

Assignment 1

Maps Replication

Sebastian Dong Uk Paik Sohn

Michael Duarte Gonçalves

Gal · la Gelpí Buxadé

Maria Victoria Suriel Nuñez

January 28, 2026

Contents

1 Part 1: Fried & Lagakos (2021) — Figure 4	3
1.1 What the map shows	3
1.2 Data sources	3
1.3 Why our map differs	3
1.4 Setup	3
1.5 Processing the survey data	5
1.6 Building the map	6
2 Part 2: Pellegrina & Sotelo (2025) — Figure 2	9
2.1 What the map shows	9
2.2 Data sources	9
2.3 Why our map differs	9
2.4 Setup	9
2.5 Loading the Data	9
2.6 Processing the data	10
2.7 Plotting the data	10
3 Part 3: Morten & Oliveira (2024) — Figure 1	15
3.1 What the map shows	15
3.2 Data sources	15
3.3 Why our map differs	15
3.4 Setup	15
3.5 Loading the Data	15
3.6 Filtering the Data	16
3.7 Building the Map	16
4 Part 4: Mettetal (2019) — Figure 2	20
4.1 What the map shows	20
4.2 Data sources	20
4.3 Why our map differs	20
4.4 Setup	20
4.5 Loading the Data	20
4.6 Align the coordinates	22
4.7 Carving elevation along rivers	22
4.8 Compute the slope and aggregate it by district	22
4.9 Classifying Districts	23
4.10 Building the Map	23

5 Part 5: Balboni (2021) — Figure 3	26
5.1 What the map shows	26
5.2 Data sources	26
5.3 Why our map differs	26
5.4 Setup	26
5.5 Map of 2015	27
5.6 Map of 2026	27
5.7 Building the Maps	28
6 References	33

1 Part 1: Fried & Lagakos (2021) — Figure 4

1.1 What the map shows

Figure 4 displays the location of surveyed villages in Ethiopia, overlaid with the country’s road network and electricity grid. The blue gradient represents population density, roads appear as black lines (thicker for major highways), the high-voltage grid is shown in red, and hollow circles mark each village from the ERSS sample. The authors use this map to show where their data comes from and to highlight the variation they exploit—some villages sit near power lines while others are far away.

1.2 Data sources

We gathered all layers from public sources:

- Electricity grid — energydata.info
- Power plants — energydata.info
- Administrative boundaries — Africa GeoPortal
- Roads — Humanitarian Data Exchange (OpenStreetMap extract)
- ERSS Wave 1 — World Bank Microdata (2011/12)
- ERSS Wave 2 — World Bank Microdata (2013/14)

1.3 Why our map differs

We could not fully replicate the original for a two reasons. First, we do not know the exact vintage of the road network they used, so we relied on OpenStreetMap data from HDX. Also, we omitted geographic labels (Somalia, Djibouti, etc.) to keep the focus on Ethiopia.

1.4 Setup

```
if (!require("pacman")) install.packages("pacman")

pacman::p_load(
  sf, dplyr, readr, haven, stringr,
  ggplot2, ggnewscale, stars, MASS,
  knitr, kableExtra
)

zip_w1 <- "./data/Fried_Lagakos_2021/ETH_2011_ERSS_v02_M_CSV.zip"
zip_w2 <- "./data/Fried_Lagakos_2021/ETH_2013_ESS_v03_M_STATA.zip"
zip_admin <- "./data/Fried_Lagakos_2021/Ethiopia_AdminBoundaries-shp.zip"
zip_roads <- "./data/Fried_Lagakos_2021/hotosm_eth_roads_lines_shp.zip"
zip_grid <- "./data/Fried_Lagakos_2021/ethiopia-electricity-transmission-network.zip"
zip_plants <- "./data/Fried_Lagakos_2021/eth_powerplants.zip"

tmp_w1 <- file.path(tempdir(), "w1")
tmp_w2 <- file.path(tempdir(), "w2")
tmp_admin <- file.path(tempdir(), "admin")
tmp_roads <- file.path(tempdir(), "roads")
tmp_grid <- file.path(tempdir(), "grid")
tmp_plants <- file.path(tempdir(), "plants")

lapply(
  c(tmp_w1, tmp_w2, tmp_admin, tmp_roads, tmp_grid, tmp_plants),
```

```

dir.create,
showWarnings = FALSE
)

## [[1]]
## [1] TRUE
##
## [[2]]
## [1] TRUE
##
## [[3]]
## [1] TRUE
##
## [[4]]
## [1] TRUE
##
## [[5]]
## [1] TRUE
##
## [[6]]
## [1] TRUE

unzip(zip_w1, exdir = tmp_w1)
unzip(zip_w2, exdir = tmp_w2)
unzip(zip_admin, exdir = tmp_admin)
unzip(zip_roads, exdir = tmp_roads)
unzip(zip_grid, exdir = tmp_grid)
unzip(zip_plants, exdir = tmp_plants)

w1_csv <- file.path(
  tmp_w1,
  "ETH_2011_ERSS_v02_M_CSV",
  "pub_eth_householdgeovariables_y1.csv"
)
w2_dta <- file.path(tmp_w2, "Pub_ETH_HouseholdGeovars_Y2.dta")

find_shp <- function(dir) {
  list.files(dir, pattern = "\\\\shp$", full.names = TRUE, recursive = TRUE)[1]
}

admin_shp <- find_shp(tmp_admin)
roads_shp <- find_shp(tmp_roads)
grid_shp <- find_shp(tmp_grid)
plants_shp <- find_shp(tmp_plants)

admin <- st_read(admin_shp, quiet = TRUE)
roads <- st_read(roads_shp, quiet = TRUE)
grid <- st_read(grid_shp, quiet = TRUE)
plants <- st_read(plants_shp, quiet = TRUE)

w1 <- read_csv(w1_csv, show_col_types = FALSE) %>%
  mutate(
    lat_dd_mod = as.numeric(LAT_DD_MOD),
    lon_dd_mod = as.numeric(LON_DD_MOD),
    h2011_tot = as.numeric(h2011_tot),

```

```

qa_type = as.integer(qa_type),
ea_id11 = str_pad(ea_id, width = 11, side = "left", pad = "0")
)

w2 <- read_dta(w2_dta) %>%
  mutate(
    lat_dd_mod = as.numeric(lat_dd_mod),
    lon_dd_mod = as.numeric(lon_dd_mod),
    ea_id11 = str_pad(ea_id, width = 11, side = "left", pad = "0")
)

```

1.5 Processing the survey data

```

# Grouping by all the families that live in the same Enumeration Area, as in the paper.
# Creating the coodenades of each area, it's mean population.
w1_ea <- w1 %>%
  group_by(ea_id11) %>%
  summarise(
    lat = mean(lat_dd_mod, na.rm = TRUE),
    lon = mean(lon_dd_mod, na.rm = TRUE),
    qa_type = first(qa_type),
    pop2011 = mean(h2011_tot, na.rm = TRUE),
    .groups = "drop"
  )

# Grouping by all the families that live in the same Enumeration Area, as in the paper,
# and creating the coodenades of each area.
w2_ea <- w2 %>%
  group_by(ea_id11) %>%
  summarise(
    lat = mean(lat_dd_mod, na.rm = TRUE),
    lon = mean(lon_dd_mod, na.rm = TRUE),
    .groups = "drop"
  )

cat("Wave 1:", nrow(w1_ea), "villages\n")

## Wave 1: 333 villages
cat("Wave 2:", nrow(w2_ea), "villages\n")

## Wave 2: 334 villages

# Restricting only to the villages in both times, as the paper does. That way you have 2
# periods of data for each village.
panel_ids <- intersect(w1_ea$ea_id11, w2_ea$ea_id11)
panel <- w1_ea %>% filter(ea_id11 %in% panel_ids)
cat("Panel villages:", length(panel_ids), "\n")

## Panel villages: 333

# Define a function to compute distances in the earth as we will need it.
haversine_km <- function(lon1, lat1, lon2, lat2) {
  r <- 6371
  rad <- pi / 180

```

```

d_lat <- (lat2 - lat1) * rad
d_lon <- (lon2 - lon1) * rad
a <- sin(d_lat / 2)^2 + cos(lat1 * rad) * cos(lat2 * rad) * sin(d_lon / 2)^2
2 * r * asin(sqrt(a))
}

# Add the distance from Addis Abeba as a column to the data
panel <- panel %>%
  mutate(dist_addis_km = haversine_km(lon, lat, 38.74, 9.03))

# Keeping only the villages included in the paper
villages <- panel %>%
  filter(qa_type == 1, dist_addis_km >= 25, pop2011 <= 10000)

cat("Final sample:", nrow(villages), "villages\n")

## Final sample: 315 villages

# Unifying projections
admin_proj <- st_transform(admin, 3857)
panel_sf <- st_as_sf(panel, coords = c("lon", "lat"), crs = 4326) %>%
  st_transform(3857)

vill_sf <- villages %>%
  st_as_sf(coords = c("lon", "lat"), crs = 4326) %>%
  st_transform(3857)

roads_proj <- st_transform(roads, 3857)
grid_proj <- st_transform(grid, 3857)
st_crs(plants) <- 4326
plants_proj <- st_transform(plants, 3857)

# Computing population density by region
points_in_regions <- st_join(panel_sf, admin_proj)

admin_density <- points_in_regions %>%
  st_drop_geometry() %>%
  group_by(REGIONNAME) %>%
  summarise(pop_total = sum(pop2011, na.rm = TRUE)) %>%
  left_join(admin_proj, "REGIONNAME") %>%
  st_as_sf() %>%
  mutate(
    area_km2 = as.numeric(st_area(.)) / 1e6,
    densitat = pop_total / area_km2
  )

# Preparing the roads (filtering them to only keep major roads as in the paper)
major_road_classes <- c("motorway", "trunk", "primary")
roads_filtered <- roads_proj %>%
  filter(highway %in% major_road_classes)

```

1.6 Building the map

```

grid_leg <- grid_proj %>% mutate(layer = "High-Voltage Grid")
plants_leg <- plants_proj %>% mutate(layer = "Power Plants")
vill_leg <- vill_sf %>% mutate(layer = "ERSS Sample Villages")

ggplot() +
  geom_sf(data = admin_density, aes(fill = densitat),
          color = "grey30",
          lwd = 0.15,
          alpha = 0.9) +
  scale_fill_gradient(
    low = "#CCEEFF", high = "#1a5276",
    name = "Population\nDensity",
    trans = "pseudo_log",
    guide = guide_colorbar(
      title.position = "top", title.hjust = 0.5,
      barwidth = 1.2, barheight = 6,
      frame.colour = "grey30", ticks.colour = "grey30",
      order = 1
    ),
    labels = c("Low", "", "High"),
    breaks = function(x) c(min(x, na.rm=T), mean(x, na.rm=T), max(x, na.rm=T))
  ) +
  new_scale_color() +
  geom_sf(data = roads_filtered, aes(color = "Major Roads"), linewidth = 0.7) +
  scale_color_manual(values = c("Major Roads" = "black"), name = NULL) +
  new_scale_color() +
  geom_sf(data = grid_proj, aes(color = "High-Voltage Grid"), linewidth = 0.4) +
  scale_color_manual(values = c("High-Voltage Grid" = "#e41a1c"), name = NULL) +
  new_scale_color() +
  geom_sf(data = vill_sf, aes(color = "ERSS Sample Villages"), size = 0.7) +
  scale_color_manual(values = c("ERSS Sample Villages" = "white"), name = NULL) +
  new_scale_fill() +
  geom_sf(data = vill_sf,
         aes(fill = "ERSS Sample Villages"),
         shape = 21,
         color = "black",
         size = 1,
         stroke = 0.2) +
  scale_fill_manual(values = c("ERSS Sample Villages" = "white"), name = NULL) +
  new_scale_color() +
  geom_sf(data = plants_proj, aes(color = "Power Plants"), size = 1, shape = 15) +
  scale_color_manual(values = c("Power Plants" = "orange"), name = NULL) +
  labs(
    title = "Population Density and Infrastructure in Ethiopia",
    subtitle = "ERSS Sample Villages, Major Roads, and Grid"
  ) +
  theme_void()

```

```

theme(
  legend.position = "right",
  plot.title = element_text(face = "bold", hjust = 0.5, size = 14),
  plot.subtitle = element_text(hjust = 0.5, size = 11),
  plot.background = element_rect(fill = "white", color = NA),
  legend.title = element_text(size = 10, face = "bold")
)

```

Population Density and Infrastructure in Ethiopia

ERSS Sample Villages, Major Roads, and Grid

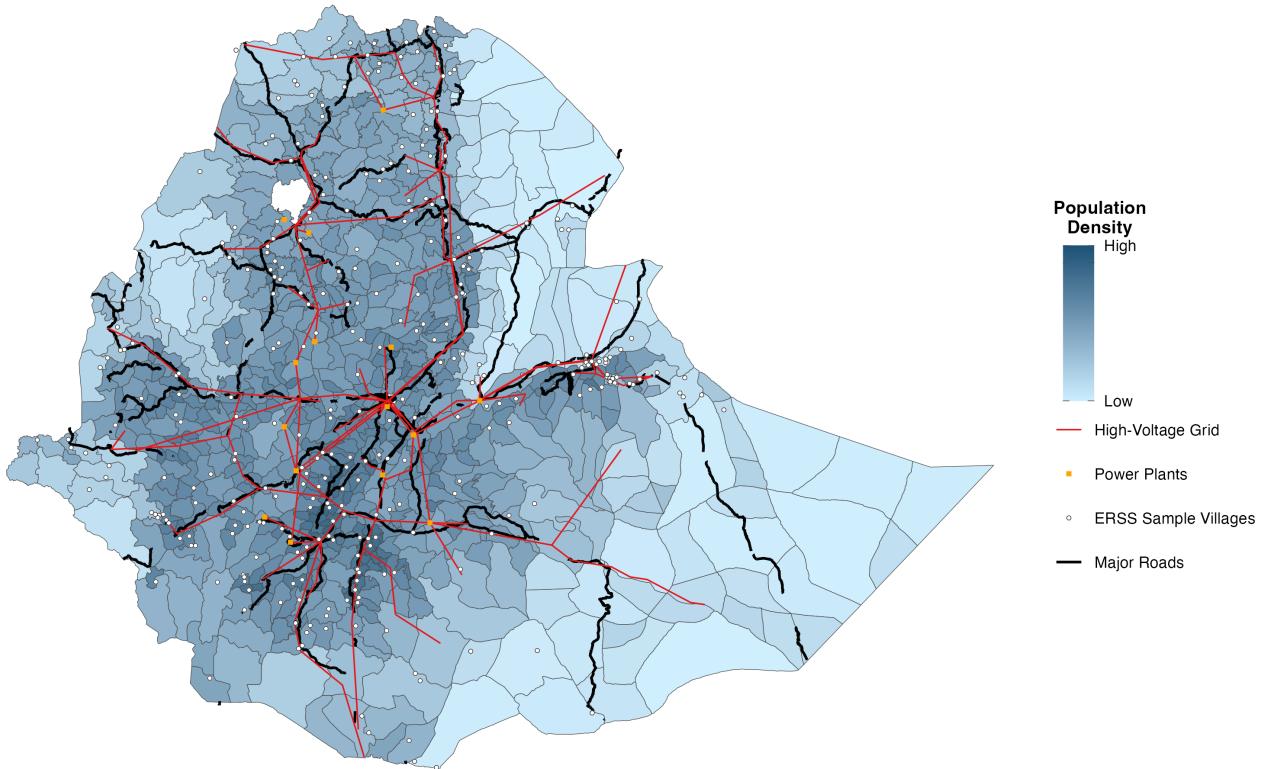


Figure 1: Our replication of Figure 4 from Fried & Lagakos (2021).

2 Part 2: Pellegrina & Sotelo (2025) — Figure 2

2.1 What the map shows

Figure 2 consists of three choropleth maps showing Brazil's population distribution by mesoregion in 1950, 1980, and 2010. Darker greens indicate higher population shares. A black line marks Brazil's border, and a red outline defines the “West” region—the agricultural frontier that received migrants during the so-called “March to the West.” The maps illustrate how population shifted from the Atlantic coast towards the interior over six decades.

2.2 Data sources

All data come from the authors' replication files, available through the journal's supplementary materials:

- Mesoregions shapefile (137 administrative units)
- West boundary shapefile
- Brazil outline shapefile
- Population shares for each year (CSV)

2.3 Why our map differs

Our replication matches the original quite closely since we used the exact same data the authors provided. The only minor differences are styling choices—we used a grey palette instead of blue, and our legend formatting is slightly different. The geographic patterns and data are identical.

2.4 Setup

```
if (!require("pacman")) install.packages("pacman")

pacman::p_load(
  dplyr,
  ggplot2,
  rmapshaper,
  sf
)
```

2.5 Loading the Data

```
brazil <- st_read(
  "./data/Pellegrina_Sotelo_2025/raw/maps/Artisanal",
  layer = "mesoregions",
  quiet = TRUE
)

west <- st_read(
  "./data/Pellegrina_Sotelo_2025/raw/maps/Artisanal",
  layer = "westregions",
  quiet = TRUE
)

brazil_all <- st_read(
  "./data/Pellegrina_Sotelo_2025/raw/maps/Artisanal",
  layer = "brazil",
```

```

    quiet = TRUE
)

brazil <- ms_simplify(brazil, keep = 0.05, keep_shapes = TRUE)
west <- ms_simplify(west, keep = 0.05, keep_shapes = TRUE)
brazil_all <- ms_simplify(brazil_all, keep = 0.05, keep_shapes = TRUE)

table_path <- "./data/Pellegrina_Sotelo_2025/output/fact_maps.csv"

data_fact <- read_csv(
  table_path,
  col_names = c("meso_code", "pop1950", "pop1980", "pop2010"),
  na = "NaN",
  show_col_types = FALSE
) %>%
  mutate(meso_code = as.character(meso_code))

```

2.6 Processing the data

```

# Adding new columns by joining brazil with data_fact by the code of the meso-region.
map_df <- brazil %>%
  left_join(data_fact, by = c("mesocodefo" = "meso_code"))

outline <- st_union(brazil_all)
west_outline <- st_union(west)

years <- c("1950", "1980", "2010")
breaks <- c(0, 0.2, 0.4, 0.6, 1, 12)

```

2.7 Plotting the data

```

for (year in years) {

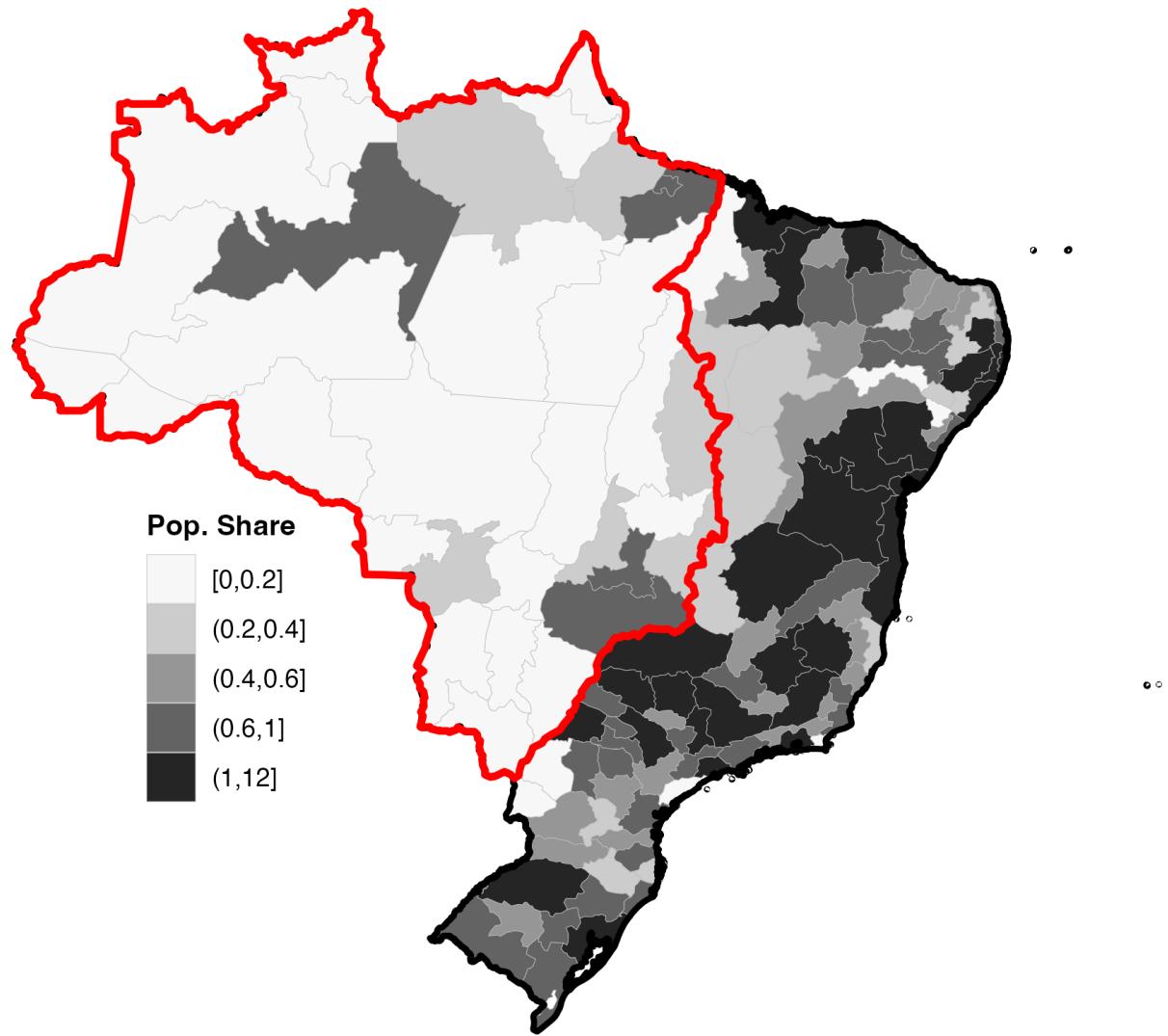
  map_df <- map_df |>
    mutate(
      value = get(paste0("pop", year)),
      quintiles = cut(value, breaks = breaks, include.lowest = TRUE)
    )

  plot_map <- ggplot(data = map_df) +
    geom_sf(aes(fill = quintiles), color = "grey", linewidth = 0.1) +
    geom_sf(data = outline, color = "black", fill = NA, linewidth = 1.00) +
    geom_sf(data = west_outline, color = "red", fill = NA, linewidth = 1.25) +
    scale_fill_brewer(
      palette = "Greys",
      name = "Pop. Share",
      na.value = "grey90"
    ) +
    labs(
      title = paste("Figure 2: Population Distribution in Brazil,", year),
      caption = "Source: Pellegrina & Sotelo (2025) replication data"
    )
}

```

```
theme_void(base_size = 11) +
theme(
  plot.title = element_text(
    size = 14, face = "bold", hjust = 0.5, margin = margin(b = 10)
  ),
  plot.caption = element_text(
    size = 8, color = "grey40", hjust = 0, margin = margin(t = 10)
  ),
  legend.position = c(0.15, 0.25),
  legend.justification = c(0, 0),
  legend.background = element_rect(fill = "white", color = NA),
  legend.title = element_text(size = 10, face = "bold"),
  legend.text = element_text(size = 9)
)
print(plot_map)
}
```

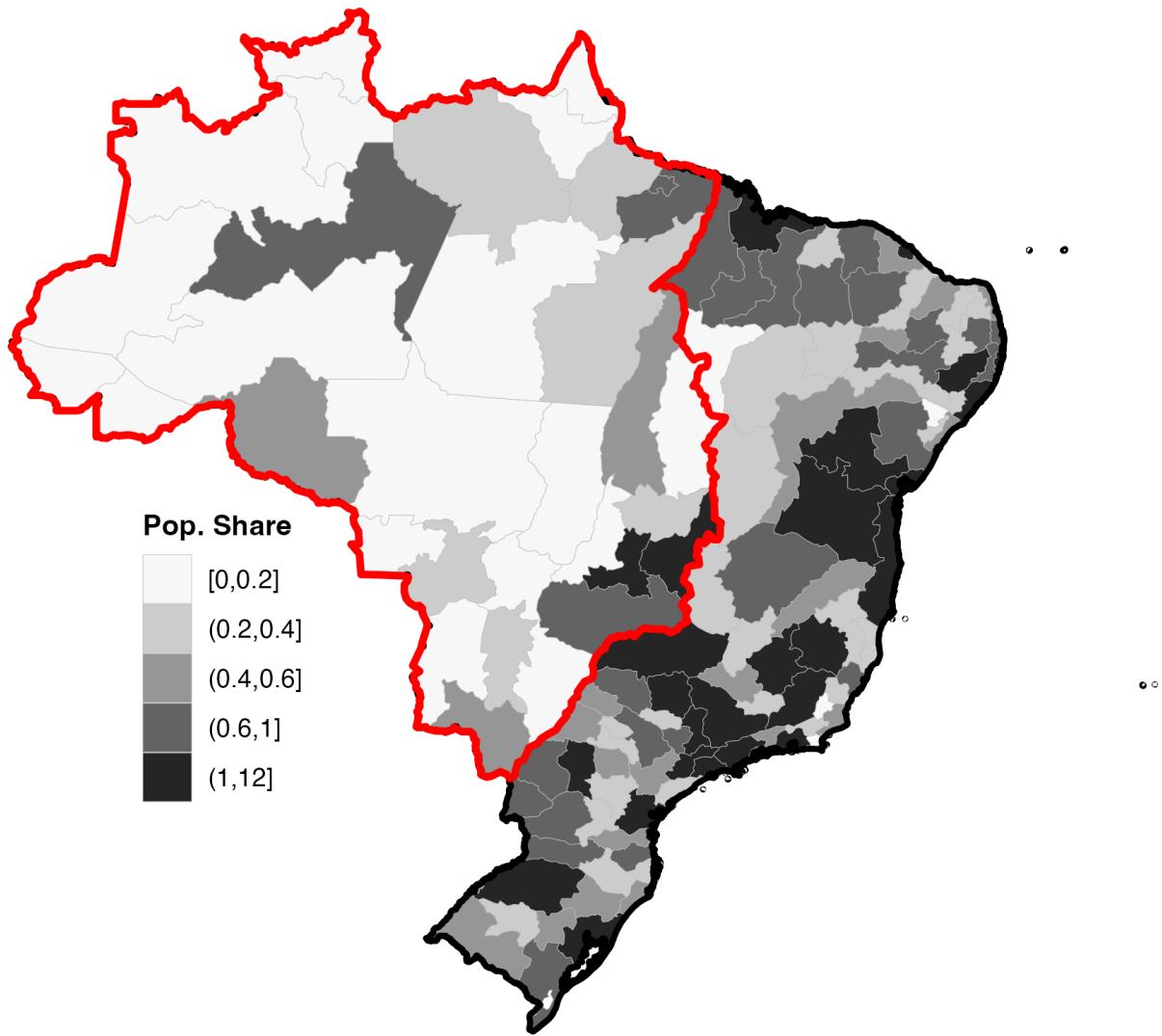
Figure 2: Population Distribution in Brazil, 1950



Source: Pellegrina & Sotelo (2025) replication data

Figure 2: Population share maps for Brazil's mesoregions in 1950, 1980, and 2010. The red outline indicates the 'West' region.

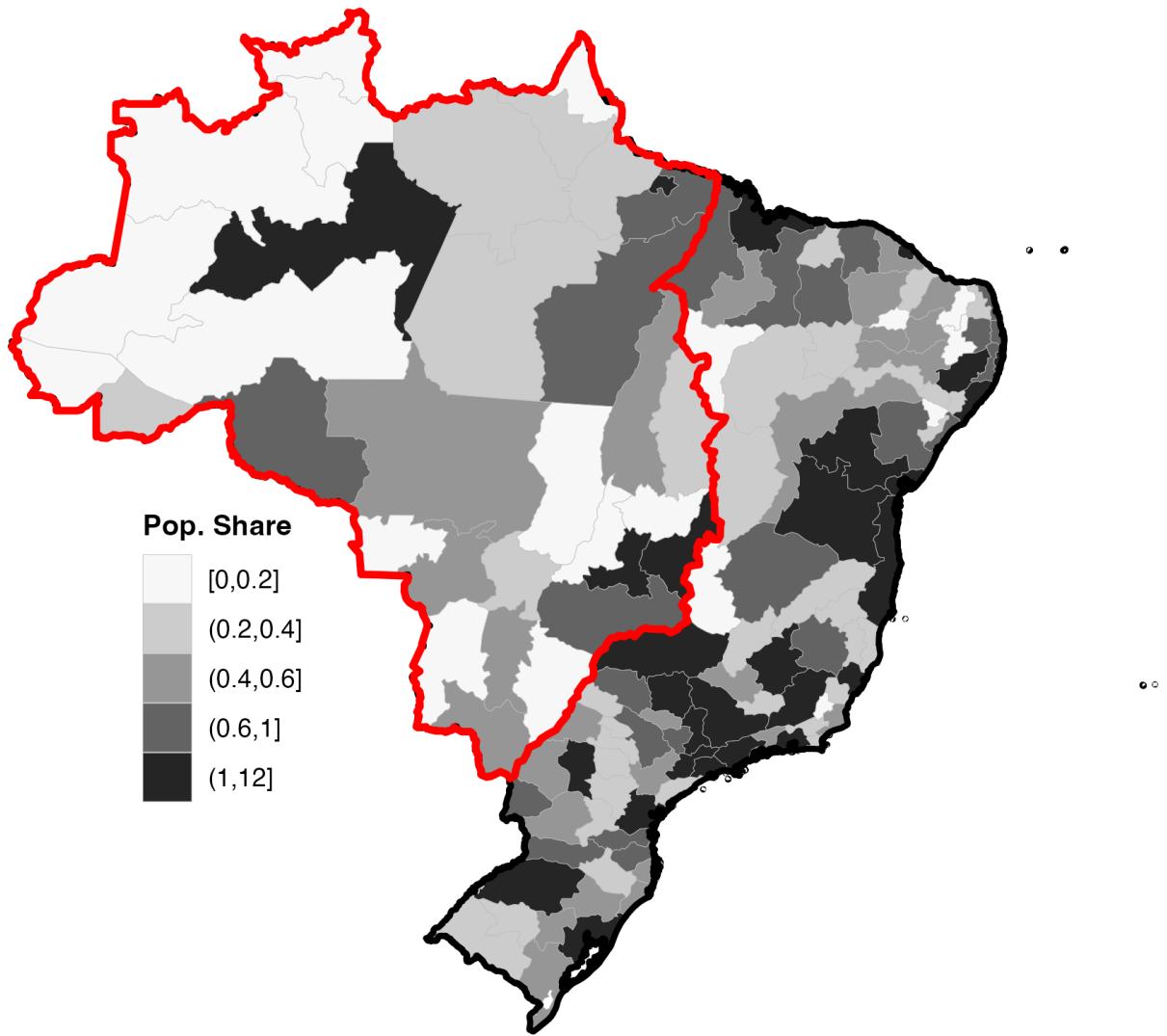
Figure 2: Population Distribution in Brazil, 1980



Source: Pellegrina & Sotelo (2025) replication data

Figure 3: Population share maps for Brazil's mesoregions in 1950, 1980, and 2010. The red outline indicates the 'West' region.

Figure 2: Population Distribution in Brazil, 2010



Source: Pellegrina & Sotelo (2025) replication data

Figure 4: Population share maps for Brazil's mesoregions in 1950, 1980, and 2010. The red outline indicates the 'West' region.

3 Part 3: Morten & Oliveira (2024) — Figure 1

3.1 What the map shows

Figure 1 displays Brazil's federal highway network as of 2000, distinguishing between radial highways (those emanating from Brasília, shown as solid lines), non-radial highways (dashed lines), and the minimum spanning tree instrument (dotted lines). The MST represents the hypothetical road network that would connect Brasília to state capitals using least-cost paths—this serves as the authors' instrument for actual road placement. Capital cities appear as points, with Brasília highlighted as the hub.

3.2 Data sources

The authors provide all data through openICPSR:

- Replication package — openICPSR #183316

This includes:

- State boundaries (1940 vintage)
- Highway network (2000)
- MST instrument
- Capital city locations

3.3 Why our map differs

We used the exact same shapefiles from the replication package, so the geographic content is identical. We only made a few aesthetic changes: we used colors (amber for MST, steel-blue for non-radial, dark charcoal for radial) instead of all-grey tones.

3.4 Setup

```
pacman::p_load(
  dplyr,
  ggplot2,
  ggrepel,
  rmapshaper,
  sf
)

gis_path <- "./data/Morten_Oliveira_2024/GIS_data"
```

3.5 Loading the Data

```
states <- st_read(
  file.path(gis_path, "uf1940/uf1940_prj.shp"),
  quiet = TRUE
)

states_simple <- ms_simplify(states, keep = 0.01, keep_shapes = TRUE)

all_highways <- st_read(
  file.path(gis_path, "roads/2000/highways_2000_prj.shp"),
  quiet = TRUE
)
```

```

all_highways_simple <- ms_simplify(all_highways, keep = 0.01, keep_shapes = TRUE)

mst_pie <- st_read(
  file.path(gis_path, "mst/mst_pie_prj.shp"),
  quiet = TRUE
)

mst_pie_simple <- ms_simplify(mst_pie, keep = 0.01, keep_shapes = TRUE)

capital_cities <- st_read(
  file.path(gis_path, "cities/brazil_capital_cities_prj.shp"),
  quiet = TRUE
)

```

3.6 Filtering the Data

```

# Find the radial highways
radial_highways <- all_highways_simple %>%
  filter(dm_anlys_p == 1 & dm_radial == 1)

# Find the nonradial highways
nonradial_highways <- all_highways_simple %>%
  filter(dm_anlys_p == 1 & dm_radial == 0)

# Keep only capital cities
cities_xy <- capital_cities %>%
  cbind(st_coordinates(capital_cities)) %>%
  rename(lng = X, lat = Y)

```

3.7 Building the Map

```

colors_mo <- c(
  "Minimum spanning tree" = "#D4A03E",
  "Non-radial highways (2000)" = "#7B8D9E",
  "Radial highways (2000)" = "#2C2C2C"
)

fig_1 <- ggplot() +
  geom_sf(
    data = states_simple,
    fill = "#FDFBF7",
    color = "#DODODO",
    linewidth = 0.3
  ) +
  geom_sf(
    data = mst_pie_simple,
    aes(color = "Minimum spanning tree"),
    linewidth = 0.9,
    linetype = "dotted",
    show.legend = "line"
  ) +
  geom_sf(

```

```

data = nonradial_highways,
aes(color = "Non-radial highways (2000)" ),
lineWidth = 0.6,
linetype = "dashed",
show.legend = "line"
) +
geom_sf(
  data = radial_highways,
  aes(color = "Radial highways (2000)" ),
  lineWidth = 1.0,
  linetype = "solid",
  show.legend = "line"
) +
geom_point(
  data = cities_xy,
  aes(x = lng, y = lat),
  size = 2.0,
  color = "#2C2C2C"
) +
geom_text_repel(
  data = cities_xy,
  aes(x = lng, y = lat, label = CITY_NAME),
  size = 2.7,
  color = "#3A3A3A",
  segment.color = "#BBBBBB",
  segment.size = 0.25,
  box.padding = 0.35,
  point.padding = 0.25,
  max.overlaps = 30,
  seed = 42
) +
scale_color_manual(
  name = NULL,
  values = colors_mo,
  guide = guide_legend(
    override.aes = list(
      linetype = c("dotted", "dashed", "solid"),
      lineWidth = c(1.3, 1.0, 1.3)
    )
  )
) +
coord_sf(
  crs = st_crs(states_simple),
  datum = NA
) +
theme_minimal(base_size = 11) +
theme(
  axis.title = element_blank(),
  axis.text = element_blank(),
  axis.ticks = element_blank(),
  panel.grid = element_blank(),
  panel.background = element_rect(fill = "#EDF3F7", color = NA),
  legend.position = c(0.17, 0.20),

```

```
legend.justification = c(0, 0),
legend.background = element_rect(
  fill = alpha("white", 0.92),
  color = "#CCCCCC",
  linewidth = 0.3
),
legend.key.width = unit(1.4, "cm"),
legend.text = element_text(size = 9),
legend.margin = margin(6, 10, 6, 8),
plot.margin = margin(5, 5, 5, 5)
)

print(fig_1)
```



Figure 5: Replication of Figure 1 from Morten & Oliveira (2024). Data source: openICPSR #183316.

4 Part 4: Mettetal (2019) — Figure 2

4.1 What the map shows

Figure 2 displays average river gradient by magisterial district in South Africa, with darker shades indicating steeper rivers. Black dots mark the location of irrigation dams. The map illustrates the paper's identification strategy: dams tend to cluster in areas with gentler river slopes (lighter shades), particularly on the eastern side of the country where most irrigated agriculture is concentrated.

4.2 Data sources

We assembled layers from several public sources:

- Digital elevation model — [geodata R package](#) (90m SRTM)
- District boundaries — GADM via geodata package
- Dam locations — South Africa Dam Safety Office (KML file)
- River network — SA Department of Water Affairs
- Dam registry — SA Department of Water Affairs (Excel file with dam purposes)

4.3 Why our map differs

The main difference is that, to reproduce the way the author computes river gradients, we had to lower the resolution of the DEM and create a 1.5 km buffer around the rivers. Additionally, our sextile classification may differ slightly from the original due to these adjustments. However, the overall pattern—steeper rivers in the west, gentler rivers and more dams in the east—matches the original figure well.

4.4 Setup

```
pacman::p_load(  
  sf,  
  terra,  
  tidyverse,  
  geodata,  
  exactextractr,  
  readxl  
)  
  
dir_data <- "./data/Mettetal_2019/"  
dir.create(dir_data, showWarnings = FALSE, recursive = TRUE)
```

4.5 Loading the Data

```
# --- DEM of South Africa --- #  
  
# We download South Africa's border  
sa_boundary <- geodata::gadm(country = "ZAF", level = 0, path = dir_data)  
  
# Then we download 90m (1 pixel 90mx90m) global elevation, and we clip the data to  
# South Africa's borders to remove unnecessary areas.  
dem_raw <- geodata::elevation_global(res = 0.5, country = "ZAF", path = dir_data)  
dem <- terra::crop(dem_raw, sa_boundary, mask = TRUE)
```

```

rm(dem_raw)
gc()

##           used   (Mb) gc trigger   (Mb) limit   (Mb)  max used   (Mb)
## Ncells  8174135 436.6   15558942  831.0        NA 11117814  593.8
## Vcells 58454654 446.0   147136180 1122.6      16384 144155163 1099.9
# --- Load the other data --- #

sf_districts <- st_read(
  file.path(dir_data, "Magisterialdistricts2005"),
  "District municipalities 2005"
) %>%
  st_make_valid()

## Reading layer `District municipalities 2005` from data source
##   `/Users/gal.lagelpibuxade/Desktop/Replication_Assignment1/data/Mettetal_2019/Magisterialdistricts2005'
##   using driver `ESRI Shapefile'
## Simple feature collection with 371 features and 18 fields
## Geometry type: MULTIPOLYGON
## Dimension:      XY
## Bounding box:  xmin: 16.45485 ymin: -34.83305 xmax: 32.89128 ymax: -22.12595
## Geodetic CRS:  GCS_Assumed_Geographic_1

sf_dams <- st_read(file.path(dir_data, "doc.kml")) %>%
  st_make_valid() %>%
  st_set_crs(4326)

## Reading layer `GEarth dams` from data source
##   `/Users/gal.lagelpibuxade/Desktop/Replication_Assignment1/data/Mettetal_2019/doc.kml'
##   using driver `KML'
## Simple feature collection with 5744 features and 2 fields
## Geometry type: POINT
## Dimension:      XYZ
## Bounding box:  xmin: 17.5755 ymin: -34.67222 xmax: 33.65056 ymax: -18.81806
## z_range:       zmin: 0 zmax: 0
## Geodetic CRS:  WGS 84

sf_rivers <- st_read(file.path(dir_data, "All"), "wriall500") %>%
  st_make_valid()

## Reading layer `wriall500` from data source
##   `/Users/gal.lagelpibuxade/Desktop/Replication_Assignment1/data/Mettetal_2019/All'
##   using driver `ESRI Shapefile'
## Simple feature collection with 10352 features and 26 fields
## Geometry type: LINESTRING
## Dimension:      XY
## Bounding box:  xmin: 15.98333 ymin: -34.76999 xmax: 33.99925 ymax: -19.79143
## Geodetic CRS:  Cape

# --- Excel with dams information --- #

dams_info <- read_excel(file.path(dir_data, "List of Registered Dams Jul2025.xlsx"))

# Clean the file because originally it has some duplicated rows, and filter it as in the
# paper (only consider dam with the purpose of irrigation)
dams_irrigation <- dams_info %>%

```

```

distinct() %>%
filter(str_detect(Purpose, regex("IRRIGATION", ignore_case = TRUE))) %>%
distinct(`No of dam`, .keep_all = TRUE)

# Finally we keep only the dams that satisfy the conditions that we want
sf_dams_filtered <- sf_dams %>%
  inner_join(dams_irrigation, by = c("Name" = "Name of dam"))

```

4.6 Align the coordinates

```

if (is.na(st_crs(sf_districts))) {
  sf_districts <- st_set_crs(sf_districts, 4326)
}

if (is.na(st_crs(sf_rivers))) {
  sf_rivers <- st_set_crs(sf_rivers, 4326)
}

if (is.na(st_crs(sf_dams_filtered))) {
  sf_dams_filtered <- st_set_crs(sf_dams_filtered, 4326)
}

rivers_vect <- project(vect(sf_rivers), dem)
districts_vect <- project(vect(sf_districts), dem)
dams_vect <- project(vect(sf_dams_filtered), dem)

```

4.7 Carving elevation along rivers

```

# First we create a buffer of 1,5km, so that we ensure the river paths are wide enough to
# overlap with the elevation pixels.
rivers_buf <- buffer(rivers_vect, width = 1500)

# Then, we create a digital stencil of the rivers and use it to erase all land data except
# the river channels.
river_mask <- rasterize(rivers_buf, dem, field = 1)
dem_rivers_only <- mask(dem, river_mask)

rm(rivers_buf, river_mask, rivers_vect)
gc()

##           used   (Mb) gc trigger   (Mb) limit   (Mb)  max used   (Mb)
## Ncells  8402231 448.8  15558942  831.0        NA 14038179  749.8
## Vcells 63345357 483.3 147136180 1122.6      16384 144155163 1099.9

```

4.8 Compute the slope and aggregate it by district

```

# Compute the slope in degrees
slope_deg <- terrain(dem_rivers_only, "slope", unit = "degrees")

# Converts the degrees into a percentage gradient as in the paper.
slope_pct <- tan(slope_deg * pi / 180) * 100

```

4.10 Building the Map

```
p <- ggplot() +
  geom_sf(
    data = sf_districts_clean,
    aes(fill = slope_percentile),
    color = "white",
    linewidth = 0.1
  ) +
  geom_sf(
    data = st_as_sf(dams_vect),
    aes(color = "Dam Location"),
    size = 0.2
  ) +
  scale_fill_grey(
    name = "Average District River Gradient",
    start = 0.9,
```

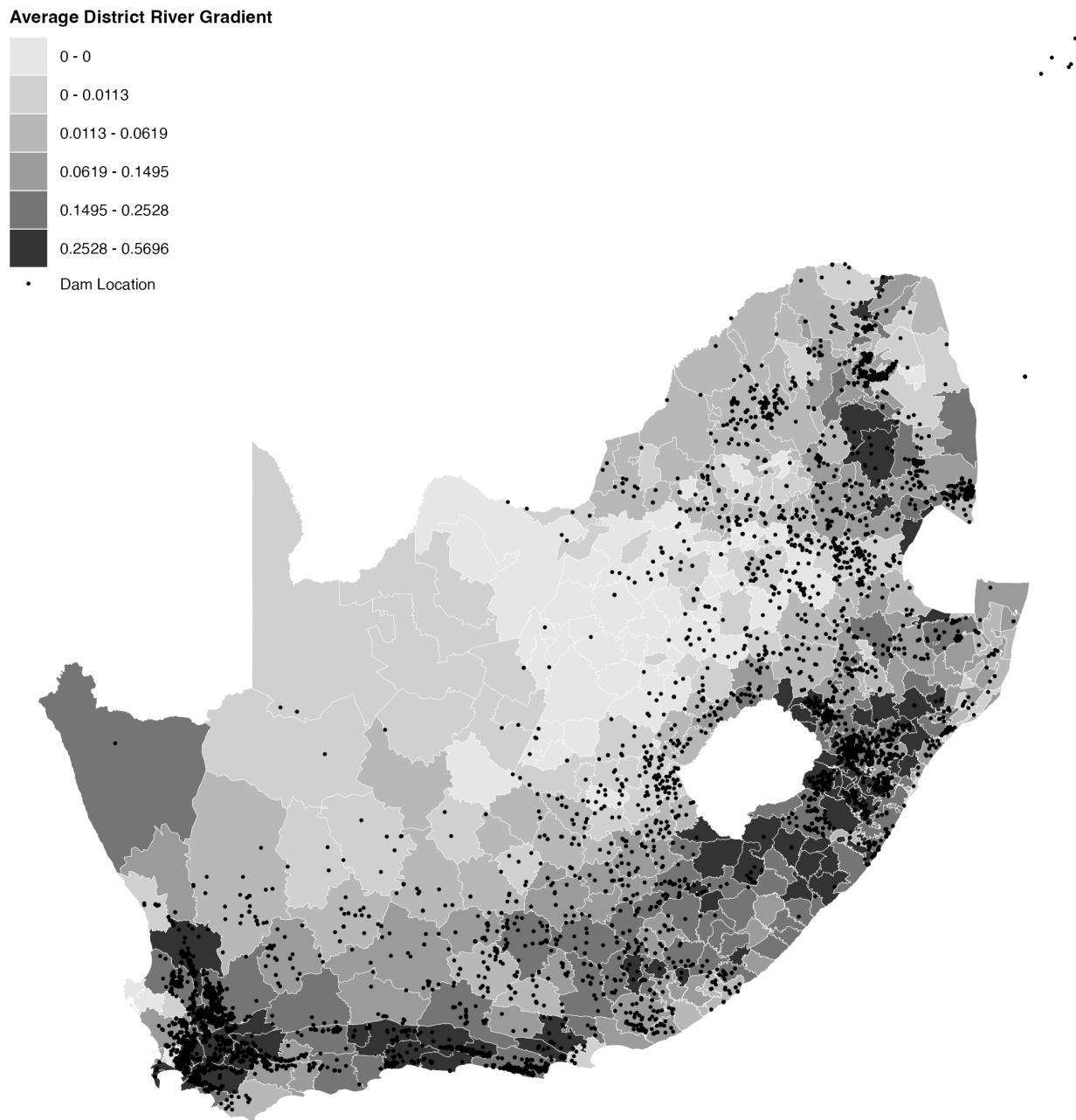
```

    end = 0.2
) +
scale_color_manual(
  name = NULL,
  values = c("Dam Location" = "black")
) +
labs(
  title = "Figure 2: River Gradients and Dam Locations - South Africa",
  caption = "Source: SA Department of Water Affairs, GADM, SRTM elevation data"
) +
theme_void() +
theme(
  plot.title = element_text(
    size = 14, face = "bold", hjust = 0.5, margin = margin(b = 10)
  ),
  plot.caption = element_text(
    size = 8, color = "grey40", hjust = 0, margin = margin(t = 10)
  ),
  legend.position = c(0.02, 0.98),
  legend.justification = c(0, 1),
  legend.title = element_text(size = 8, face = "bold"),
  legend.text = element_text(size = 7),
  legend.spacing.y = unit(-0.05, "cm")
) +
guides(
  fill = guide_legend(order = 1),
  color = guide_legend(order = 2, override.aes = list(linetype = 0))
)

print(p)

```

Figure 2: River Gradients and Dam Locations - South Africa



Source: SA Department of Water Affairs, GADM, SRTM elevation data

Figure 6: Replication of Figure 2 from Mettetal (2019). Darker shades indicate steeper rivers; black dots show irrigation dams.

5 Part 5: Balboni (2021) — Figure 3

5.1 What the map shows

Figure 3 displays Vietnam’s road network in 2000 and 2010, illustrating the spatial distribution of road upgrades during a period of major transport infrastructure investment. Roads are categorized into five types: freeways, dual carriageways, major roads, minor roads, and other roads. The maps reveal how investments were concentrated in the low elevation coastal zone, particularly in the Red River Delta and Mekong River Delta regions. The paper uses this infrastructure data to study whether road investments should continue to favor coastal areas given rising sea levels and climate vulnerability.

5.2 Data sources

Since the original georeferenced road data from the paper is not publicly available, we use OpenStreetMap extracts as proxies:

- [Vietnam Roads 2015](#) — Humanitarian Data Exchange (OSM extract)
- [Vietnam Roads 2026](#) — Geofabrik (current OSM extract)
- Country boundary — GADM via `{geodata}` R package

5.3 Why our map differs

We could not replicate the original maps exactly because the author’s manually digitized road network data from ITMB travel maps is not publicly available. Instead, we use OpenStreetMap data from 2015 and Geofabrik for 2026 data as modern approximations. The road classification also differs slightly: we infer road hierarchy from OSM’s `highway` tags rather than the author’s six-category system based on physical road characteristics. Despite these differences, the maps capture the same essential pattern—a dense network concentrated along the coast and major corridors, with the highest-quality roads connecting major urban centers.

5.4 Setup

```
pacman::p_load(
  sf,
  dplyr,
  ggplot2,
  geodata
)

colors_balboni <- c("blue", "darkgreen", "red", "orange", "gold")

vehicular_classes <- c(
  "motorway", "motorway_link",
  "trunk", "trunk_link",
  "primary", "primary_link",
  "secondary", "secondary_link",
  "tertiary", "tertiary_link",
  "road"
)

lineweights <- c(0.8, 0.6, 0.4, 0.2, 0.1)
legend_widths <- c(1.2, 0.9, 0.6, 0.3, 0.15)
```

```
vn_border <- gadm(country = "VNM", level = 0, path = tempdir()) |>
  st_as_sf()
```

5.5 Map of 2015

```
r15_raw <- st_read(
  "./data/Balboni_2021/vnm_rdsl_2015_osm/",
  "vnm_rdsl_2015_OSM",
  quiet = TRUE
)

# Filter only for vehicular roads, excluding footways, steps, and paths
r15 <- r15_raw %>%
  filter(type %in% vehicular_classes) %>%
  mutate(
    road_type = case_when(
      type %in% c("motorway", "motorway_link") ~ "Freeway",
      type %in% c("trunk", "trunk_link") ~ "Trunk / Dual Carriageway",
      type %in% c("primary", "primary_link") ~ "Primary Road",
      type %in% c("secondary", "secondary_link") ~ "Secondary Road",
      TRUE ~ "Tertiary / Local Road"
    ),
    road_type = factor(
      road_type,
      levels = c(
        "Freeway",
        "Trunk / Dual Carriageway",
        "Primary Road",
        "Secondary Road",
        "Tertiary / Local Road"
      )
    ),
    road_type = droplevels(road_type)
  )
```

5.6 Map of 2026

```
r26_raw <- st_read(
  "./data/Balboni_2021/vietnam-260126-free/",
  "gis_osm_roads_free_1",
  quiet = TRUE
)

# Filter only for the roads that we want to keep
r26 <- r26_raw %>%
  filter(fclass %in% vehicular_classes) %>%
  mutate(
    road_type = case_when(
      fclass %in% c("motorway", "motorway_link") ~ "Freeway",
      fclass %in% c("trunk", "primary") & oneway %in% c("T", "B") ~ "Dual carriageway",
      fclass %in% c("trunk", "trunk_link", "primary", "primary_link") ~ "Major roads",
      fclass %in% c("secondary", "secondary_link") ~ "Minor roads",
      TRUE ~ "Local roads"
    )
  )
```

```

    TRUE ~ "Other roads"
),
road_type = factor(
  road_type,
  levels = c(
    "Freeway",
    "Dual carriageway",
    "Major roads",
    "Minor roads",
    "Other roads"
  )
),
road_type = droplevels(road_type)
)

```

5.7 Building the Maps

```

n_levels_15 <- length(levels(r15$road_type))

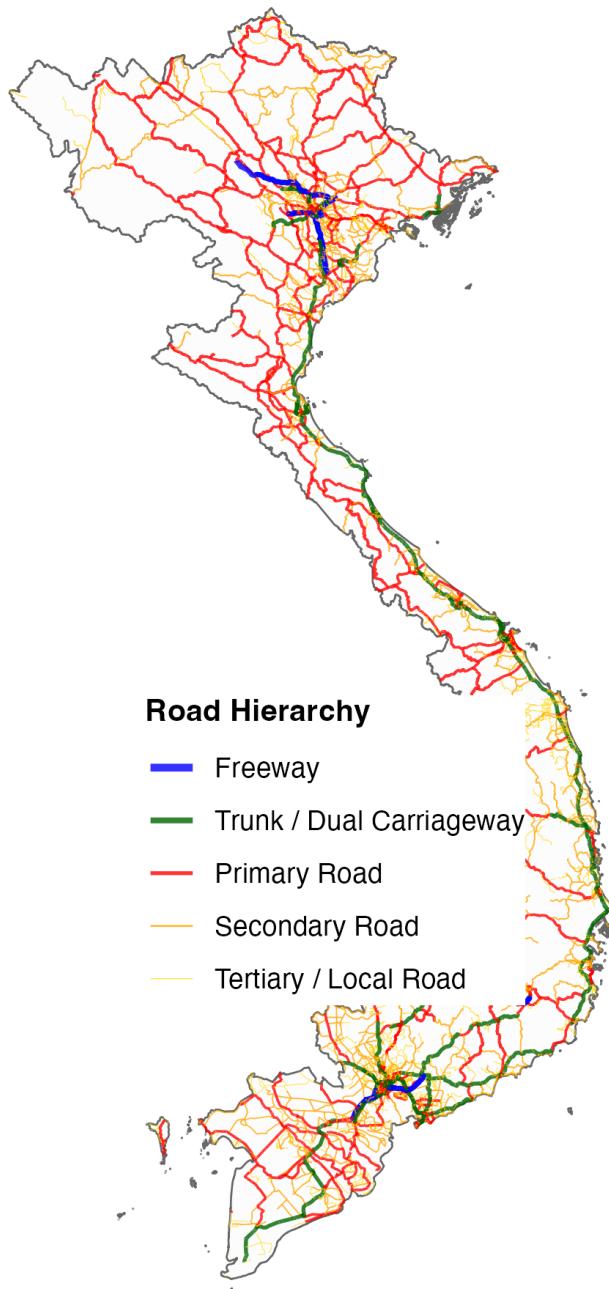
p_2015 <- ggplot() +
  geom_sf(
    data = vn_border,
    fill = "#fcfcfc",
    color = "#666666",
    linewidth = 0.3
  ) +
  geom_sf(
    data = r15,
    aes(color = road_type, linewidth = road_type),
    alpha = 0.8
  ) +
  scale_color_manual(
    values = colors_balboni[1:n_levels_15],
    name = "Road Hierarchy"
  ) +
  scale_linewidth_manual(
    values = linewidths[1:n_levels_15],
    name = "Road Hierarchy"
  ) +
  guides(
    color = guide_legend(
      override.aes = list(linewidth = legend_widths[1:n_levels_15])
    )
  ) +
  labs(
    title = "Figure 3a: Vietnam Road Network (2015)",
    caption = "Data: OpenStreetMap via Humanitarian Data Exchange"
  ) +
  theme_void(base_size = 11) +
  theme(
    plot.title = element_text(
      size = 14, face = "bold", hjust = 0.5, margin = margin(b = 10)
    ),

```

```
plot.caption = element_text(
  size = 8, color = "grey40", hjust = 0, margin = margin(t = 10)
),
legend.position = c(0.25, 0.25),
legend.justification = c(0, 0),
legend.background = element_rect(fill = "white", color = NA),
legend.title = element_text(size = 10, face = "bold"),
legend.text = element_text(size = 9)
)

print(p_2015)
```

Figure 3a: Vietnam Road Network (2015)



Data: OpenStreetMap via Humanitarian Data Exchange

Figure 7: Vietnam road network (2015 OSM data).

```
n_levels_26 <- length(levels(r26$road_type))
```

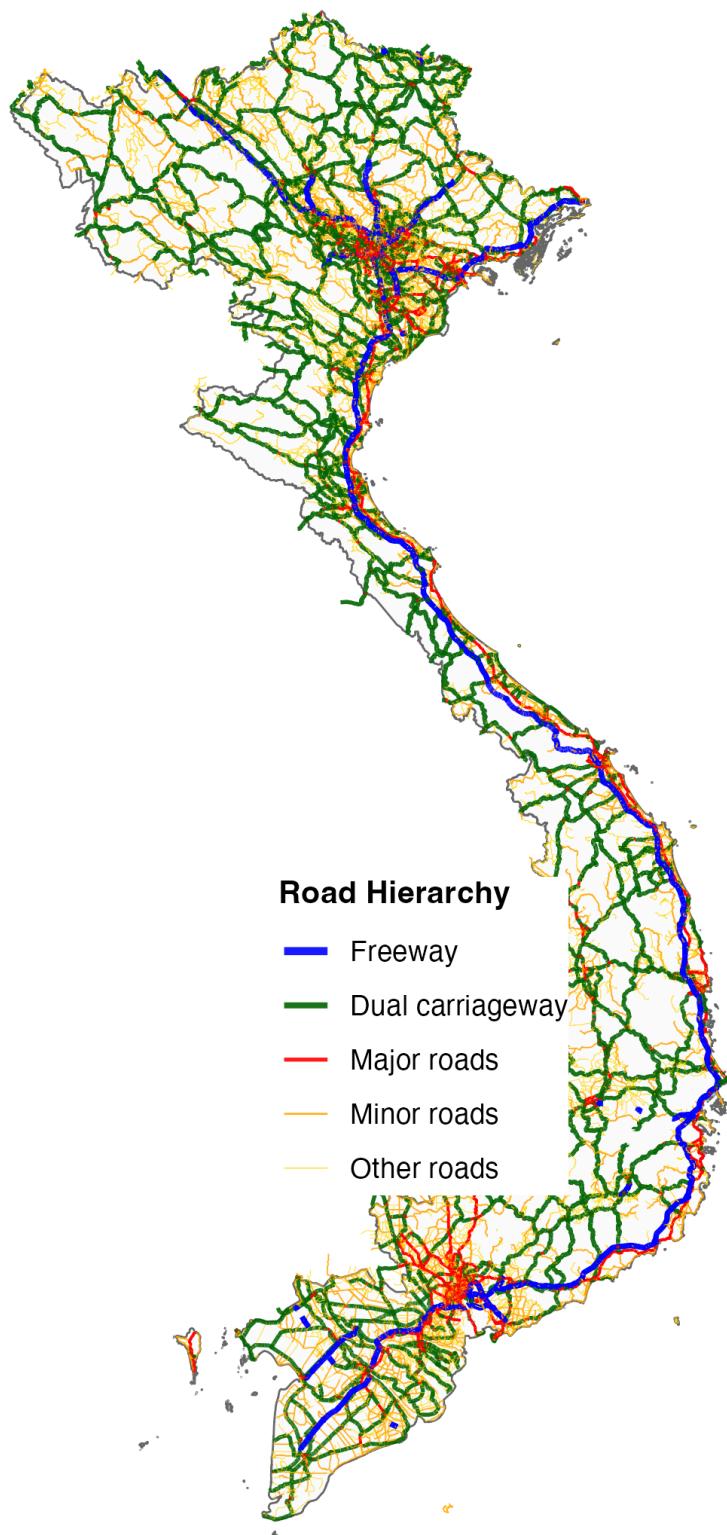
```

p_2026 <- ggplot() +
  geom_sf(
    data = vn_border,
    fill = "#fafafa",
    color = "#666666",
    linewidth = 0.3
  ) +
  geom_sf(
    data = r26,
    aes(color = road_type, linewidth = road_type),
    alpha = 0.9
  ) +
  scale_color_manual(
    values = colors_balboni[1:n_levels_26],
    name = "Road Hierarchy"
  ) +
  scale_linewidth_manual(
    values = linewidths[1:n_levels_26],
    name = "Road Hierarchy"
  ) +
  guides(
    color = guide_legend(
      override.aes = list(linewidth = legend_widths[1:n_levels_26])
    )
  ) +
  labs(
    title = "Figure 3b: Vietnam Road Network (2026)",
    caption = "Data: OpenStreetMap via Geofabrik"
  ) +
  theme_void(base_size = 11) +
  theme(
    plot.title = element_text(
      size = 14, face = "bold", hjust = 0.5, margin = margin(b = 10)
    ),
    plot.caption = element_text(
      size = 8, color = "grey40", hjust = 0, margin = margin(t = 10)
    ),
    legend.position = c(0.25, 0.25),
    legend.justification = c(0, 0),
    legend.background = element_rect(fill = "white", color = NA),
    legend.title = element_text(size = 10, face = "bold"),
    legend.text = element_text(size = 9)
  )
}

print(p_2026)

```

Figure 3b: Vietnam Road Network (2026)



6 References

- Balboni, C. (2025). In Harm's Way? Infrastructure Investments and the Persistence of Coastal Cities. *American Economic Review*.
- Fried, S., & Lagakos, D. (2021). Rural Electrification, Migration and Structural Transformation: Evidence from Ethiopia. *Regional Science and Urban Economics*.
- Mettetal, E. (2019). Irrigation dams, water and infant mortality: Evidence from South Africa. *Journal of Development Economics*.
- Morten, M., & Oliveira, J. (2024). The Effects of Roads on Trade and Migration: Evidence from a Planned Capital City. *American Economic Journal: Applied Economics*.
- Pellegrina, H. S., & Sotelo, S. (2025). Migration, Specialization, and Trade: Evidence from Brazil's March to the West. *Journal of Political Economy*.