Source code for *Mapping Elephant Movements in Kruger*

Pratik R. Gupte

2020-12-30

Contents

1 Introduction

2 Data Sources

2.1 Load libraries

```
# load libs
library(data.table)
library(ggplot2)
library(sf)
library(rnaturalearth)
```

2.2 Data pre-processing

```
# load data
data <- fread("data/elephant_data.csv")

# drop NAs
data <- na.omit(data, cols = c("xutm", "yutm"))
```

2.3 Loacations in each 5km

```
# round data
data_count <- copy(data)
data_count[, c("xutm", "yutm") := lapply(list(xutm, yutm),
  round,
```

```
  digits = -3
)]

data_count <- data_count[, .N, by = list(xutm, yutm)]

# save
fwrite(data_count, file = "data/data_ele_count.csv")
```

```
# make sf and lines
data <- split(data, by = "id")

# get histogram of speeds
data <- rbindlist(data)

# split data again
data <- split(data, by = c("id"))

# make multilinestring
geometry <- st_sfc(
  lapply(data, function(x) {
    st_linestring(
      as.matrix(x[, c("long", "lat")])
    )
  }),
  crs = 4326
)

# retransform
geometry <- st_transform(geometry, 32736)
```

2.4 Make sf

```
# get data
data_sf <- mapply(
  function(df) {
    df[1, c("id")]
  },
  data,
  SIMPLIFY = FALSE
)

# add geometry
data_sf <- rbindlist(data_sf)
data_sf[, geometry := geometry]

# make sf
data_sf <- st_sf(data_sf, crs = 32736)

# save
st_write(data_sf,
  dsn = "data/data_lines_elephants.gpkg",
  append = FALSE
)
```

2.5 Get ZA

```
# get natural earth data
land <- ne_countries(
  continent = "africa",
  scale = "small",
  returnclass = "sf"
)

# save
st_write(land, "data/za_boundary.gpkg", append = F)
```

2.6 Load libraries

```
# load libraries
library(sf)
library(ggplot2)
library(ggspatial)
library(pals)
library(rnaturalearth)
library(data.table)
library(scico)
```

2.7 Load data

```
# prepare bounding box
bbox <- c(
  xmin = 342000,
  xmax = 396000,
  ymin = 7260000,
  ymax = 7298050
)
bbox_sf <- st_bbox(bbox)
bbox_sf <- st_as_sfc(bbox_sf)
st_crs(bbox_sf) <- 32736

# get data
data <- st_read("data/data_lines_elephants.gpkg")

# get data 253
data_253 <- data[data$id == "AM253", ]

# get other data
data_rest <- data[data$id %in% c("AM255", "AM99", "AM239"), ]

# get kruger data
kruger <- st_read("data/kruger_clip/kruger_clip.shp")
kruger <- st_transform(kruger, 32736)

# get kruger point
kruger_point <- st_point(c(31.5, -24))
kruger_point <- st_sfc(kruger_point, crs = 4326)
kruger_point <- st_transform(kruger_point, 32736)

# get za
za <- st_read("data/za_boundary.gpkg")
```

```
za <- st_transform(za, 32736)

# get rivers
rivers <- st_read("data/river_crop/")

# river buffer
rivers <- st_transform(rivers[is.na(rivers$seasonal), ], 32736)
# river_buffer <- st_crop(river_buffer, bbox)
# river_buffer <- st_buffer(river_buffer, 200)
# river_buffer <- st_union(river_buffer)

# waterholes
waterholes <- st_read("data/waterholes/")

# get slope
library(raster)

res_init <- res(raster("data/kruger_temperature_UTM.tif"))
res_final <- res_init * 200 / res_init
gdalUtils::gdalwarp(
  srcfile = "data/kruger_temperature_UTM.tif",
  dstfile = "data/kruger_temp_200m.tif",
  tr = c(res_final), r = "average",
  te = c(bbox(raster("data/kruger_temperature_UTM.tif")))
)

# read in cropped raster
# elev <- raster("data/kruger_elevation_200m.tif")
temp <- raster("data/kruger_temp_200m.tif")

temp <- raster::crop(temp, as(bbox_sf, "Spatial"))
temp <- cbind(coordinates(temp), values(temp))
temp <- data.table(temp)
temp <- temp[V3 > 22, ]
```

2.8 Make main plot

```
bbox <- c(
  xmin = 342000,
  xmax = 393000,
  ymin = 7260000,
  ymax = 7298050
)

textbox <- glue::glue(
  "**Kruger Elephants Shuttle to Water**

  African elephants move as they please, ignoring park boundaries \\
  when it suits them. Yet they need water to help them through the thermal landscape. \\
  Kruger elephants frequent water sources during the afternoon, the \\
  hottest part of the day; arriving and leaving at high speed. \\
  Here, elephant _AM253_ (red) and her herd keep to their own side of water sources, \\
  while other herds keep to theirs."
)
# texttitle = "Elephants Shuttle to Water"
textdata <- data.table(
```

```
  x = bbox["xmin"] + 6000,
  y = bbox["ymax"] - 9000,
  label = textbox
)

# make plot
fig_main <-
  ggplot() +
  geom_sf(
    data = kruger,
    col = NA,
    fill = "antiquewhite",
  ) +
  geom_tile(
    data = temp,
    aes(x, y, fill = V3),
    show.legend = F,
    # col = "grey",
    alpha = 0.45
  ) +
  geom_sf(
    data = rivers[is.na(rivers$seasonal), ],
    lwd = 1,
    col = scico::scico(3,
      palette = "nuuk"
    )[1],
    alpha = 0.35
  ) +
  geom_sf(
    data = data_rest,
    lwd = c(0.1, 0.1, 0.1),
    lty = 1,
    alpha = c(0.2, 0.8, 0.15),
    col = scico::scico(7,
      palette = "turku"
    )[c(2, 3, 2)]
  ) +
  geom_sf(
    data = data_253,
    lwd = 0.15,
    alpha = 1,
    col = scico::scico(7,
      palette = "bilbao"
    )[6]
  ) +
  geom_sf(
    data = waterholes,
    col = scico::scico(5,
      palette = "nuuk"
    )[1],
    alpha = 0.45
    # col = "grey"
  ) +
  geom_sf(
```

```
  data = kruger,
  col = "grey50",
  lwd = 0.2,
  fill = NA,
  lty = 2
) +
geom_textbox(
  data = textdata,
  aes(
    x, y,
    label = label
  ),
  family = "IBM Plex Sans",
  size = 4,
  colour = "grey30",
  fill = alpha("antiquewhite", 0.3),
  box.color = NA
) +
scale_fill_gradientn(
  colours = scico(10,
    palette = "vik",
    direction = 1,
    begin = 0., end = 1
  )
  # colours = rev(RColorBrewer::brewer.pal(9, "RdYlBu"))#,
  # guide = guide_colorbar(frame.colour = "grey40",
  #                        ticks.colour = "grey40")
) +
annotation_north_arrow(
  style = north_arrow_minimal(
    text_family = "IBM Plex Sans",
    text_size = 10,
    text_col = "grey50",
    line_col = "grey50",
    fill = "grey50"
  ),
  location = "br"
) +
annotation_scale(
  bar_cols = c("grey50", "grey90"),
  height = unit(1, units = "mm"),
  text_family = "IBM Plex Sans"
) +
theme_void() +
theme(
  panel.background = element_rect(
    colour = "grey",
    fill = "grey90"
  ),
  plot.margin = unit(rep(5, 4), "mm")
) +
coord_sf(
  xlim = bbox[c("xmin", "xmax")],
  ylim = bbox[c("ymin", "ymax")],
```

```
    expand = FALSE
  ) +
  annotation_custom(
    grob = ggplotGrob(
      fig_inset_b
    ),
    xmin = bbox[c("xmin")] + 1000,
    xmax = bbox[c("xmin")] + 7500,
    ymax = bbox["ymin"] + 13500,
    ymin = bbox["ymin"] + 1000
  ) +
  annotation_custom(
    grob = ggplotGrob(
      fig_inset_a
    ),
    xmin = bbox[c("xmin")] + 1000,
    xmax = bbox[c("xmin")] + 7500,
    ymax = bbox["ymin"] + 21000,
    ymin = bbox["ymin"] + 13600
  )

ggsave(fig_main,
  filename = "figures/fig_map4.png",
  height = 9, width = 12
)
```

2.9 Make where plot

```
fig_inset_a <-
  ggplot() +
  geom_sf(
    data = za,
    fill = "tan",
    # alpha = 0.5,
    show.legend = F,
    col = NA
  ) +
  geom_sf(
    data = kruger_point,
    size = 5
  ) +
  scale_y_continuous(
    breaks = seq(-22, -34, -6)
  ) +
  theme_void(base_size = 8) +
  theme(
    panel.background = element_rect(
      fill = "aliceblue",
      colour = "grey20"
    ),
    plot.margin = unit(rep(2, 4), "mm")
  ) +
  coord_sf(
    expand = T
  )
```

2.10 Make count plot

```
pal <- scico(7, palette = "turku")

fig_inset_b <-
  ggplot() +
  geom_sf(
    data = kruger,
    fill = "aliceblue",
    alpha = 0.8,
    col = NA,
    lwd = 0.2
  ) +
  geom_sf(
    data = data,
    lwd = c(0.1),
    lty = 1,
    alpha = c(0.4),
    col = pal[3]
  ) +
  annotate(
    geom = "rect",
    fill = NA,
    col = "grey20",
    lwd = 0.3,
    xmin = bbox["xmin"],
    xmax = bbox["xmax"],
    ymin = bbox["ymin"],
    ymax = bbox["ymax"]
  ) +
  annotation_scale(
    bar_cols = c("grey50", "grey70"),
    height = unit(1, units = "mm"),
    text_family = "IBM Plex Sans"
  ) +
  theme_void() +
  coord_sf(
    crs = 32736,
    expand = FALSE,
    xlim = c(325000, NA) # ,
    # ylim = c(7213050, 7335050)
  ) +
  theme(
    panel.background = element_rect(
      fill = "tan"
    ),
    panel.border = element_rect(
      colour = "grey20",
      fill = NA
      # size = 0.2,
    ),
    plot.margin = unit(rep(1, 4), "mm")
  )
```