Data Visualization and Maps II

HES 505 Fall 2023: Session 26

Carolyn Koehn

Objectives

By the end of today you should be able to:

- Generate complicated plot layouts without additional pre-processing
- Construct a map using `ggplot2` and `tmap`
- Combine vector and raster data in the same map

Building Choropleth Maps

`tidycensus` package

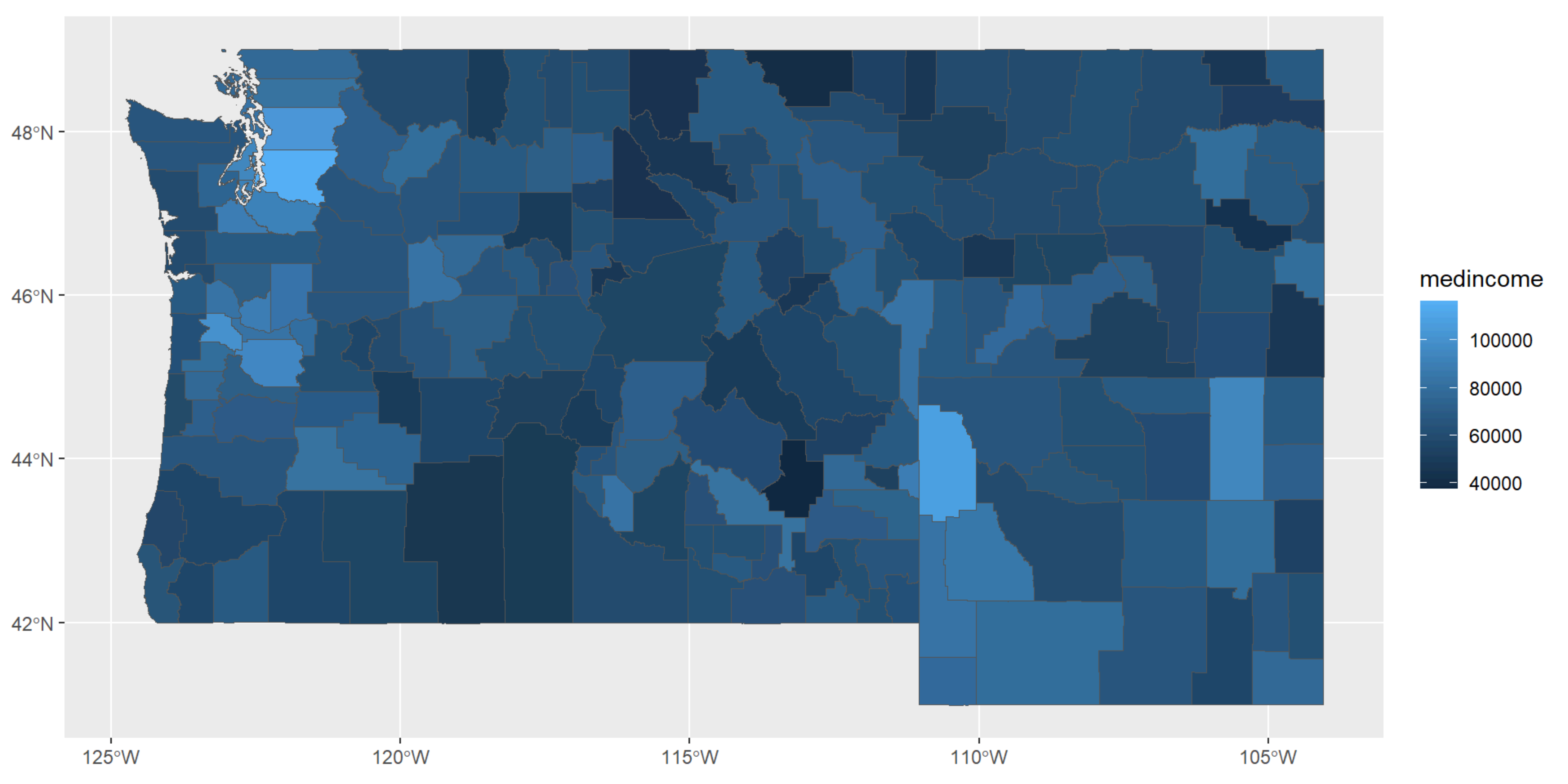
https://walker-data.com/tidycensus/articles/basic-usage.html

Using ggplot2

```r
cty.info <- get_acs(geography = "county",
                      variables = c(pop="B01003_001",
                                    medincome = "B19013_001"),
                      survey="acs5",
                      state = c("WA", "OR", "ID", "MT", "WY"),
                      geometry = TRUE, key = censkey, progress_bar=FALSE) %
  select(., -moe) %>%
  pivot_wider(
    names_from = "variable",
    values_from = "estimate"
  )

p <- ggplot(data=cty.info) +
  geom_sf(mapping=aes(fill=medincome))
```

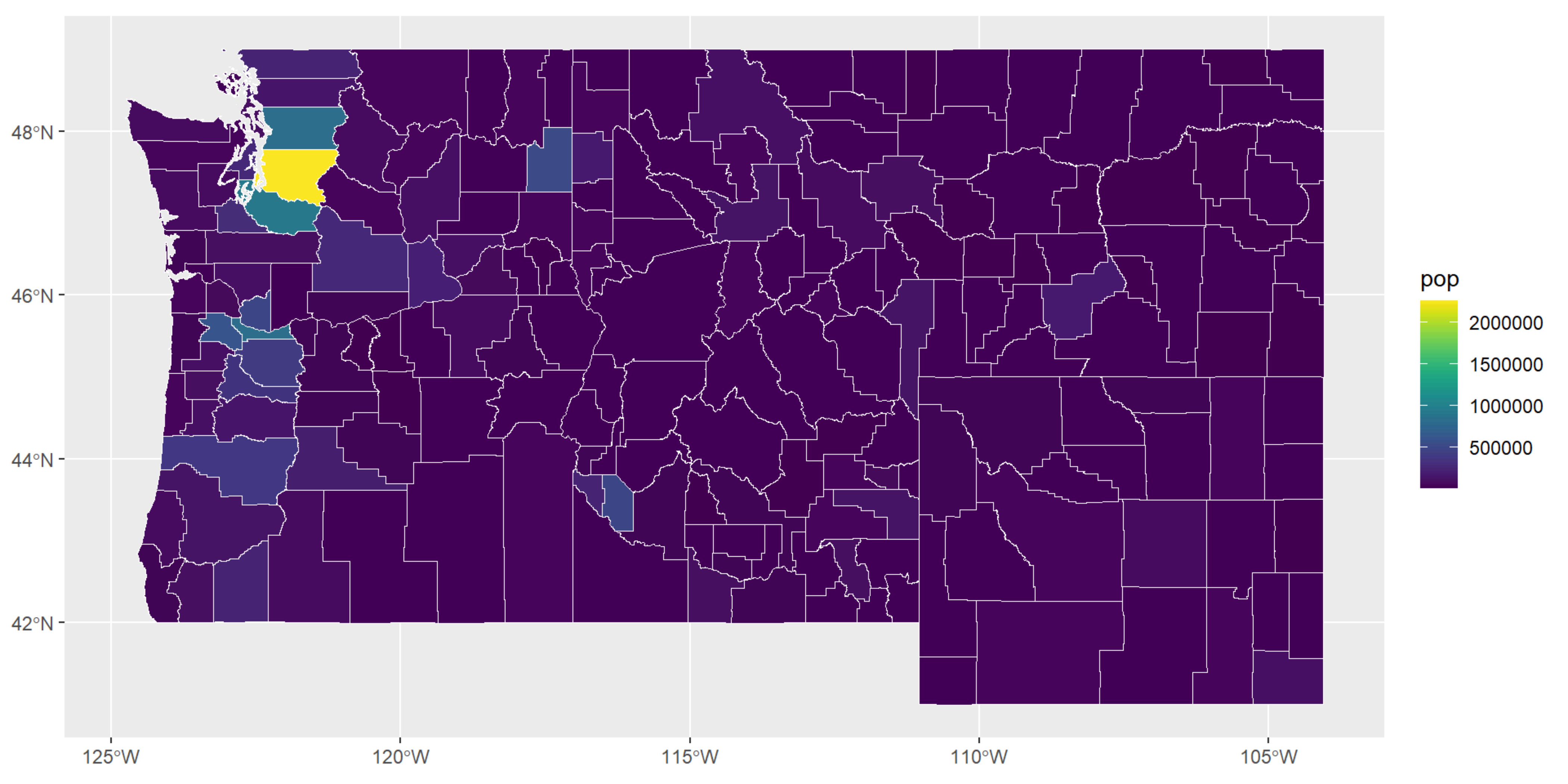
Static Maps with ggplot2



Changing aesthetics

```r
p <- ggplot(data=cty.info) +
  geom_sf(mapping=aes(fill=pop), color="white") +
  scale_fill_viridis()
```

Changing aesthetics

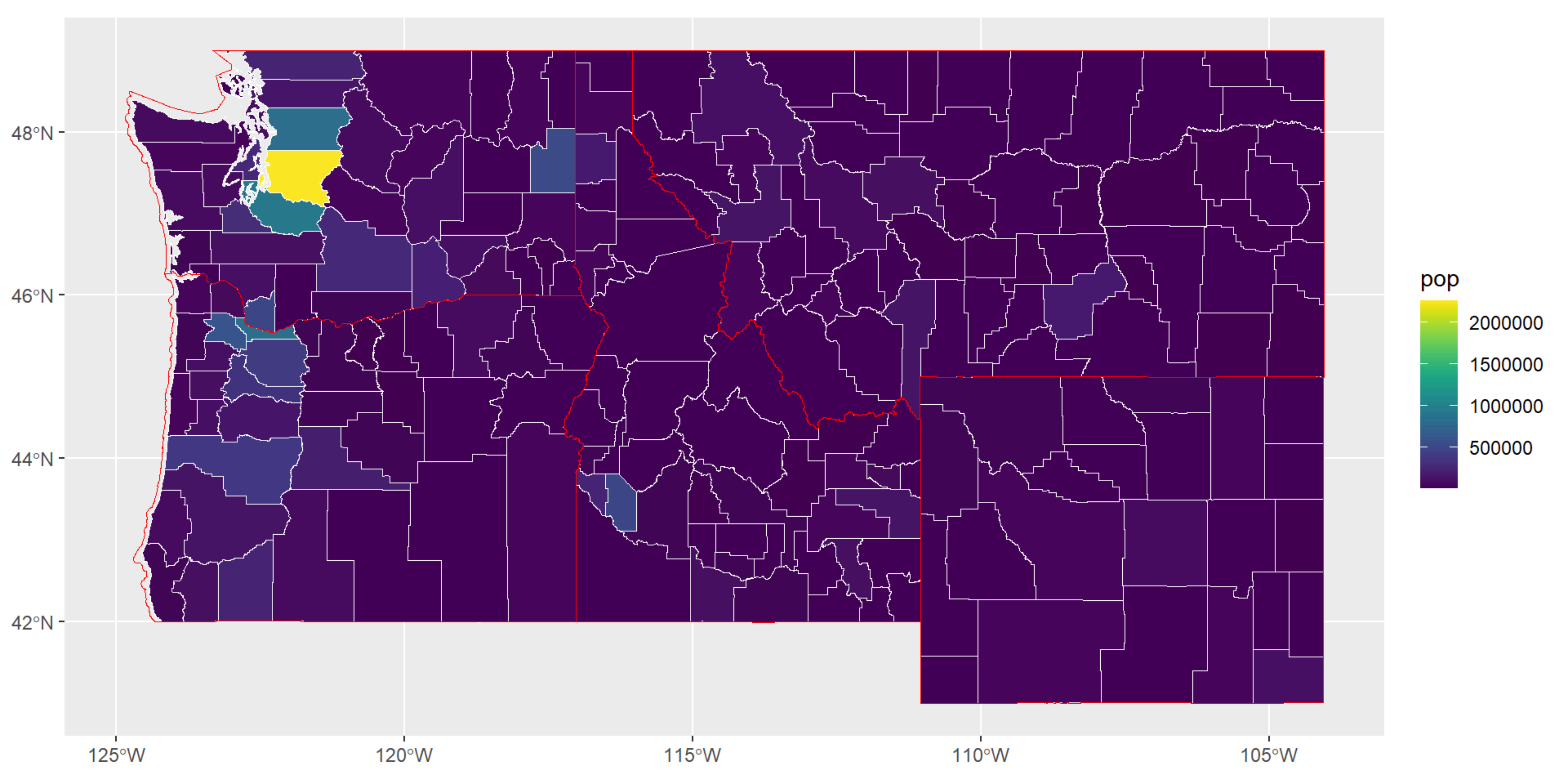


Adding layers

```r
st <- tigris::states(progress_bar=FALSE) %>%
  filter(., STUSPS %in% c("WA", "OR", "ID", "MT", "WY"))

p <- ggplot(data=cty.info) +
  geom_sf(mapping=aes(fill=pop), color="white") +
  geom_sf(data=st, fill=NA, color="red") +
  scale_fill_viridis()
```

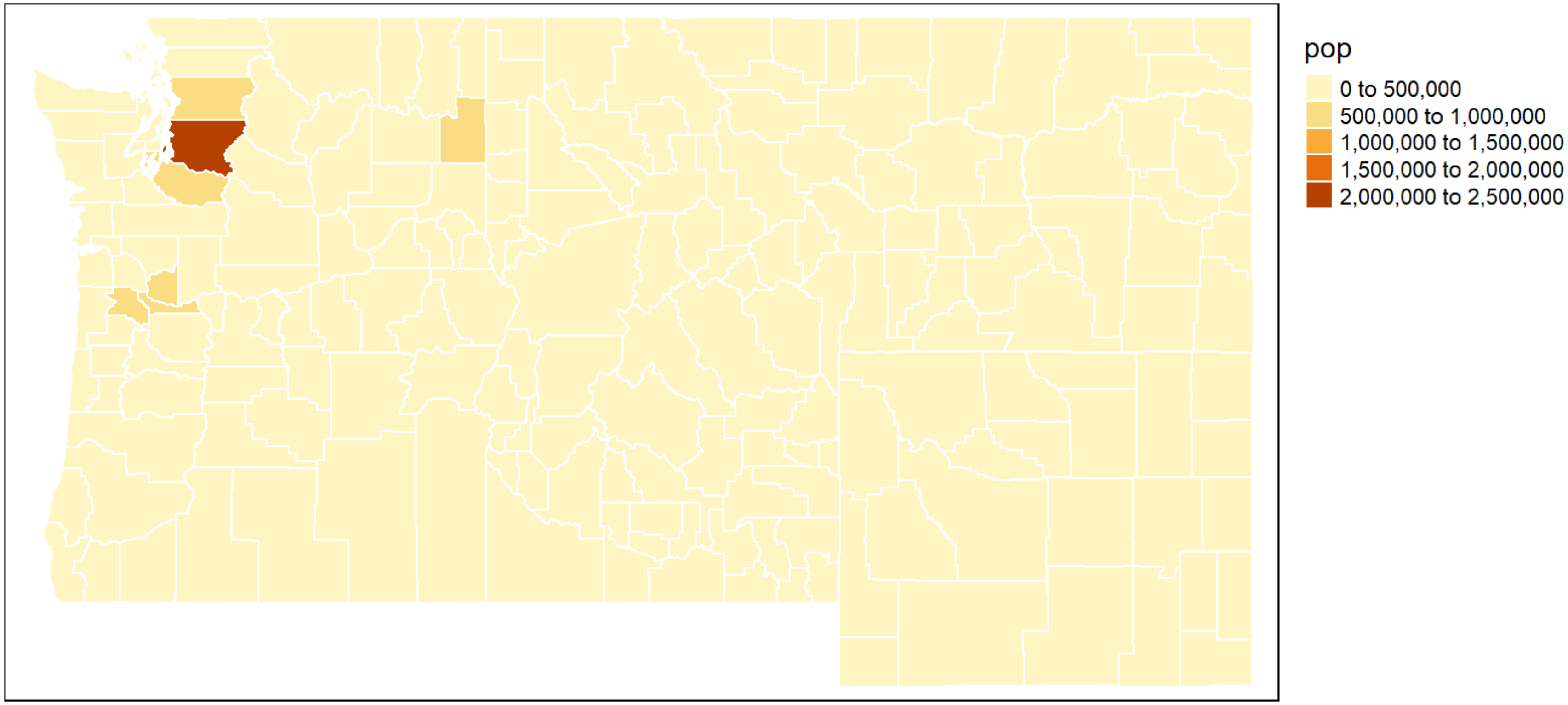
Adding layers



Using tmap

```r
pt <- tm_shape(cty.info) +
  tm_polygons(col = "pop",
              border.col = "white") +
  tm_legend(outside = TRUE)
```

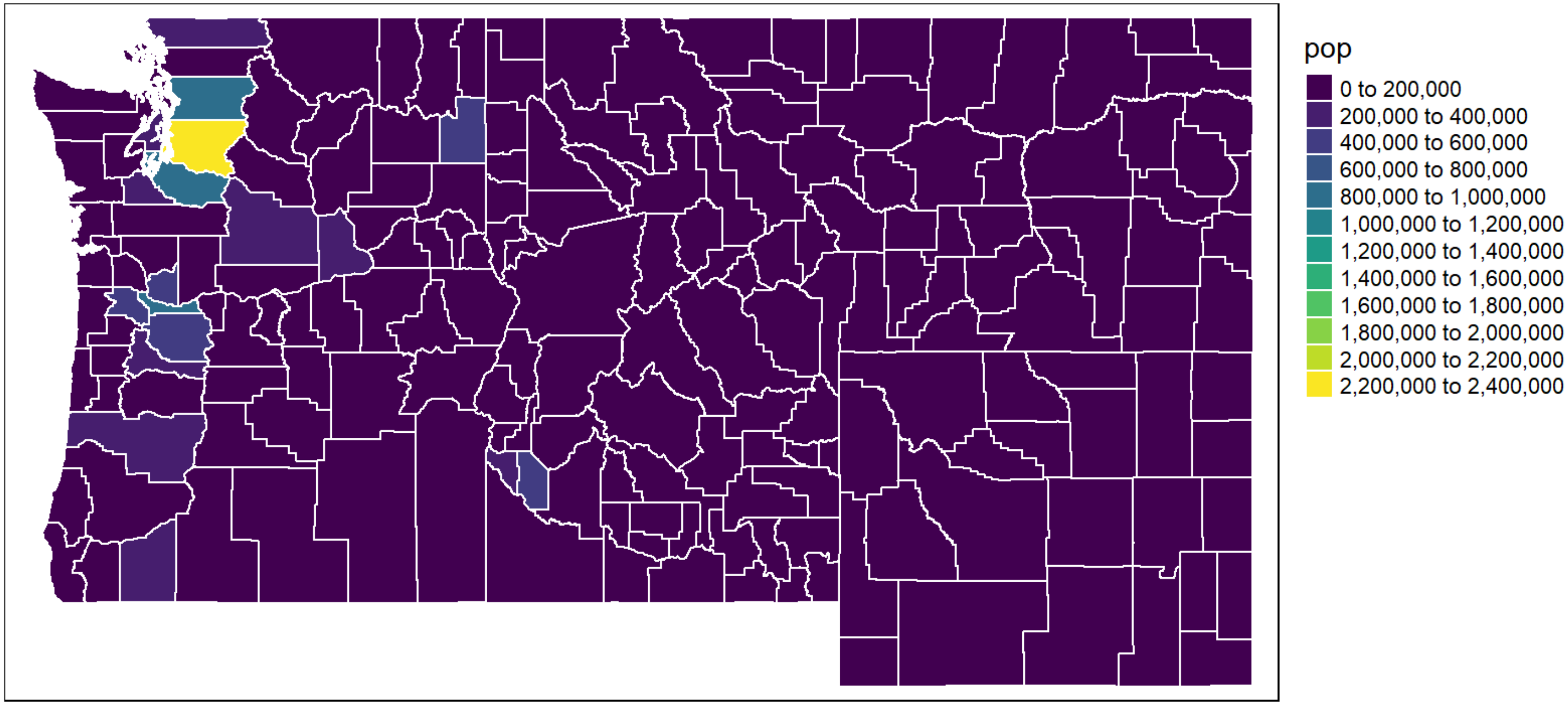
Using `tmap`


pop
0 to 500,000
500,000 to 1,000,000
1,000,000 to 1,500,000
1,500,000 to 2,000,000
2,000,000 to 2,500,000

Changing aesthetics

```r
pt <- tm_shape(cty.info) +
  tm_polygons(col = "pop", n=10,palette=viridis(10),
              border.col = "white") +
  tm_legend(outside = TRUE)
```

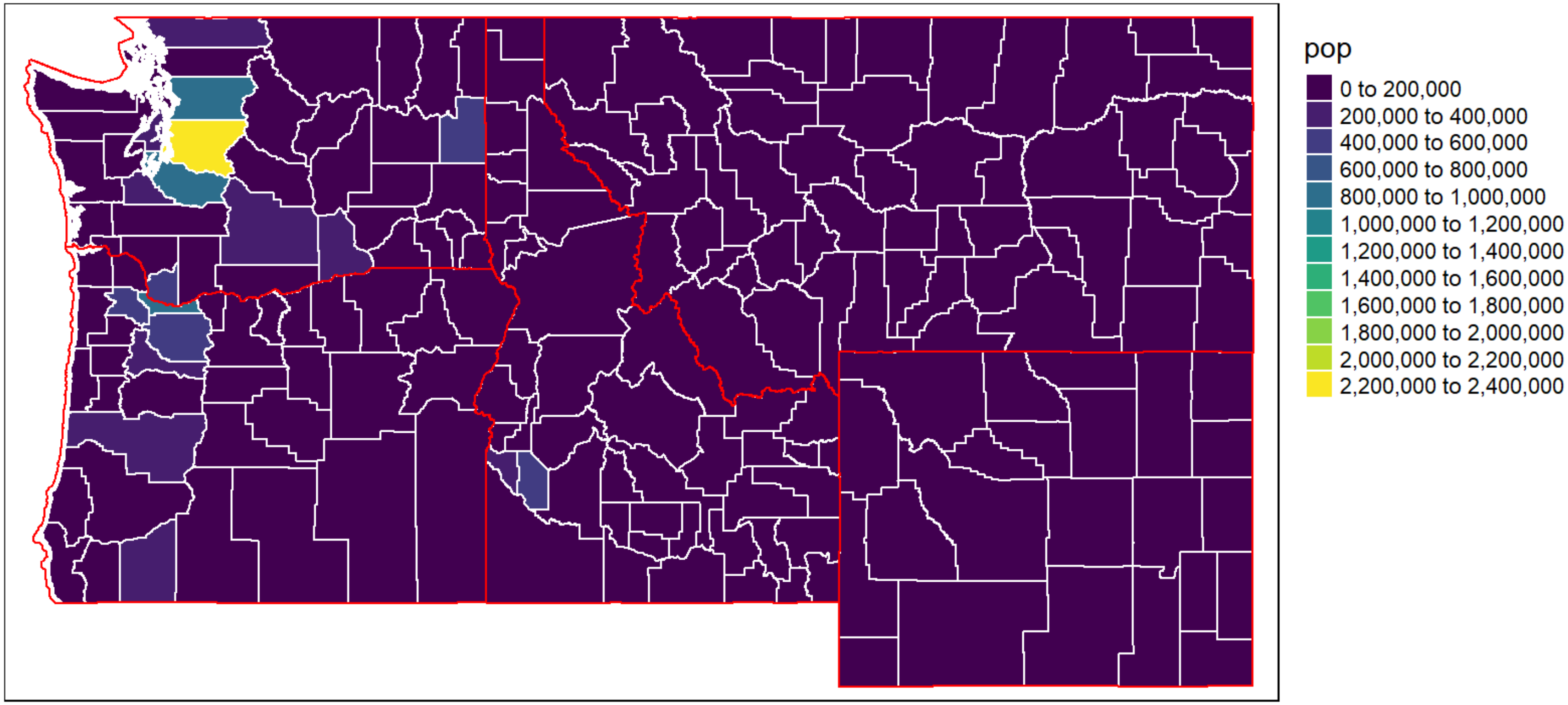
Changing aesthetics



Adding layers

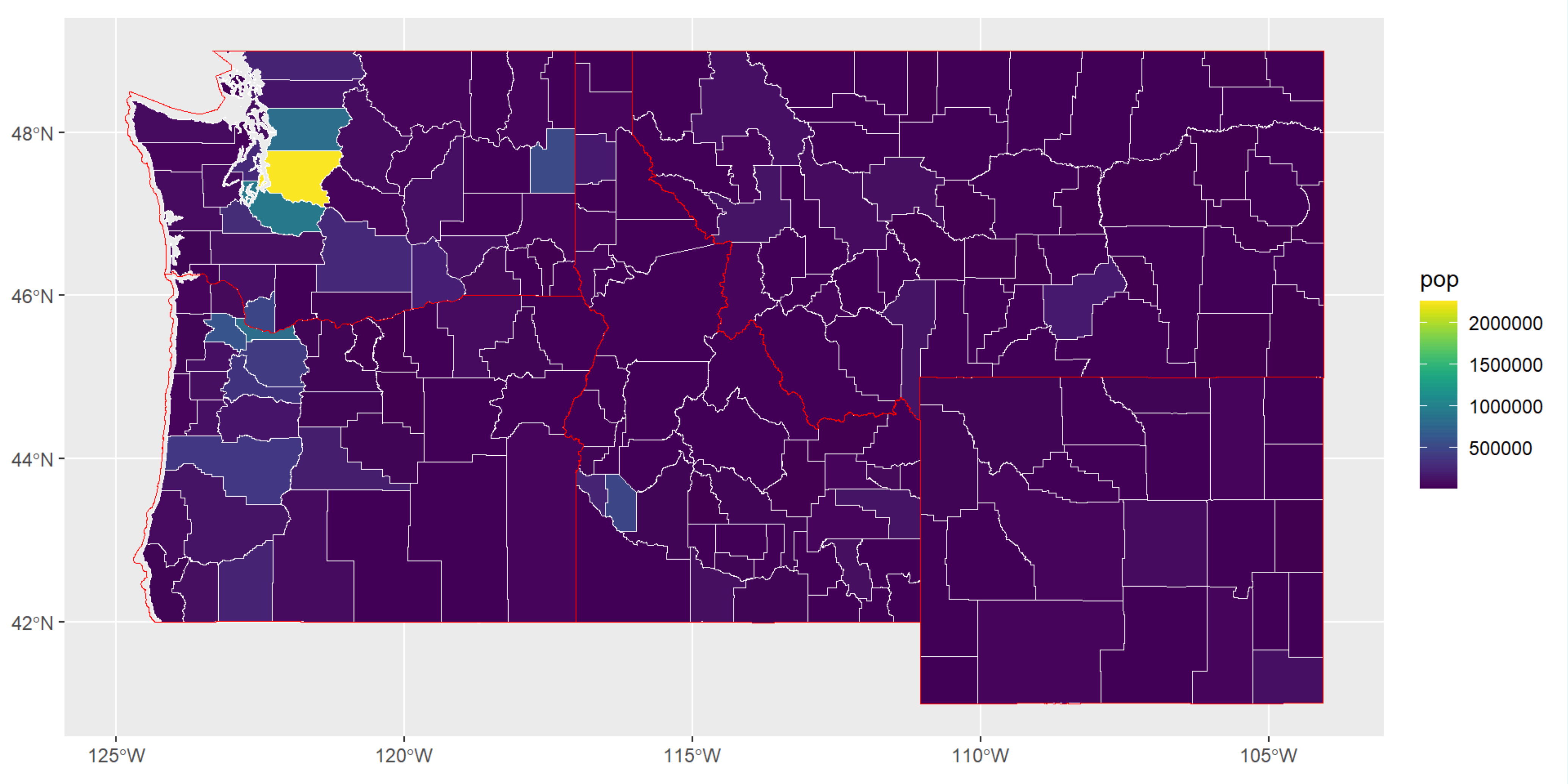
```r
pt <- tm_shape(cty.info) +
  tm_polygons(col = "pop", n=10,palette=viridis(10),
                   border.col = "white") +
  tm_shape(st) +
  tm_borders("red") +
  tm_legend(outside = TRUE)
```

Adding layers



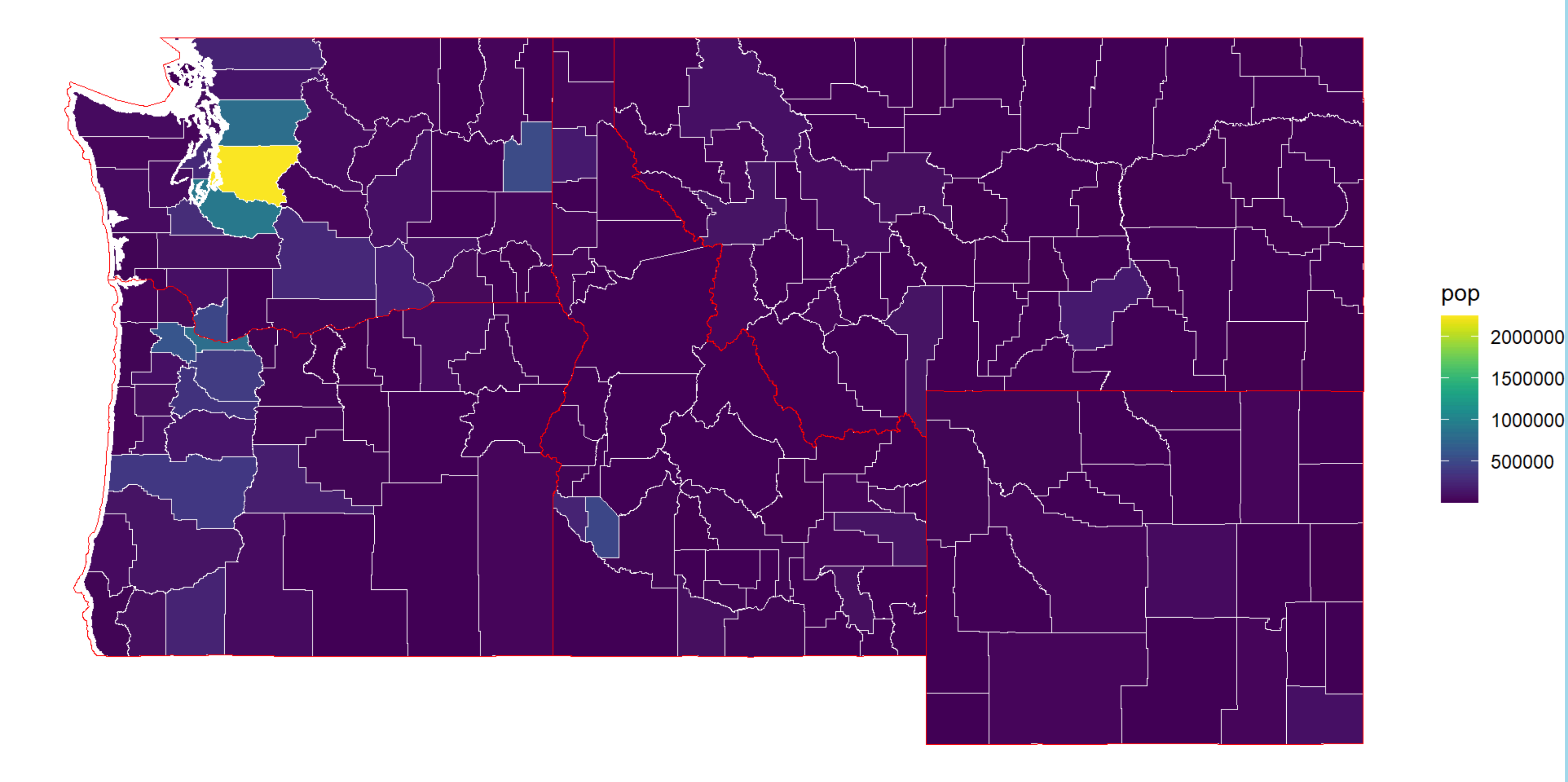
Themes

```
p
```


48°N
46°N
44°N
42°N
125°W
120°W
115°W
110°W
105°W
pop
2000000
1500000
1000000
500000

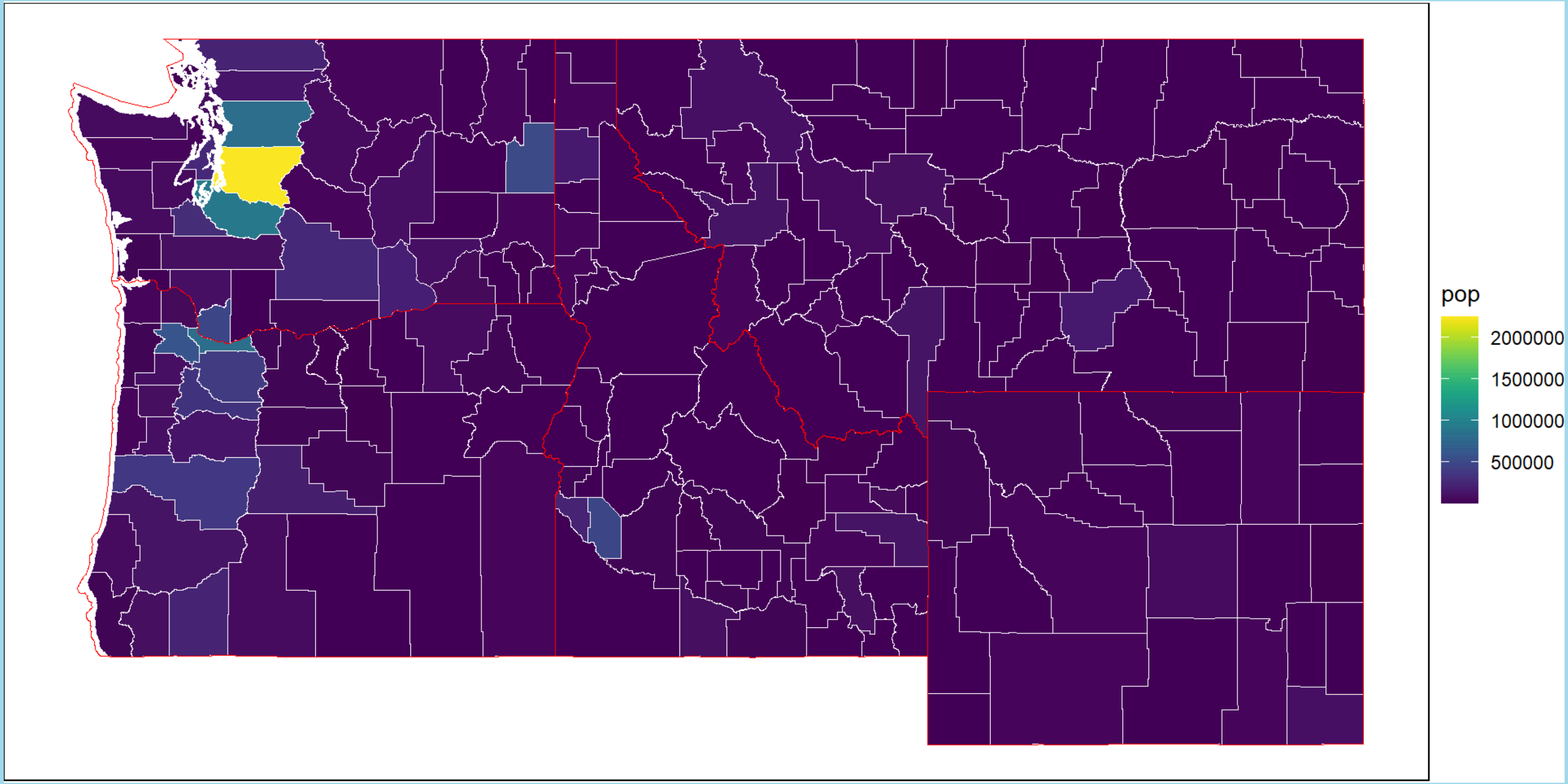
Themes

```
p +
  theme_void()
```



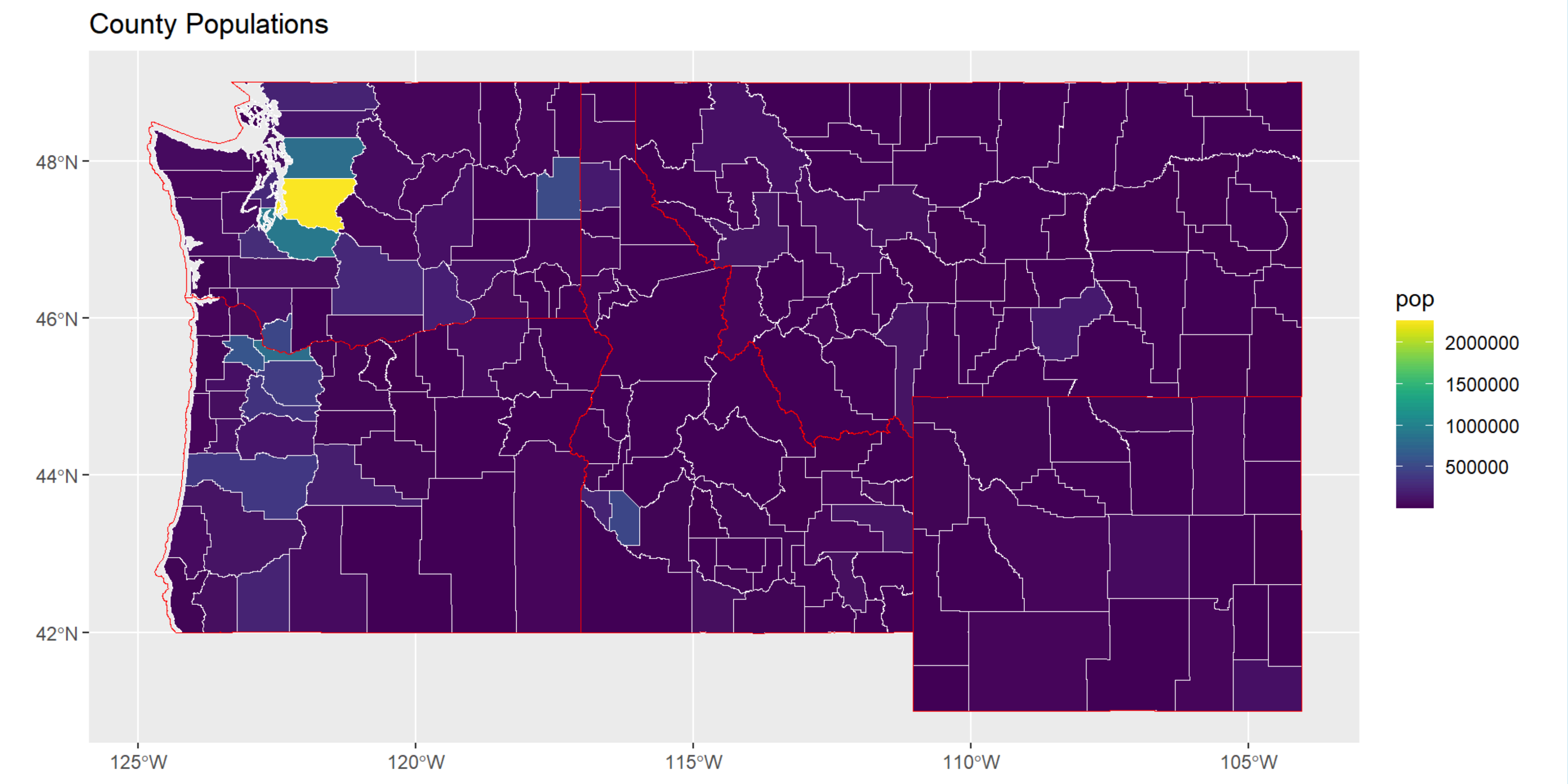
Themes

```r
p +
  theme_void() +
  theme(panel.background = element_rect(fill="white", color="black"))
```



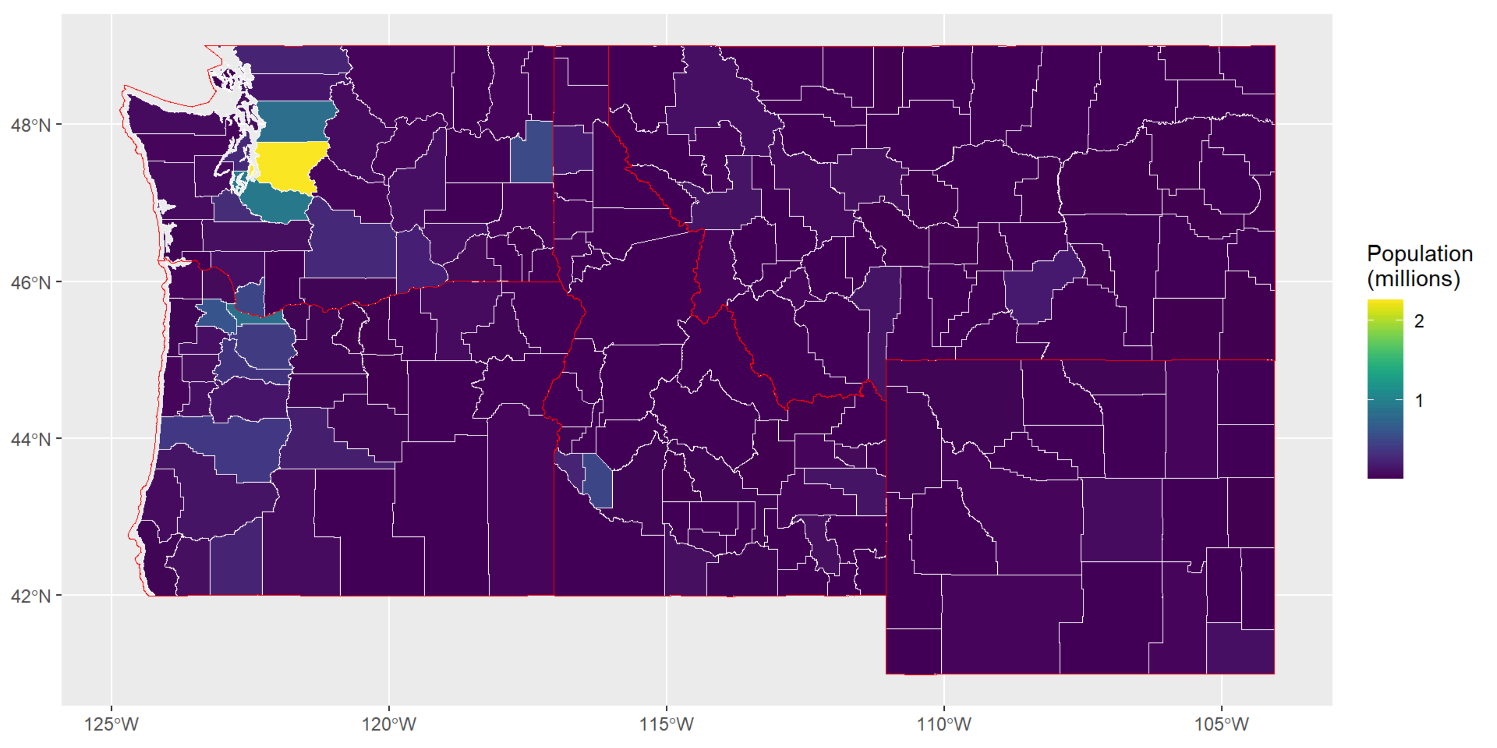
Labels

```r
p +
  labs(title = "County Populations")
```



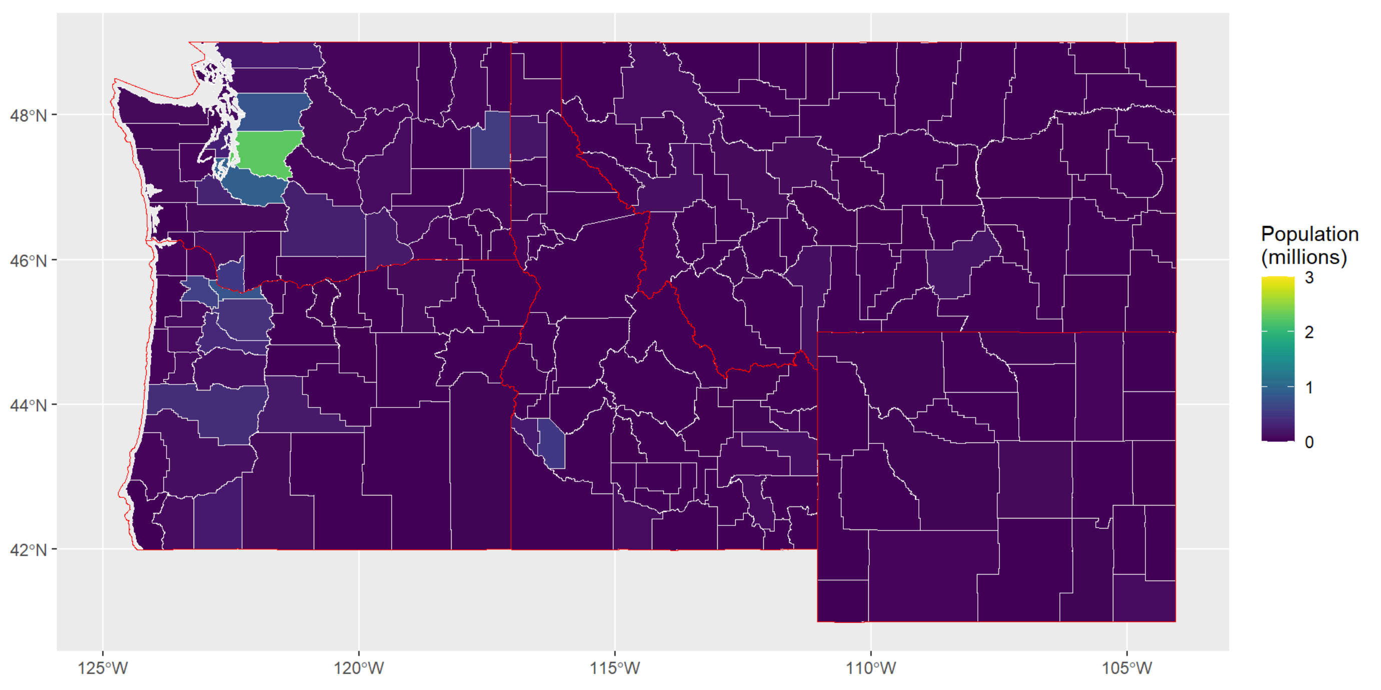
Legend customization

```r
ggplot(data=cty.info) +
  geom_sf(mapping=aes(fill=pop), color="white") +
  geom_sf(data=st, fill=NA, color="red") +
  scale_fill_viridis(name = "Population\n(millions)",
                     breaks = seq(from=0, to=3e6, by=1e6),
                     labels = seq(from=0, to=3e6, by=1e6)/1e6)
```



Legend customization

```r
ggplot(data=cty.info) +
  geom_sf(mapping=aes(fill=pop), color="white") +
  geom_sf(data=st, fill=NA, color="red") +
  scale_fill_viridis(name = "Population\n(millions)",
                     limits = c(0, 3e6),
                     breaks = seq(from=0, to=3e6, by=1e6),
                     labels = seq(from=0, to=3e6, by=1e6)/1e6)
```



Rasters in `ggplot2`

Convert raster to tidy format

To visualize rasters in `ggplot2`, they need to be converted to a dataframe.

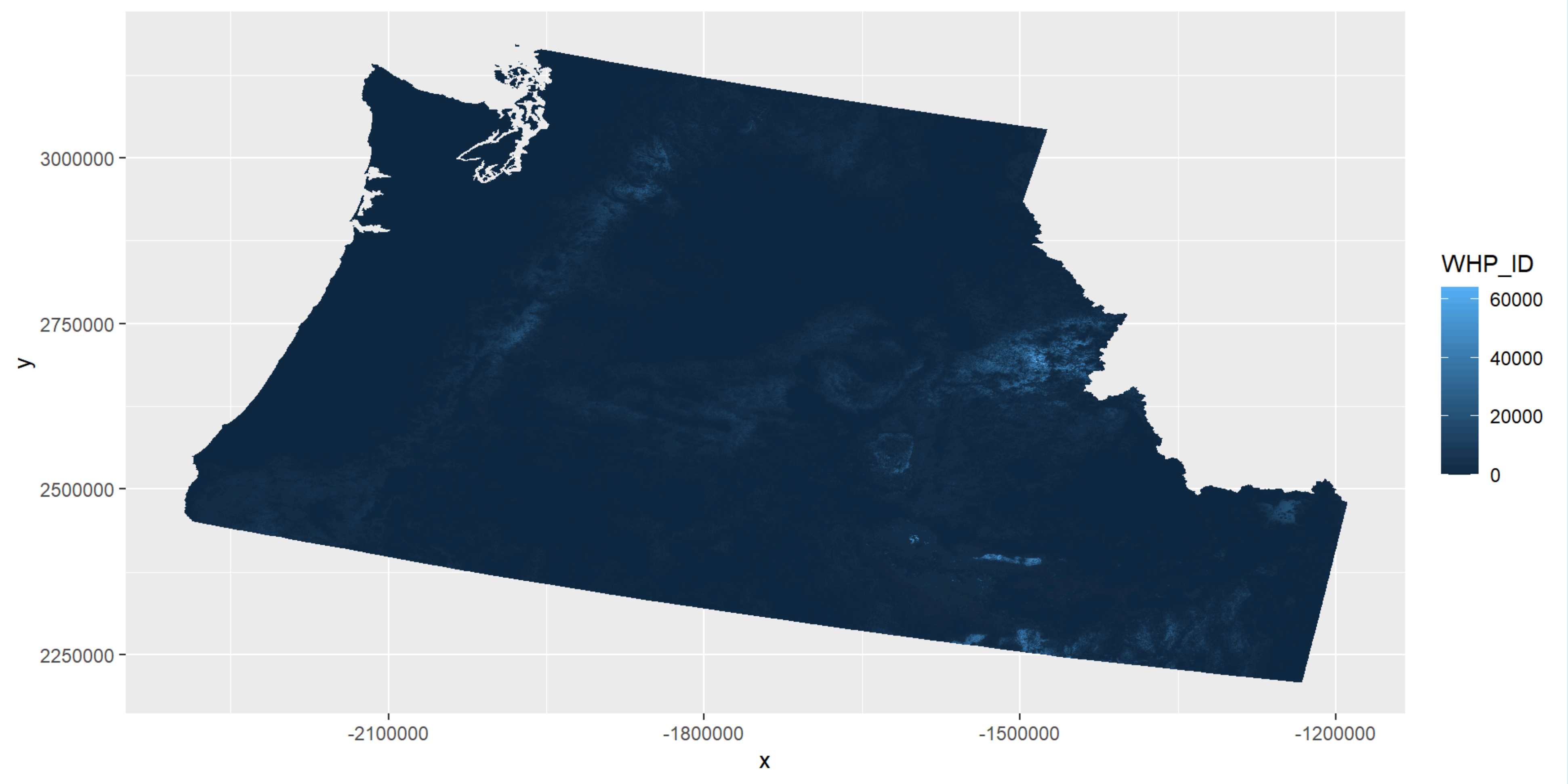
```r
fire.haz <- rast("/opt/data/data/assignment01/wildfire_hazard_agg.tif")

fire.haz_df <- as.data.frame(fire.haz)

str(fire.haz_df)
```

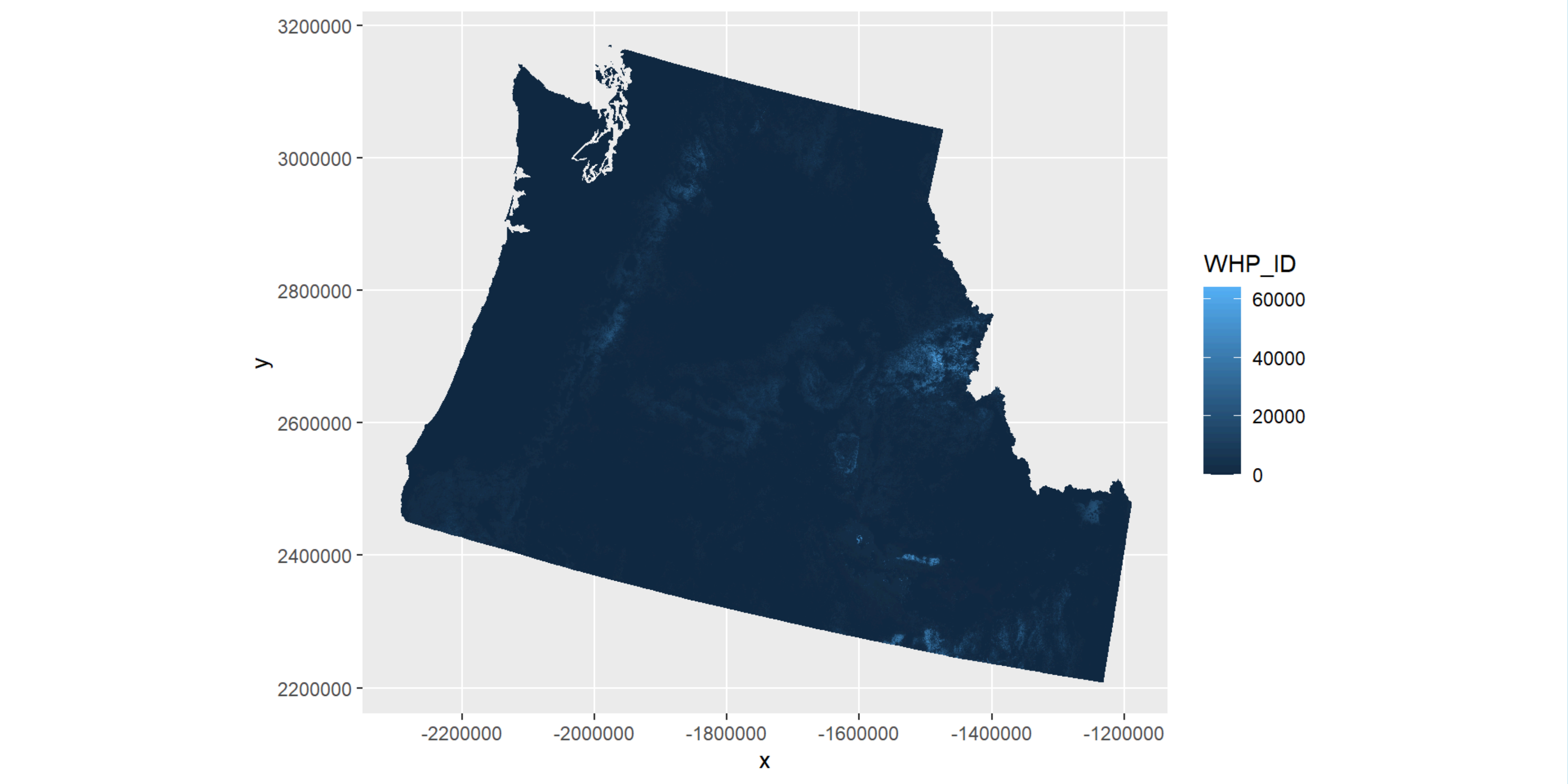
Use geom_raster

```r
ggplot(data = fire.haz_df, aes(x=x, y=y, fill = WHP_ID)) +
  geom_raster()
```



Coordinate system

```r
ggplot(data = fire.haz_df, aes(x=x, y=y, fill = WHP_ID)) +
  geom_raster() +
  coord_sf(default_crs = crs(fire.haz))
```



Layering rasters and vectors

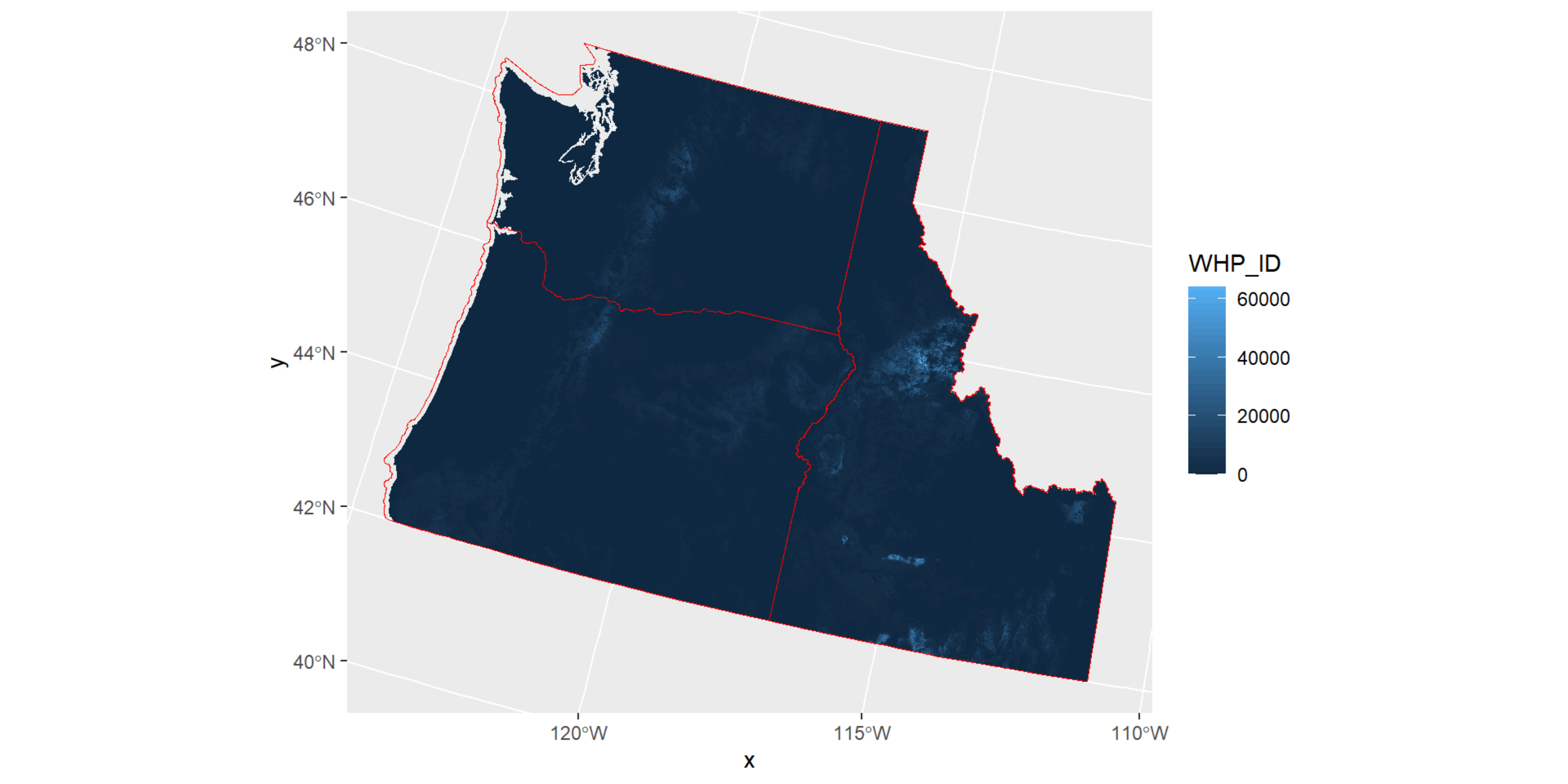
```r
states <- tigris::states(progress_bar=FALSE) %>%
  filter(., STUSPS %in% c("WA", "OR", "ID")) %>%
  st_transform(crs = st_crs(fire.haz))
```

```
Retrieving data for the year 2021
```

```r
p2 <- ggplot(data = fire.haz_df, aes(x=x, y=y, fill = WHP_ID)) +
  geom_raster() +
  geom_sf(data = states, fill=NA, color="red", inherit.aes = FALSE) +
  coord_sf(default_crs = crs(fire.haz))

p3 <- ggplot() +
  geom_raster(data = fire.haz_df, aes(x=x, y=y, fill = WHP_ID)) +
  geom_sf(data = states, fill=NA, color="red") +
  coord_sf(default_crs = crs(fire.haz))
```

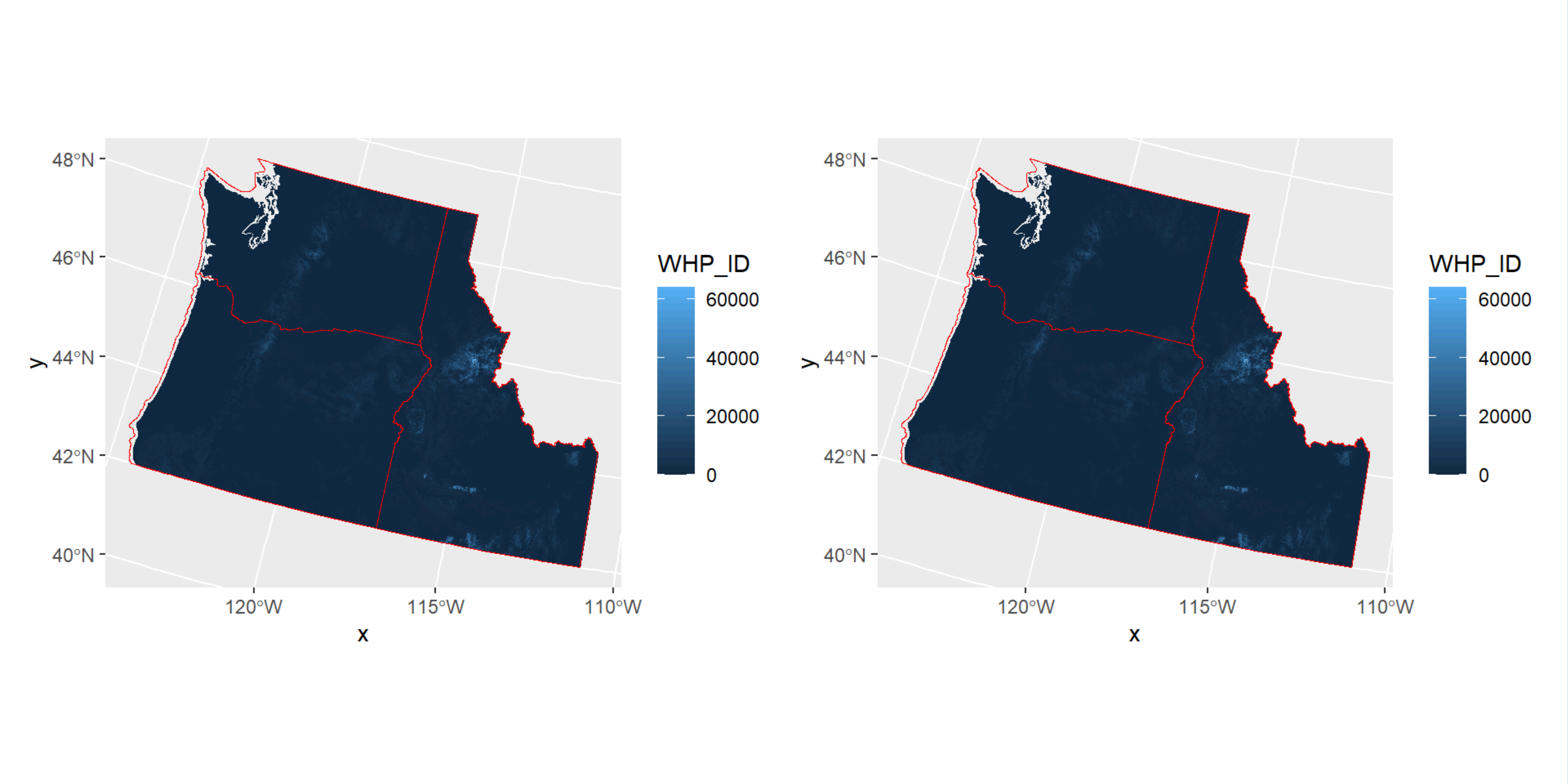
Layering rasters and vectors



Complicated layouts with patchwork

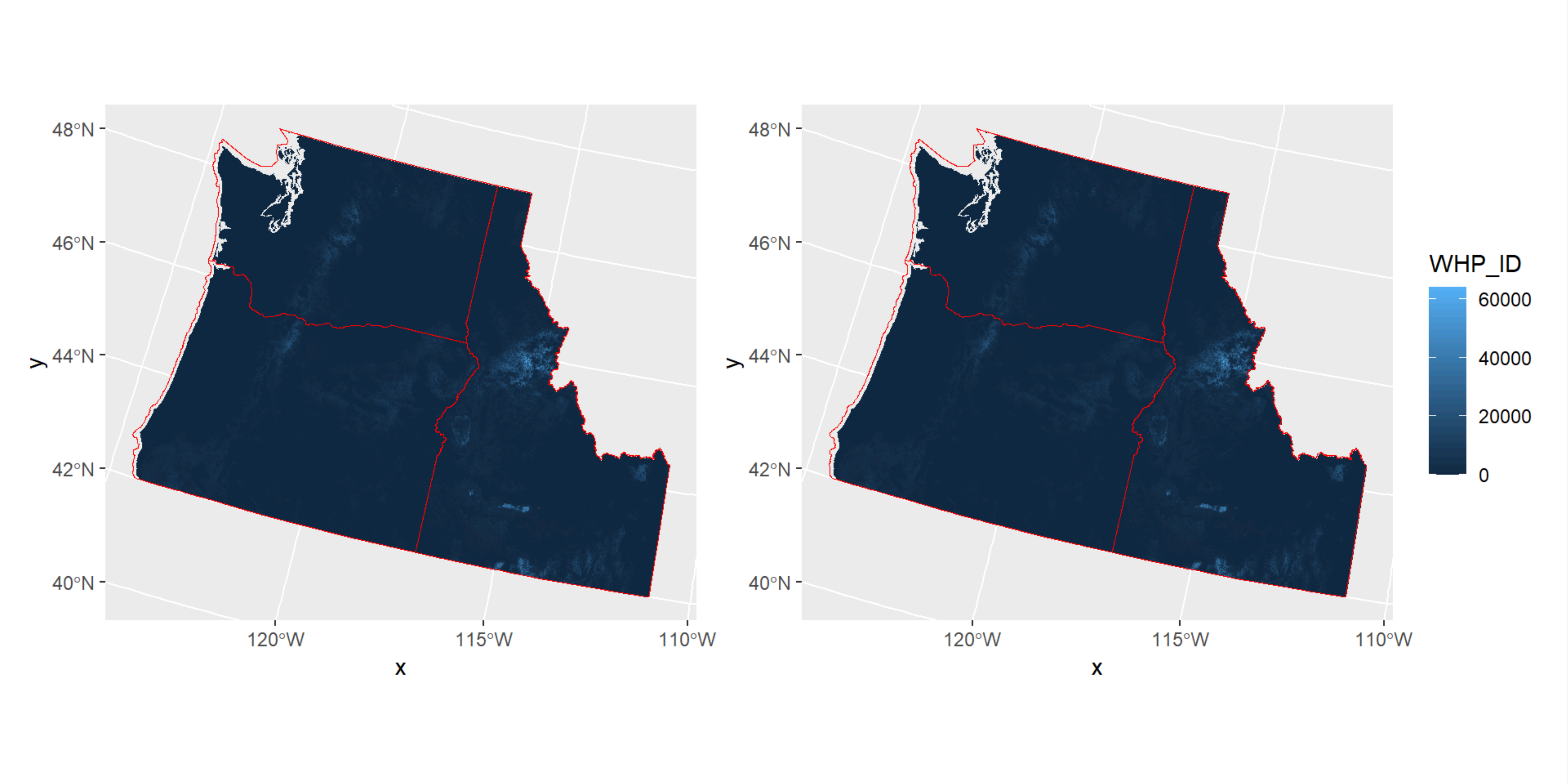
Subplots

```
p2+p3
```



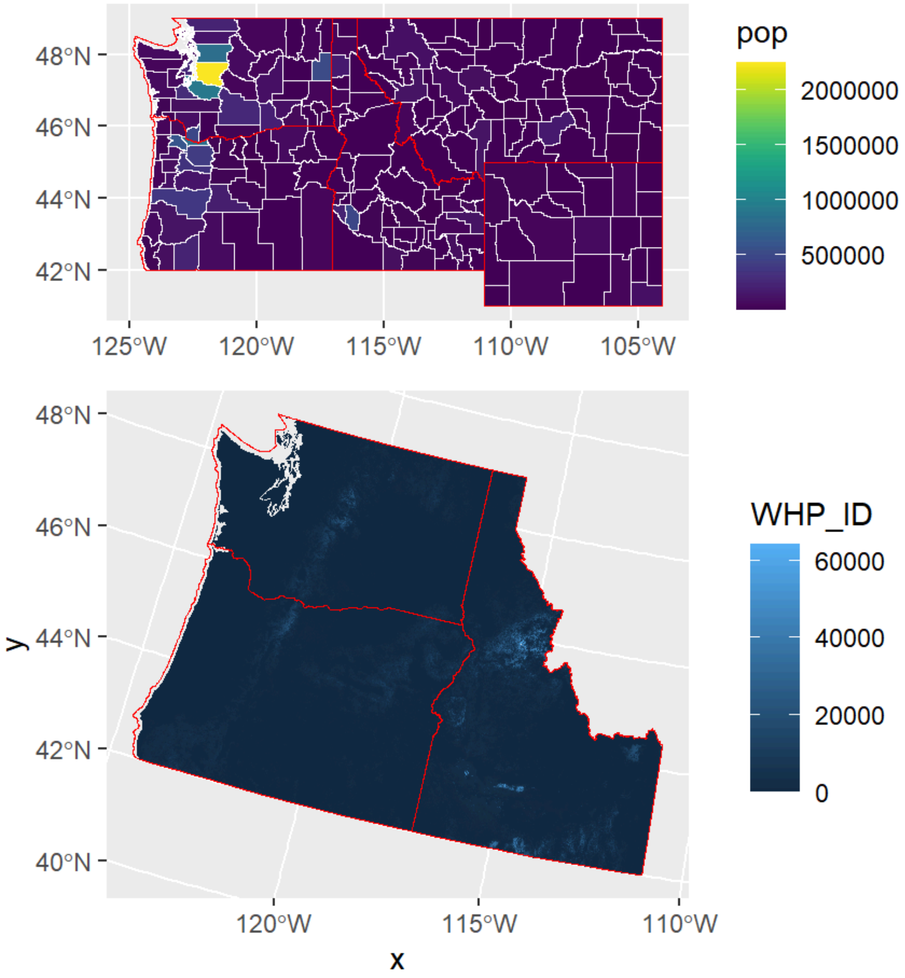
Combine legends

```r
p2+p3 + plot_layout(guides="collect")
```



Change layout

```r
p+p3 + plot_layout(nrow=2)
```



Map insets

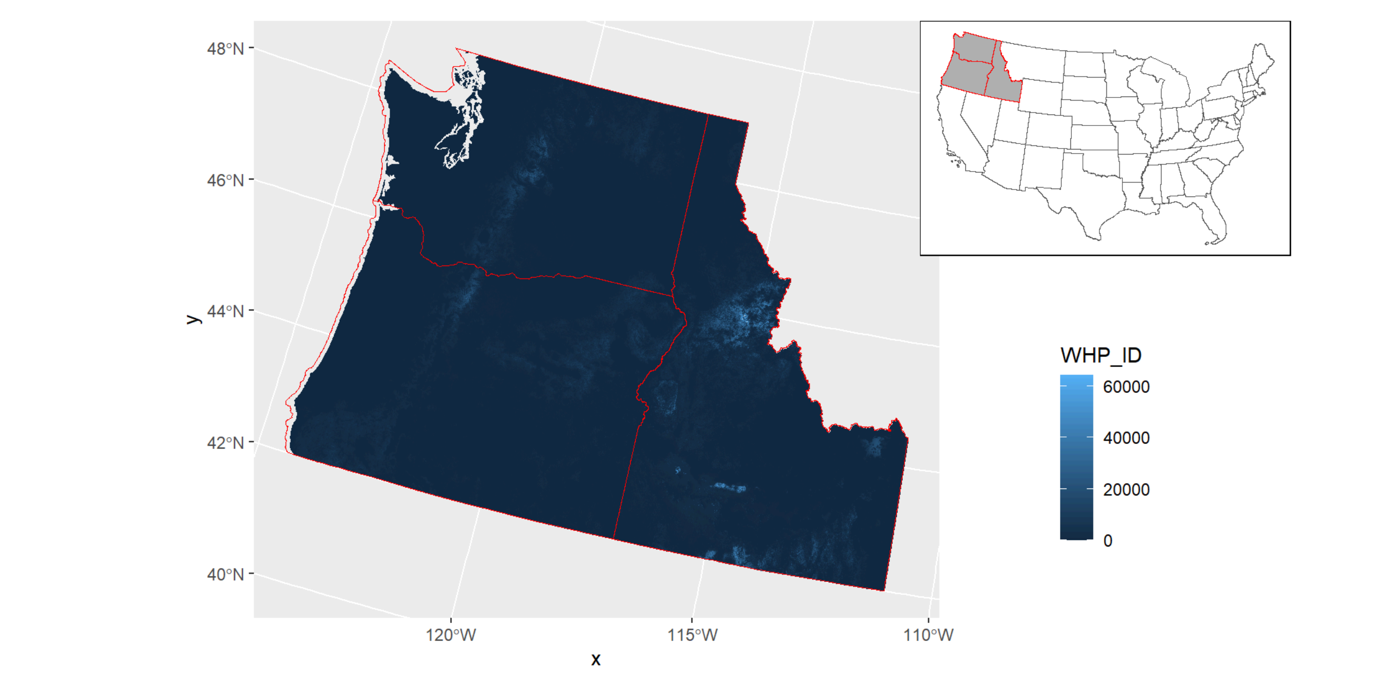
```r
conus <- tigris::states() %>%
  filter(!(DIVISION == 0 | STUSPS %in% c("HI", "AK"))) %>%
  st_transform(st_crs(fire.haz))

inset <- ggplot(data = conus) +
  geom_sf(fill="white") +
  geom_sf(data = filter(conus, STUSPS %in% c("ID", "OR", "WA")),
          fill = "gray70", color="red") +
  theme_void() +
  theme(panel.background = element_rect(fill="white", color="black")) +
  coord_sf()
```

Map insets

```r
layout <- c(
  patchwork::area(t = 1, l = 1, b = 5, r = 4),
  patchwork::area(t = 1, l = 4, b = 2, r = 5),
  patchwork::area(t = 3, l = 4, b = 5, r = 5)
)

p2 + inset + guide_area() + plot_layout(design=layout, guides='collect')
```


48°N
46°N
44°N
42°N
40°N
y
120°W
115°W
110°W
x
WHP_ID
60000
40000
20000
0