

Untitled

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
library(readr)
library(readr)
library(ggplot2)
library(tidyverse)
library(lubridate)
library(epiR)
library(incidence)
library(janitor)

influenza_NY <- read_csv("Influenza_Laboratory-Confirmed_Cases_By_County__Beginning_2009-10_Season.csv")
clean_names() %>% remove_empty()
```

Including Plots

You can also embed plots, for example:

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

```
rowMeans(influenza_NY[sapply(influenza_NY, is.numeric)])
rowSums(influenza_NY[sapply(influenza_NY, is.numeric)])
colSums(influenza_NY[sapply(influenza_NY, is.numeric)])
colMeans(influenza_NY[sapply(influenza_NY, is.numeric)])

sum(central$count)
sum(capital$count)
sum(metro$count)
sum(NYC$count)
sum(western$count)
sum(influenza_unknown$count)
sum(influenzaA$count)
sum(influenzaB$count)

t.test(influenza_unknown$count, influenzaA$count)
t.test(influenza_unknown$count, influenzaB$count)
```

```
t.test(influenzaA$count, influenzaB$count)
```

```
t.test(metro$count, NYC$count)
t.test(NYC$count, capital$count)
```

```
library(scales)
library(pacman)
library(incidence)
library(patchwork)
```

```
influenza_region <- ggplot(influenza_NY, aes(x=region, y=count,)) +
  theme(text = element_text(size = 10)) +
  geom_boxplot()
plot(influenza_region)
```

```
influenza_disease<- ggplot(influenza_NY, aes(x=disease, y=count)) +
  geom_boxplot()
plot(influenza_disease)
```

```
epicurve_influenza <- ggplot(influenza_NY, aes(x= influenza_NY$week_ending_date , y=count)) +
  geom_line(aes(color = disease)) +
  labs(title = "Influenza Disease Type Trends",
        y = "Count", x = "Year") +
  coord_cartesian(ylim= c(0, 1500), expand = FALSE) +
  facet_wrap(disease ~ .)
```

```
influenza_region/ epicurve_influenza
```

```
library(ggplot2)
library(dplyr)
library(magrittr)
library(outbreaks)
library(earlyR)
library(projections)
library(prophet)
```

```
incidence_influenza_NY <- as.data.frame(incidence(influenza_NY$week_ending_date)) %>%
  rename("I" = "counts")
```

```
plot(incidence_influenza_NY)
```

```
incidence_influenza_NY2 <- as.incidence(influenza_NY$count, dates = influenza_NY$week_ending_date, inter=1)
plot(incidence_influenza_NY2)
```

```
is_flu_season <- function(ds) {
  dates <- as.Date(ds)
  month <- as.numeric(format(dates, '%m'))
  return(month >= 10 | month <= 5)
}
```

```
ny_incid$on_season <- is_flu_season(ny_incid$ds)
ny_incid$off_season <- !is_flu_season(ny_incid$ds)
```

```
flu_season_predict<- prophet(weekly.seasonality=FALSE)
```

```

flu_season_predict<- add_seasonality(flu_season_predict, name='weekly_on_season', period=7, fourier.order=3)
flu_season_predict<- add_seasonality(flu_season_predict, name='weekly_off_season', period=7, fourier.order=3)
flu_season_predict<- fit.prophet(flu_season_predict, ny_incid)

forecast <- make_future_dataframe(flu_season_predict, periods= 104, freq = "week") %>%
  filter(is_flu_season(ds))

forecast$on_season <- is_flu_season(forecast$ds)
forecast$off_season <- !is_flu_season(forecast$ds)

f2 <- predict(flu_season_predict, forecast)

plot_predictions<- prophet_plot_components(flu_season_predict, f2)
plot_predictions2<- dyplot.prophet(flu_season_predict, f2)
plot_predictions2

```