Forecasting Demo

Josh Tyler

Introduction

This is an example analysis report for the mpox forecasting competition. Here we outline an analysis workflow that starts at loading in the data and ends with the creation of the Estimates in the correct submission format.

Organising The Data

In order to create a forecast, we must first load and clean the raw Incidence data. The historic data was filtered to include only rows where there is no missing data and then joined to a table providing England regions. We have left the incidence rates as per 100,000 population.

```
library(tidyverse)
library(sf)
library(geodata)

dat <- read.csv("data/FULL-UKHSA-2017-2022-Lyme-Disease.csv") |>
  filter(!is.na(Value))

dat <- dat |>
  filter(Area.Type %in% c("UA","District")) |>
  select(Area.Code,Area.Name,Area.Type,
         Time.period,Value,Denominator,
         Lower.CI.95.0.limit,Upper.CI.95.0.limit)


region_lookup<-read.csv("data/Region_Lookup.csv") |>
  select(LTLA22CD,LTLA22NM,UTLA22CD,UTLA22NM,RGN22CD,RGN22NM) |>
  distinct()
```

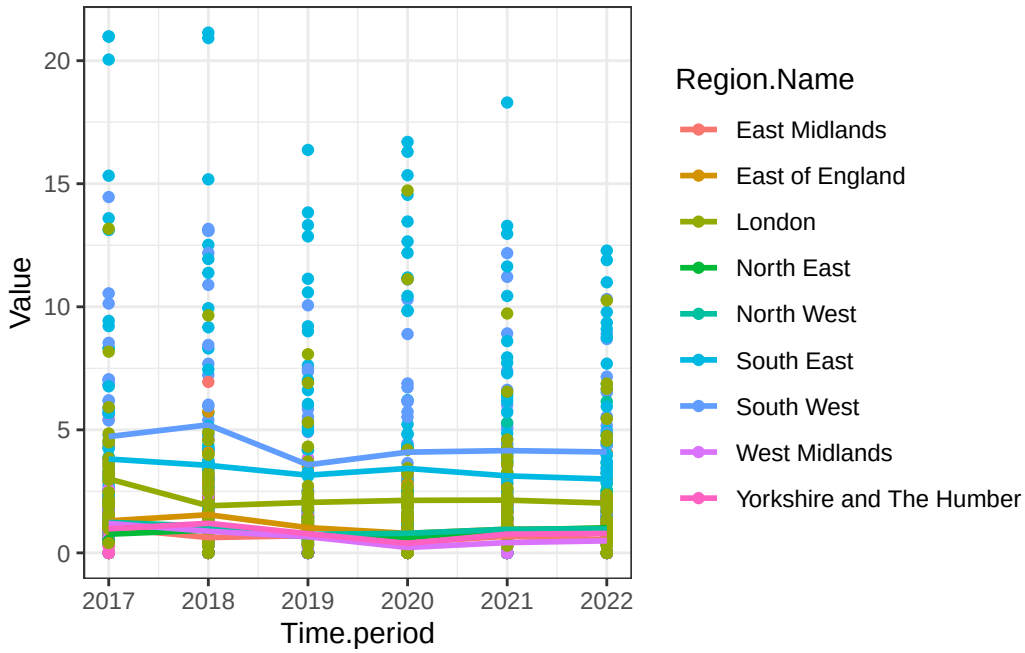
```r
df<- dat |>
  left_join(region_lookup |>
              select("LTLA22CD","RGN22CD","RGN22NM") |>
              rename("Area.Code" = "LTLA22CD",
                     "Region.Code"= "RGN22CD",
                     "Region.Name" = "RGN22NM"),
            by = "Area.Code"
            )
```

Plotting The Data

In order to assess whether there is a significant difference between English Regions, we have plotted Incidence against time, with a mean value for each region as a line plot.

```r
# Calculate average Value per Region.Name and Time.period
average_df <- df %>%
  group_by(Region.Name, Time.period) %>%
  summarise(Average_Value = mean(Value, na.rm = TRUE))
```

```r
# Create the ggplot with points and lines
ggplot(data = df) +
  geom_point(aes(x = Time.period, y = Value, colour = Region.Name)) +
  geom_line(data = average_df,
            aes(x = Time.period, y = Average_Value,
                colour = Region.Name), size = 1) +
  theme_bw()
```



Creating a Forecast

This example uses a simple regression model whereby each area has its own coefficients.

```
results<-expand_grid(Area.Name=unique(dat$Area.Name),Time.period=c(2023,2024)) |>
  as.data.frame()

model<-lm(formula = Value ~ Time.period + Area.Name,data = df)

prediction<-predict(object = model,newdata = results)

prediction[which(prediction<0)]<-0

results<-cbind(results,prediction)
```

Creating the results table

We have constructed a data.frame to match the expected output columns, as given in the output template.

```r
library(gt)

forecast<-results |>
  select(Time.period,Area.Name,prediction) |>
  rename("Year"=Time.period,"Council"="Area.Name","Incidence"=prediction) |>
  mutate(Lower_95CI=Incidence) |>
  mutate(Upper_95_CI=Incidence) |>
  arrange(Year,Council)

write.csv(x = forecast,file = "example_forecast.csv")

#gt(forecast)
```