John Ross in WWII

Chris Cote

2020-02-22

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.5.2
```

```
## -- Attaching packages ------------------------------------------------ tidyverse 1.3.0 --
```

```
## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.4
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

```
## Warning: package 'ggplot2' was built under R version 3.5.2
```

```
## Warning: package 'tibble' was built under R version 3.5.2
```

```
## Warning: package 'tidyr' was built under R version 3.5.2
```

```
## Warning: package 'purrr' was built under R version 3.5.2
```

```
## Warning: package 'dplyr' was built under R version 3.5.2
```

```
## Warning: package 'stringr' was built under R version 3.5.2
```

```
## Warning: package 'forcats' was built under R version 3.5.2
```

```
## -- Conflicts --------------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

create point data frame

```
jr_points <- read_csv("~/Documents/HKS/rtutorial/data/johnross/jr_raw.csv") %>%
  mutate(year = str_extract(`Date Arrived`, "-4[0-9]") %>% str_replace(., "-", ""),
         month = str_extract(`Date Arrived`, "[A-Z][a-z][a-z]") %>% str_replace_all(., "-", ""),
         day = str_extract(`Date Arrived`, "^[0-9]([0-9])?"),
         year = ifelse(is.na(year), lag(year), year),
         date = paste0("19",year,"-", month,"-", day) %>% as.Date("%Y-%b-%d")) %>%
  filter(!is.na(Lat)) %>%
  select(date, year, month, day, everything()) %>%
  mutate(index = 1:n()) %>%
  set_names(str_replace(str_to_lower(names(.)), " ", "_")) %>%
  mutate(date_arrived = as.Date(paste0("19", year, "-", month, "-", day), format = "%Y-%b-%d")) %>%
  arrange(date_arrived) %>%
  mutate(date_left = lead(date_arrived)) %>%
  mutate(location = paste(location_1, location_2, location_3, sep = ", ")) %>%
  select(date_arrived, date_left, location, what, lat, lon) %>%
  mutate(new_location = ifelse(location == lag(location), 0 , 1),
         new_location = ifelse(row_number() == 1, 0, new_location),
         new_location = cumsum(new_location)) %>%
```

```
  group_by(new_location) %>%
  fill(what) %>%
  ungroup() %>%
  distinct(new_location, .keep_all = TRUE) %>%
  mutate(date_left = lead(date_arrived)) %>%
  mutate(country = ifelse(str_detect(location, "Italy"), "Italy",
                          ifelse(str_detect(location, "USA"), "USA", "NA")),
         lead_country = lead(country),
         lead_country = ifelse(is.na(date_left), "USA", lead_country),
         map = paste0(country, lead_country))
```

```
## Parsed with column specification:
## cols(
##   `Date Arrived` = col_character(),
##   `Location 1` = col_character(),
##   `Location 2` = col_character(),
##   `Location 3` = col_character(),
##   What = col_character(),
##   Transport = col_character(),
##   Lat = col_double(),
##   Lon = col_double()
## )
```

```
jr_labels <-
   jr_points %>% distinct(date_arrived, date_left, location, lat, lon) %>%
   mutate(label = paste0(location, ": ",  scales::date_format("%b-%d-%y")(date_arrived), " to ", scales
   mutate(label = str_replace(label, ", NA", "")) %>%
   mutate(country = ifelse(str_detect(location, "Italy"), "Italy",
                           ifelse(str_detect(location, "USA"), "USA", "NA")),
          lead_country = lead(country),
          lead_country = ifelse(is.na(date_left), "USA", lead_country),
          map = factor(paste0(country, lead_country))) %>%
   mutate(duration = as.numeric(date_left - date_arrived)) %>%
   group_by(location) %>%
   mutate(duration = sum(duration, na.rm = TRUE)) %>%
   ungroup()

jr_labels
```

```
## # A tibble: 25 x 10
##    date_arrived date_left  location   lat   lon label country lead_country map
##    <date>       <date>     <chr>    <dbl> <dbl> <chr> <chr>   <chr>        <fct>
##  1 1943-03-08   1943-04-03 Fort Di~  40.0 -74.7 Fort~ USA     USA          USAU~
##  2 1943-04-03   1943-07-01 Miami B~  25.8 -80.2 Miam~ USA     USA          USAU~
##  3 1943-07-01   1943-10-01 Fort Be~  38.7 -77.2 Fort~ USA     USA          USAU~
##  4 1943-10-01   1943-12-15 Will Ro~  35.4 -97.6 Will~ USA     USA          USAU~
##  5 1943-12-15   1944-04-14 Nashvil~  36.2 -87.1 Nash~ USA     USA          USAU~
##  6 1944-04-14   1944-12-08 DeRidde~  30.9 -93.3 DeRi~ USA     USA          USAU~
##  7 1944-12-08   1945-01-01 Greensb~  36.1 -80.0 Gree~ USA     USA          USAU~
##  8 1945-01-01   1945-01-13 Camp Pa~  37.1 -76.5 Camp~ USA     USA          USAU~
##  9 1945-01-13   1945-01-25 NYC, NY~  40.7 -74.3 NYC,~ USA     Italy        USAI~
## 10 1945-01-25   1945-01-25 Naples,~  40.9  14.2 Napl~ Italy   Italy        Ital~
## # ... with 15 more rows, and 1 more variable: duration <dbl>
```

Create long function to make map plot

```
make_map <- function(choose_map) {
 jr_labels <-
   jr_points %>% distinct(date_arrived, date_left, location, lat, lon) %>%
   mutate(label = paste0(location, ": ",  scales::date_format("%b-%d-%y")(date_arrived), " to ", scales
   mutate(label = str_replace(label, ", NA", "")) %>%
   mutate(country = ifelse(str_detect(location, "Italy"), "Italy",
                           ifelse(str_detect(location, "USA"), "USA", "NA")),
          lead_country = lead(country),
          lead_country = ifelse(is.na(date_left), "USA", lead_country),
          map = factor(paste0(country, lead_country))) %>%
   mutate(duration = as.numeric(date_left - date_arrived)) %>%
   group_by(location) %>%
   mutate(duration = sum(duration, na.rm = TRUE)) %>%
   ungroup() %>%
   mutate(duration = (scale(duration) + 3)^3)
  if(missing(choose_map)){

   choose_map <- ""


  }else{
    jr_points <- jr_points %>%
    filter(str_detect(map, choose_map))
  jr_labels <- jr_labels %>%
    filter(str_detect(map, choose_map)) }



  if(!missing(choose_map) & choose_map == "USAUSA"){
    map_usa <- ggplot2::map_data("state")
ggplot(jr_points) +
    geom_map(data = map_usa, map = map_usa, aes(map_id = region),
             fill = "gray90", alpha = 1/2,
             color = "gray50") +
    geom_curve(data = jr_points %>%
                 mutate(alternate = rep_along(date_arrived, 1:2),
                        alternate = factor(alternate)),
               aes(x = lon, xend = lead(lon), y = lat, yend = lead(lat), lty = alternate),
               arrow = arrow(length = unit(0.2, "inches"), type = "closed"), show.legend = FALSE) +
    geom_point(data = jr_labels,
               aes(x = lon, y = lat, color = location, size =  duration), show.legend = FALSE) +
    ggrepel::geom_label_repel(data = jr_labels %>%
                                mutate(number = row_number(),
                                       number_location = paste0(number, ": ", str_replace_all(label, ",
                                group_by(location) %>%
                                mutate(number_location = paste(number_location, collapse = "\\\n")) %>%
                                distinct(number_location, lat, lon),
                              aes(x = lon, y = lat, label = number_location), size = 2.7, show.legend =
    coord_equal() +
    theme_void()
  }else {
    world <- ggplot2::map_data("world")
    ggplot(jr_points) +
```

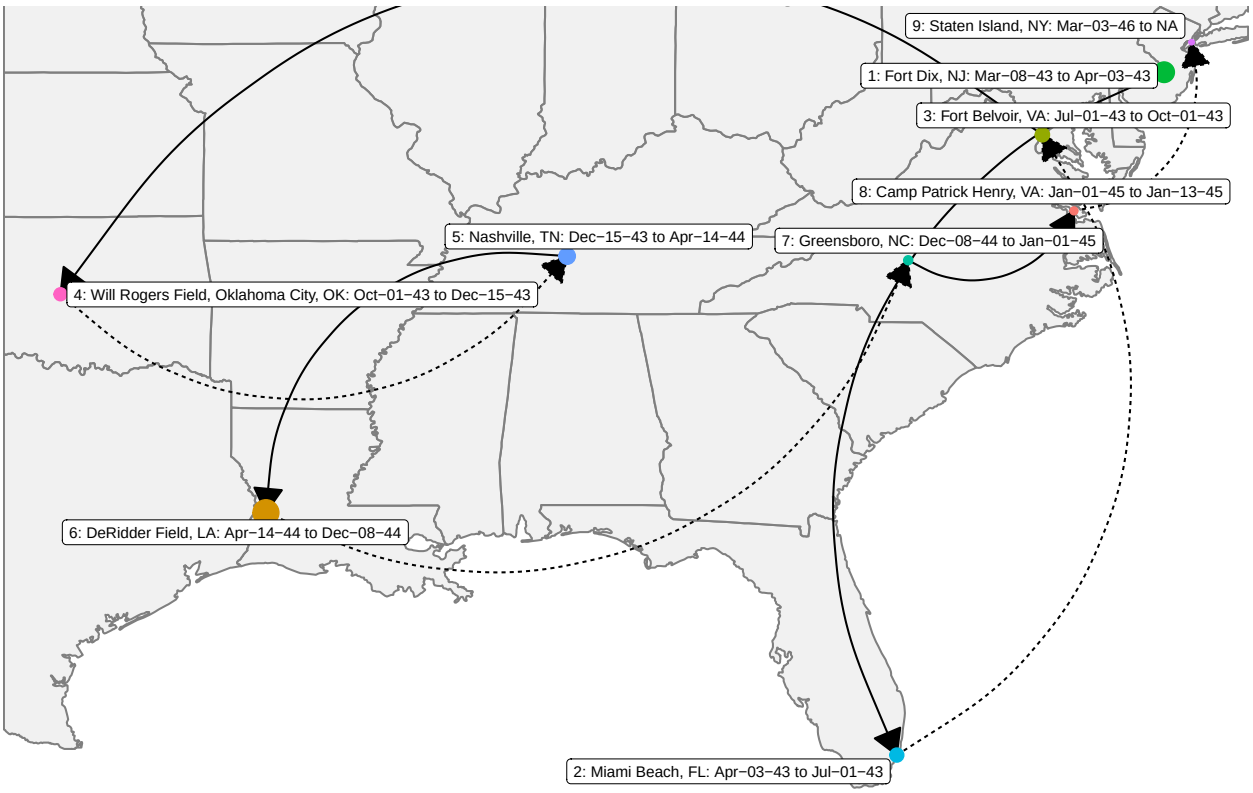
```
      geom_map(data = world, map = world, aes(map_id = region),
               fill = "gray90", alpha = 1/2,
               color = "gray50") +
      geom_curve(data = jr_points %>%
                   mutate(alternate = rep_along(date_arrived, 1:2),
                          alternate = factor(alternate)),
                 aes(x = lon, xend = lead(lon), y = lat, yend = lead(lat), lty = alternate),
                 arrow = arrow(angle = 120, length = unit(0.1, "inches"), type = "closed"), show.legend
      geom_point(data = jr_labels,
                 aes(x = lon, y = lat, color = location, size = log(duration)),  show.legend = FALSE) +
      ggrepel::geom_label_repel(data = jr_labels %>%
                                  mutate(number = row_number(),
                                         number_location = paste0(number, ": ", str_replace_all(label, "
                                  group_by(location) %>%
                                  mutate(number_location = paste(number_location, collapse = "\\\n")) %>%
                                  distinct(number_location, lat, lon),
                                aes(x = lon, y = lat, label = number_location), size = 3, show.legend = F
      coord_equal() +
      theme_void() +
      theme(panel.background = element_rect(fill = "NA"))
  }

}
```

use the function!

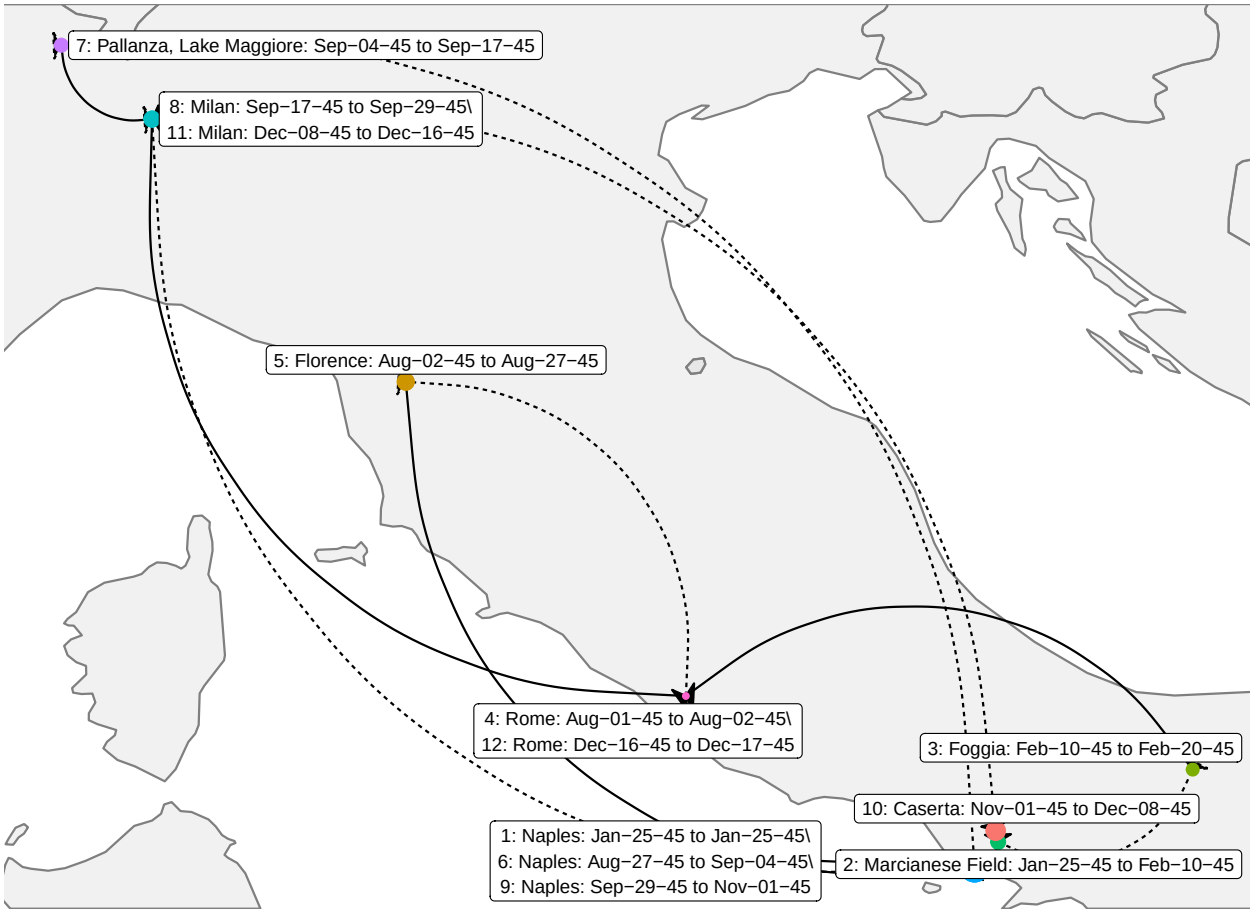
Service in USA

```
make_map("USAUSA")
```



Service in Italy

```
make_map("ItalyItaly")
```


7: Pallanza, Lake Maggiore: Sep−04−45 to Sep−17−45
8: Milan: Sep−17−45 to Sep−29−45\
11: Milan: Dec−08−45 to Dec−16−45
5: Florence: Aug−02−45 to Aug−27−45
4: Rome: Aug−01−45 to Aug−02−45\
12: Rome: Dec−16−45 to Dec−17−45
3: Foggia: Feb−10−45 to Feb−20−45
10: Caserta: Nov−01−45 to Dec−08−45
1: Naples: Jan−25−45 to Jan−25−45\
6: Naples: Aug−27−45 to Sep−04−45\
9: Naples: Sep−29−45 to Nov−01−45
2: Marcianese Field: Jan−25−45 to Feb−10−45