eBirdst Script for Chapter 1

Andres N. Rosales

2024-10-10

Contents

Analyzing Landscape-Level Factors Influencing Upland Sandpiper Population Dynamics

This code extracts and manipulates eBird status and trends data for the first chapter of my thesis. More infomation and visualizations can be found at https://science.ebird.org/en/status-and-trends/species/uplsan/trends-map

Library

```
library(dplyr)
library(purrr)
library(ggplot2)
library(rnaturalearth)
library(sf)
library(raster)
library(terra)
library(ebirdst)
```

Extract Status and Trend Data

This code views and extracts available status and trend data for Upland Sandpiper.

Set unique key to access eBirdst data (expires Jan 18th 2025)

```
set_ebirdst_access_key("3t1ddtda3uh3", overwrite = TRUE)
```

```
## eBird Status and Trends access key stored in: C:/Users/ROSALESA/Documents/.Renviron
```

Show all species with status and trend data available. We will subset to show Upland Sandpiper data availability.

```
trends_runs <- ebirdst_runs %>%
  filter(has_trends) %>%
  dplyr::select(
    species_code, common_name, trends_season, trends_region,
    trends_start_year, trends_end_year, trends_start_date,
    trends_end_date, rsquared, beta0
  )

upsa <- trends_runs %>%
  filter(common_name == "Upland Sandpiper") %>%
  dplyr::select(trends_start_year, trends_end_year,
                trends_start_date, trends_end_date)

upsa
```

```
## # A tibble: 1 x 4
##   trends_start_year trends_end_year trends_start_date trends_end_date
##               <dbl>           <dbl> <chr>             <chr>
## 1              2012            2022 05-31             06-28
```

Create an object of Upland Sandpiper trends (2012-2022). *waiting for new eBird data release in November

```
uppies_trends <- load_trends("uplsan")

(uppies_trends)
```

```
## # A tibble: 3,773 x 17
##    species_code season  start_year end_year start_date end_date srd_id longitude
##    <chr>        <chr>        <int>    <int> <chr>      <chr>     <int>     <dbl>
##  1 uplsan       breedi~       2012     2022 05-31      06-28    126707     -147.
##  2 uplsan       breedi~       2012     2022 05-31      06-28    126708     -146.
##  3 uplsan       breedi~       2012     2022 05-31      06-28    126709     -145.
##  4 uplsan       breedi~       2012     2022 05-31      06-28    126710     -145.
##  5 uplsan       breedi~       2012     2022 05-31      06-28    128207     -147.
##  6 uplsan       breedi~       2012     2022 05-31      06-28    128208     -146.
##  7 uplsan       breedi~       2012     2022 05-31      06-28    128209     -145.
##  8 uplsan       breedi~       2012     2022 05-31      06-28    128210     -145.
##  9 uplsan       breedi~       2012     2022 05-31      06-28    128211     -144.
## 10 uplsan       breedi~       2012     2022 05-31      06-28    128212     -143.
## # i 3,763 more rows
## # i 9 more variables: latitude <dbl>, abd <dbl>, abd_ppy <dbl>,
## #   abd_ppy_lower <dbl>, abd_ppy_upper <dbl>, abd_ppy_nonzero <lgl>,
## #   abd_trend <dbl>, abd_trend_lower <dbl>, abd_trend_upper <dbl>
```

```
write.csv(uppies_trends, file = "C:/Users/ROSALESA/OneDrive - University of Saskatchewan/17_Analysis/Cha
```

Calculate Abundance for each year (2012-2022)

Use the equation

$Ay = A2017Ö(1 + r)^{y-2017}$

- Ay is the abundance in year y.
- A2017 is the abundance in 2017 (median year).
- r is the annual percent change expressed as a decimal.
- y is the year of interest.

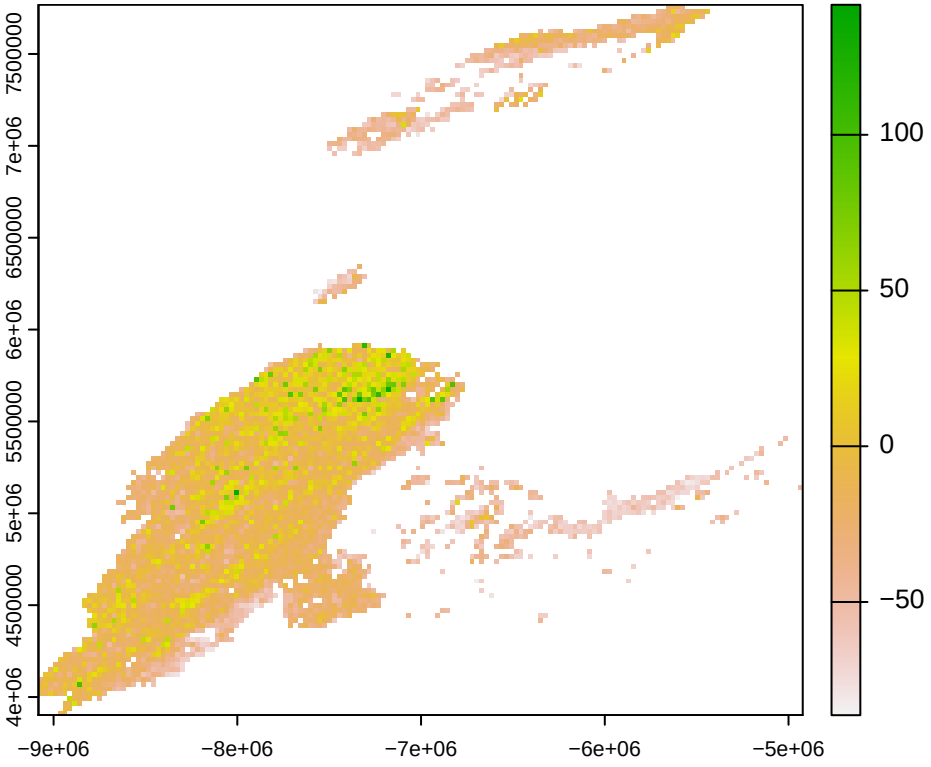
```
uppies_trends <- uppies_trends %>%
  bind_cols(map_dfc(2012:2022,
                    ~ tibble(!!paste0("abd_", .x) := uppies_trends$abd *
                               (1 + uppies_trends$abd_ppy) ^ (.x - 2017))))
```

Rasterize Upland Sandpiper trends

rasterize cumulative trend estimate using `terra` package

```
#keep projection and change in arcpro
#save as a geotiff for use in arcpro
writeRaster(trends_raster, filename = "C:/Users/ROSALESA/OneDrive - University of Saskatchewan/14_Arcpro

plot(trends_raster)
```



Vector of Spatial Trends and Abundance

load vector of the pixel points

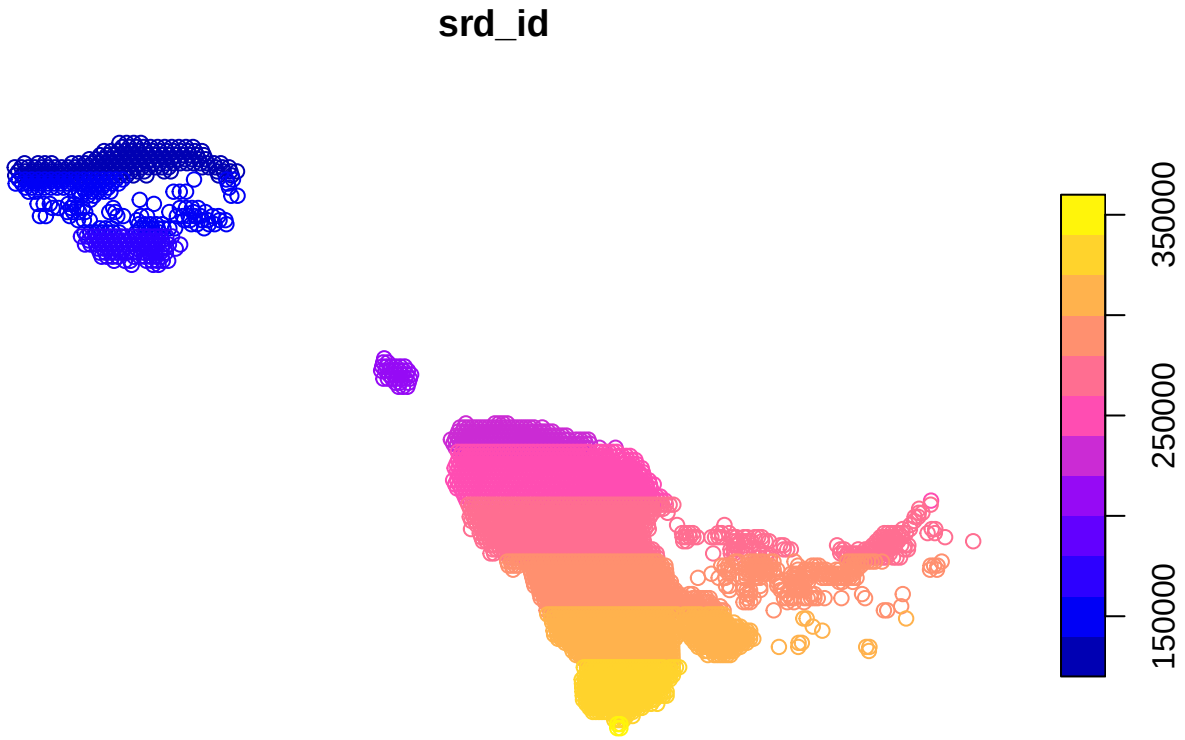
```
trends_pixel <- st_as_sf(uppies_trends,
                         coords = c("longitude", "latitude"),
                                    crs = 4326)
# uses WGS 84 coordinate system (EPSG 4326)
print(trends_pixel)
```

```
## Simple feature collection with 3773 features and 26 fields
## Geometry type: POINT
## Dimension:     XY
## Bounding box:  xmin: -157.0055 ymin: 35.20425 xmax: -64.17485 ymax: 69.73636
## Geodetic CRS:  WGS 84
## # A tibble: 3,773 x 27
##    species_code season   start_year end_year start_date end_date srd_id      abd
##  * <chr>        <chr>         <int>    <int> <chr>      <chr>     <int>    <dbl>
##  1 uplsan       breeding       2012     2022 05-31      06-28    126707 0.00356
##  2 uplsan       breeding       2012     2022 05-31      06-28    126708 0.00346
##  3 uplsan       breeding       2012     2022 05-31      06-28    126709 0.0186
##  4 uplsan       breeding       2012     2022 05-31      06-28    126710 0.000733
##  5 uplsan       breeding       2012     2022 05-31      06-28    128207 0.0227
##  6 uplsan       breeding       2012     2022 05-31      06-28    128208 0.0436
```

```
##  7 uplsan       breeding       2012     2022 05-31      06-28    128209 0.0220
##  8 uplsan       breeding       2012     2022 05-31      06-28    128210 0.0361
##  9 uplsan       breeding       2012     2022 05-31      06-28    128211 0.0504
## 10 uplsan       breeding       2012     2022 05-31      06-28    128212 0.0697
## # i 3,763 more rows
## # i 19 more variables: abd_ppy <dbl>, abd_ppy_lower <dbl>, abd_ppy_upper <dbl>,
## #   abd_ppy_nonzero <lgl>, abd_trend <dbl>, abd_trend_lower <dbl>,
## #   abd_trend_upper <dbl>, abd_2012 <dbl>, abd_2013 <dbl>, abd_2014 <dbl>,
## #   abd_2015 <dbl>, abd_2016 <dbl>, abd_2017 <dbl>, abd_2018 <dbl>,
## #   abd_2019 <dbl>, abd_2020 <dbl>, abd_2021 <dbl>, abd_2022 <dbl>,
## #   geometry <POINT [°]>
```

Export shapefile to path

```
write_sf(trends_pixel,"C:/Users/ROSALESA/OneDrive - University of Saskatchewan/14_Arcpro/Chapter1/Shape
         layer = "uppies_trends")
#plot the vector of trend locations using the srd_id (unique code for every pixel)
plot(trends_pixel["srd_id"])
```



Session Info

```
sessionInfo()
```

```
## R version 4.3.1 (2023-06-16 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 11 x64 (build 22631)
##
## Matrix products: default
##
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## time zone: America/Regina
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] ebirdst_3.2022.1    terra_1.7-55        raster_3.6-26
## [4] sp_2.1-1            sf_1.0-14           rnaturalearth_0.3.4
## [7] ggplot2_3.4.4       purrr_1.0.2         dplyr_1.1.3
##
## loaded via a namespace (and not attached):
##  [1] utf8_1.2.3         generics_0.1.3     class_7.3-22       KernSmooth_2.23-21
##  [5] stringi_1.7.12     lattice_0.21-8     digest_0.6.33      magrittr_2.0.3
##  [9] evaluate_0.23      grid_4.3.1         fastmap_1.1.1      jsonlite_1.8.7
## [13] e1071_1.7-13       DBI_1.1.3          httr_1.4.7         fansi_1.0.4
## [17] scales_1.2.1       codetools_0.2-19   cli_3.6.1          rlang_1.1.1
## [21] units_0.8-4        bit64_4.0.5        munsell_0.5.0      withr_2.5.1
## [25] yaml_2.3.7         tools_4.3.1        tzdb_0.4.0         colorspace_2.1-0
## [29] assertthat_0.2.1   vctrs_0.6.3        R6_2.5.1           proxy_0.4-27
## [33] lifecycle_1.0.3    classInt_0.4-10    stringr_1.5.0      bit_4.0.5
## [37] arrow_13.0.0.1     pkgconfig_2.0.3    pillar_1.9.0       gtable_0.3.4
## [41] glue_1.6.2         Rcpp_1.0.11        highr_0.11         xfun_0.47
## [45] tibble_3.2.1       tidyselect_1.2.1   rstudioapi_0.15.0  knitr_1.48
## [49] htmltools_0.5.7    rmarkdown_2.28     compiler_4.3.1
```