD' or 'YYYY-M-D' format. The function `format_date` can be used to create character vectors in the needed format to help quickly for mass downloading.

```
# Download Snodas SWE data for April 1st in 2014 and 2015
snodas_april1 <- download_snodas(dates = format_dates(day = 1,
                                                    month = 4,
                                                    year = 2014:2015),
                                masked = TRUE,
                                remove_zip = TRUE,
                                data_saved = c("swe"),
                                out_dir = paste0("/Users/loganschneider/Desktop/GitHub",
                                                  "/rsnodas/vignette/snodas_data"),
                                GTiff = FALSE)
```

After downloading SNODAS, you can crop it to the area or state of interest. These steps aren't shown but this map was cropped to the state of Utah.

```
# Plot SNODAS April 1st 2015 SWE map of Utah with blue outline
g <- ggplot() +
  stars::geom_stars(data = snodas_ut_2014) +
  geom_sf(data = ut_map, fill = "NA", size = 1, color = "blue") +
  ggtitle("2014 SNODAS SWE predictions") +
  scale_fill_viridis_c(option = "A") +
  theme(plot.title = element_text(hjust = 0.5, size = 24),
        text = element_text(size = 22))
```



2.2 Download SNOTEL

The `data-raw` folder contains an script titled `DATASET` that shows how to download the all SNOTEL data by using `download_all_ghcnd_stations`. Note that there is more than 30 gigabytes of data and this will take time to download. The functions `download_all_ghcnd_stations` and `download_all_ghcnd_stations` are for the purpose of sifting through all the data to get the stations in a specific state. These functions were used to create the dataset `april_1_snotel_data`.

```
# Code to download all stations in your working directory. A file path could
# have been used instead of ".".
download_all_ghcnd_stations(directory = ".")

# april 1st data for SNOTEL stations in Utah and subset to just April 1st, 2014.
snotel_ut <- rsnodas::april_1_snotel_data
snotel_ut_2014 <- snotel_ut[snotel_ut$DATE == "2014-04-01", ]
```

2.3 Download PRISM climate data

The PRISM climate group has been provided climate maps since 1895 for the Precipitation (PPT), Minimum and maximum temperature (Tmin and Tmax), Minimum and maximum vapor pressure deficit (vpdmin and vpdmax), mean dew point (tdmean), Elevation (Elev), Total global shortwave solar radiation.

Most of this data from PRISM can be downloaded by utilizing the R package, `prism`, and more examples and information is available at [this link](#). The function `download_prism` can be used to download and store PRISM climate data that can be utilized later for model prediction. These are examples of downloading daily, monthly, and yearly data from separate years and multiple types and the store them in the directory.

Downloading PRISM can also be completed using `rsnodas` by using `download_prism`. This is just one function that can download daily, monthly, or yearly PRISM maps.

```
## NOT RUN, This was ran earlier
# Example of downloading 4km monthly precipitation maps from Dec. 2013 - Feb. 2014
download_prism(sp_res = "4km", data = c("ppt"), t_res = "monthly",
               start_date = as.Date("2013-12-01"), end_date = as.Date("2014-02-15"),
               out_dir = "/Users/loganschneider/Desktop/GitHub/prism")

# Example of downloading daily 4km mean and max temperature maps for Jan. 1-2, 2004
download_prism(sp_res = "4km", data = c("tmax", "tmean"), t_res = "daily",
               start_date = as.Date("2013-11-15"), end_date = as.Date("2013-11-17"),
               out_dir = "/Users/loganschneider/Desktop/GitHub/prism")

# Example of downloading yearly Daily minimum and max vapor pressure deficit
# at 4km resolution maps for Jan. 1, 2017-2019
download_prism(sp_res = "4km", data = c("vpdmin", "vpdmax"), t_res = "yearly",
               start_date = as.Date("2017-01-01"), end_date = as.Date("2019-01-15"),
               out_dir = "/Users/loganschneider/Desktop/GitHub/prism")
```

Note that all PRISM files need to be in the same location in order to use the function `gam_to_df` in the next section.

3 GAM to Raster

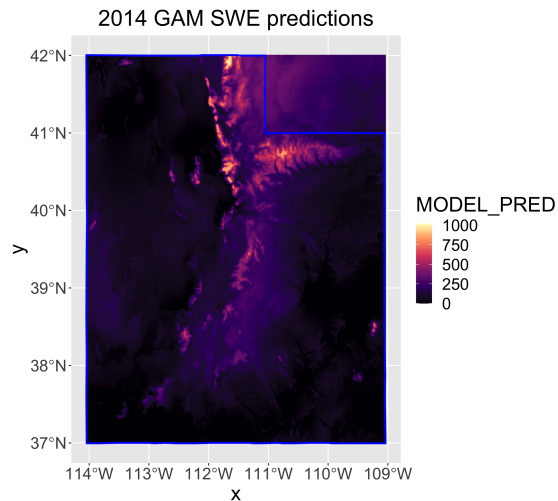
After downloading the point data provided by SNOTEL sites and climate variables from PRISM. The process of creating gridded raster estimates will be demonstrated. This uses the function, `gam_to_raster` and accesses files in one directory. This allows users to explore different models and explore different climate variable combinations.

```
# creation of a data frame with all the PRISM variables of precipitation and elevation
gam_2014 <- gam_to_df(model_data = snotel_ut_2014,
                     raster_template = snodas_ut_2014,
                     path_to_prism = "/Users/loganschneider/Desktop/GitHub/prism",
                     model_x <- c("ppt_normal_annual", "elevation"),
                     model_y <- c("VALUE"),
                     coords = c("LONGITUDE", "LATITUDE"))

# This allows users to explore using different models with the same climate variables.
model <- mgcv::gam(data = snotel_ut_2014,
                  VALUE ~ s(LONGITUDE, LATITUDE, bs = "sos", k = 25) +
                    s(ppt_normal_annual) +
                    s(elevation),
                  method = "REML")

# After creating a model, we can make predictions with the information available.
gam_rast <- df_to_raster(model = model, df = gam_2014)
```

```
g <- ggplot() +
  stars::geom_stars(data = gam_rast) +
  geom_sf(data = ut_map, fill = "NA", size = 1, color = "blue") +
  ggtitle("2014 GAM SWE predictions") +
  scale_fill_viridis_c(option = "A") +
  theme(plot.title = element_text(hjust = 0.5, size = 24),
        text = element_text(size = 22))
```



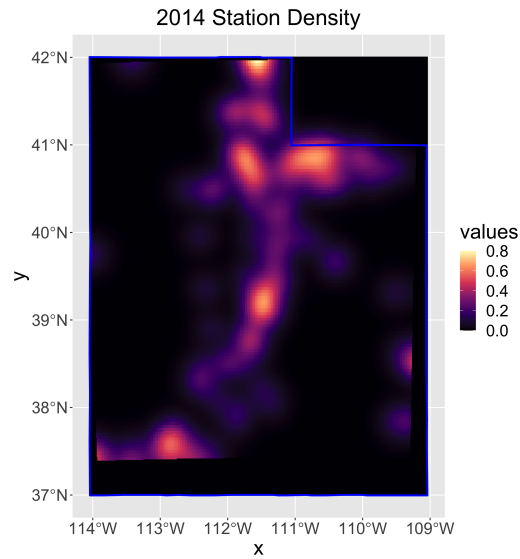
3.1 Station Density

Before we can start blending these maps together, there needs to be some kind of weight associated with both maps. This is going to be accomplished by looking at observation or station density. The `sigma` argument will extend the distance and is in meters while the gridded product is in kilometers. Changing the `max_weight` argument gives more weight to the Generalized Additive model or the model from the in-situ measurements.

```
dens_2014 <- points_to_density_stars(sp_points = snotel_ut_2014,
                                     coords = c("LONGITUDE", "LATITUDE"),
                                     raster_template = snodas_ut_2014,
                                     sigma = 15000,
                                     max_weight = .80,
                                     flat_crs = "+proj=utm + zone=12 + datum=WGS84")

ggplot() +
  stars::geom_stars(data = dens_2014) +
  geom_sf(data = ut_map, fill = "NA", size = 1, color = "blue") +
  ggtitle("2014 Station Density") +
  scale_fill_viridis_c(option = "B") +
  theme(plot.title = element_text(hjust = 0.5, size = 24),
        text = element_text(size = 22))
```

The weighting scheme can change in the future.



3.2 Ensemble SNODAS and GAM predictions

Lastly, you can use all the previous maps to create a final prediction or estimate of SWE. This process can be applied to multiple types of spatial problems.

```
comb_map <- ((dens_2014) * gam_rast) +
  ((1 - dens_2014) * snodas_ut_2014)

g <- ggplot() +
  stars::geom_stars(data = comb_map) +
  geom_sf(data = ut_map, fill = "NA", size = 1, color = "blue") +
  ggtitle("2014 Ensemble Map") +
  scale_fill_viridis_c(option = "B") +
  theme(plot.title = element_text(hjust = 0.5, size = 24),
        text = element_text(size = 22))
```

