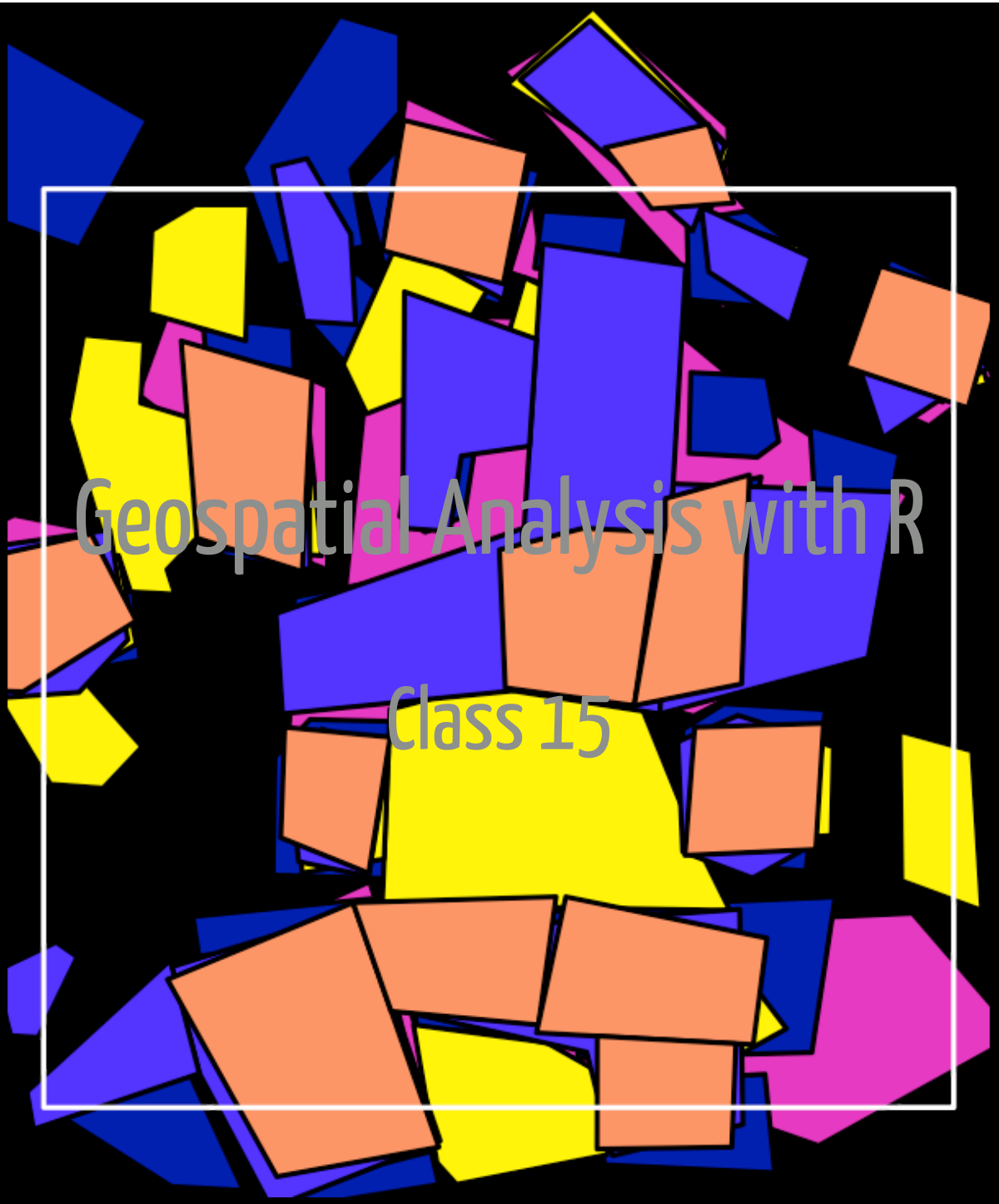

Geospatial Analysis with R

Class 15

Today

- More visualization
- A bit more analytics practice
- Maybe some spatial to start

Visualization

- `graphics` versus `ggplot`

```
library(tidyverse)
fs <- dir(system.file("extdata/", package = "geospaar"), pattern = "FAOSTAT",
          full.names = TRUE)
crop_ylds <-  map(fs, function(x) {
  readr::read_csv(x) %>% dplyr::select(Item, Area, Element, Year, Value) %>%
    spread(key = Element, value = Value) %>%
    rename(crop = Item, country = Area, year = Year,
           harv_area = `Area harvested`, prod = Production) %>%
    mutate(yields = prod / harv_area) %>%
    mutate(country = ifelse(country == "South Africa", "ZAF", country)) %>%
    mutate(country = ifelse(country == "Zambia", "ZMB", country))
}) %>% reduce(rbind)
```

`graphics` examples

- Scatter
- Histogram
- multi-panel

Scatter

```
# single line
cols <- c("red", "orange", "blue", "purple")
par(mfrow = c(2, 2), mar = rep(0, 4))
for(i in 1:4) {
  crop_ylds %>% filter(country == "ZAF" & crop == "Maize") %>%
    plot(yields ~ year, data = ., pch = i, col = cols[i], axes = FALSE)
}
# multiple lines
par(mar = rep(3, 4))
# crop_ylds[crop_ylds$country == "ZAF" & crop_ylds$crop == "Maize", ]
crop_ylds %>% filter(country == "ZAF" & crop == "Maize") %>%
  plot(yields ~ year, data = ., pch = i, col = cols[i], axes = FALSE)
# dat <- crop_ylds[crop_ylds$country == "ZAF" & crop_ylds$crop == "Maize", ]
# plot(yields ~ year, data = dat, pch = i, col = cols[i], axes = FALSE)
axis(side = 1, las = 2)
axis(side = 2, las = 2)
```

Scatter 2

```
crop_ylds %>% filter(country == "ZAF" & crop == "Maize") %>%
  plot(yields ~ year, data = ., col = "blue", type = "l")
crop_ylds %>% filter(country == "ZMB" & crop == "Maize") %>%
  lines(yields ~ year, data = ., col = "red")

# change axes
dat <- crop_ylds[crop_ylds$country == "ZAF" & crop_ylds$crop == "Maize", ]
plot(yields ~ year, data = dat, pch = i, col = cols[i], axes = FALSE)
axis(side = 1, las = 2)
axis(side = 2, las = 2)

# multiple panels
par(mfrow = c(2, 1))
crop_ylds %>% filter(country == "ZAF" & crop == "Maize") %>%
  plot(yields ~ year, data = ., col = "blue", type = "l", ylim = c(0, 7))
crop_ylds %>% filter(country == "ZMB" & crop == "Maize") %>%
  plot(yields ~ year, data = ., col = "red", type = "l", ylim = c(0, 7))
```

Histogram

```
# basic
par(mar = c(3, 3, 1, 1))
crop_ylds %>% filter(crop == "Maize") %>%
  pull(yields) %>% hist(., breaks = seq(0, 7, 0.5), main = "Maize yields")

# side-by-side
par(mfrow = c(1, 2))
crop_ylds %>% filter(crop == "Maize" & country == "ZAF") %>%
  pull(yields) %>%
  hist(., breaks = seq(0, 7, 0.5), main = "Maize yields", xlim = c(0, 10))
crop_ylds %>% filter(crop == "Maize" & country == "ZMB") %>%
  pull(yields) %>%
  hist(., breaks = seq(0, 7, 0.5), main = "Maize yields", xlim = c(0, 10))
```

ggplot examples

scatters

```
# basic points
crop_ylds %>% filter(crop == "Maize") %>%
  ggplot() + geom_point(aes(x = year, y = yields))

# basic lines
crop_ylds %>% filter(crop == "Maize") %>%
  ggplot() + geom_line(mapping = aes(x = year, y = yields))

# multiple lines
crop_ylds %>% filter(crop == "Maize") %>%
  ggplot() + geom_line(aes(x = year, y = yields, color = country)) +
  scale_color_manual(values = c("green", "blue"))

# change axes/theme
crop_ylds %>% filter(crop == "Maize") %>%
  ggplot() + geom_line(aes(x = year, y = yields, color = country)) +
  scale_color_manual(values = c("green", "blue")) +
  scale_x_continuous(breaks = seq(1960, 2020, 5), expand = c(0, 0)) +
  theme(axis.text.x = element_text(angle = 90),
        panel.background = element_blank())
```

multi-panel

```
# from a single variable, facet_grid, facet_wrap
crop_ylds %>%
  ggplot() + geom_line(aes(x = year, y = yields, color = country)) +
  scale_color_manual(values = c("green", "blue")) +
  facet_grid(cols = vars(crop))

# multiple variables
p1 <- crop_ylds %>% filter(crop == "Maize") %>%
  ggplot() + geom_line(aes(x = year, y = yields, color = country)) +
  scale_color_manual(values = c("green", "blue"))
p2 <- crop_ylds %>% filter(crop == "Maize") %>%
  ggplot() + geom_line(aes(x = year, y = harv_area, color = country)) +
  scale_color_manual(values = c("green", "blue"))

gridExtra::grid.arrange(p1, p2, ncol = 2)
g1 <- cowplot::plot_grid(p1 + theme(legend.position = "none"),
                         p2 + theme(legend.position = "none"))
cowplot::plot_grid(g1, cowplot::get_legend(p1), rel_widths = c(2, 0.2))
```

Practical - data analysis and plotting

- Calculate the mean and standard deviations of crop yield by crop and country
- Use `graphics::plot` to create a red line plot of South African maize harvested area by year. Use `lines` to add Zambia maize yields to the same plot (in blue). Make sure they have the same scale (use an appropriate "ylim"). Make the axis labels nicer (e.g. "Harvested area (ha)", "Year")
- Use ggplot to plot the histograms of maize yields by country
- Use ggplot to plot the histograms of yields by country and crop (hint: you need to use `facet_grid` on the *crop* variable)
- Fit a regression model to maize yield, where year is the dependent variable. Use base R's `lm` function
- Plot the linear regression fit between Zambian and South Africa wheat yields, using `ggplot` and `geom_smooth`, with `method = "lm"`

Practical answers

Buried in the Rmarkdown

Spatial beginnings

- Spatial vector data and analyses

```
library(geospaar)
roads <- system.file("extdata/roads.shp", package = "geospaar") %>% st_read()
districts <- system.file("extdata/districts.shp", package = "geospaar") %>%
  st_read()
farmers <- system.file("extdata/farmer_spatial.csv", package = "geospaar") %>%
  read_csv() %>% st_as_sf(coords = c("x", "y"), crs = 4326)
```