HW3 by AJ Acacio

Preliminaries

Recap:

To recap, the three terms I chose and downloaded 2004-present Google Trends data for were:

- University of Pennsylvania
- Johns Hopkins University
- London School of Economics

Note that for the purposes of this project, I use only worldwide data for the following reasons:

- While the universities are Western, they are all well-known globally.
- In my opinion, the geographies I selected in HW2 offer limited additional data. The data for the United States simply mirrors global trends, while the data for China appears to be limited as some months yield zero hits.
- I used `gtrendsR` and attempted to make my code reproducible. If we we want to analyze or visualize the data from the United States and China, we can use the exact same code and simply add an argument to specify certain geographies.

Loading Libraries:

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──
```

```
## ✓ ggplot2 3.3.5     ✓ purrr   0.3.4
## ✓ tibble  3.1.5     ✓ dplyr   1.0.7
## ✓ tidyr   1.1.4     ✓ stringr 1.4.0
## ✓ readr   2.1.1     ✓ forcats 0.5.1
```

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

```
library(readr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(ggplot2)
library(gtrendsR)
library(ggsci)
library(ggthemes)
library(tidyquant)
```

```
## Loading required package: PerformanceAnalytics
```

```
## Loading required package: xts
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
##
## Attaching package: 'xts'
```

```
## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```
##
## Attaching package: 'PerformanceAnalytics'
```

```
## The following object is masked from 'package:graphics':
##
##     legend
```

```
## Loading required package: quantmod
```

```
## Loading required package: TTR
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
## == Need to Learn tidyquant? ====================================================
## Business Science offers a 1-hour course - Learning Lab #9: Performance Analysis & Por
tfolio Optimization with tidyquant!
## </> Learn more at: https://university.business-science.io/p/learning-labs-pro </>
```

```
library(directlabels)
```

Question 1

Instead of downloading.csv data as we did in HW2, I decided to use the `gtrendsR` package, which will automate many parts of my analysis. This package automatically retrieves up-to-date information from Google Trends, and all I have to do is specify the search terms, time frame, and geography.

In visualizing this “raw” data, I used the geom_line as it best tracks the trends and allows us to follow even slight fluctuations in each term’s popularity over time.

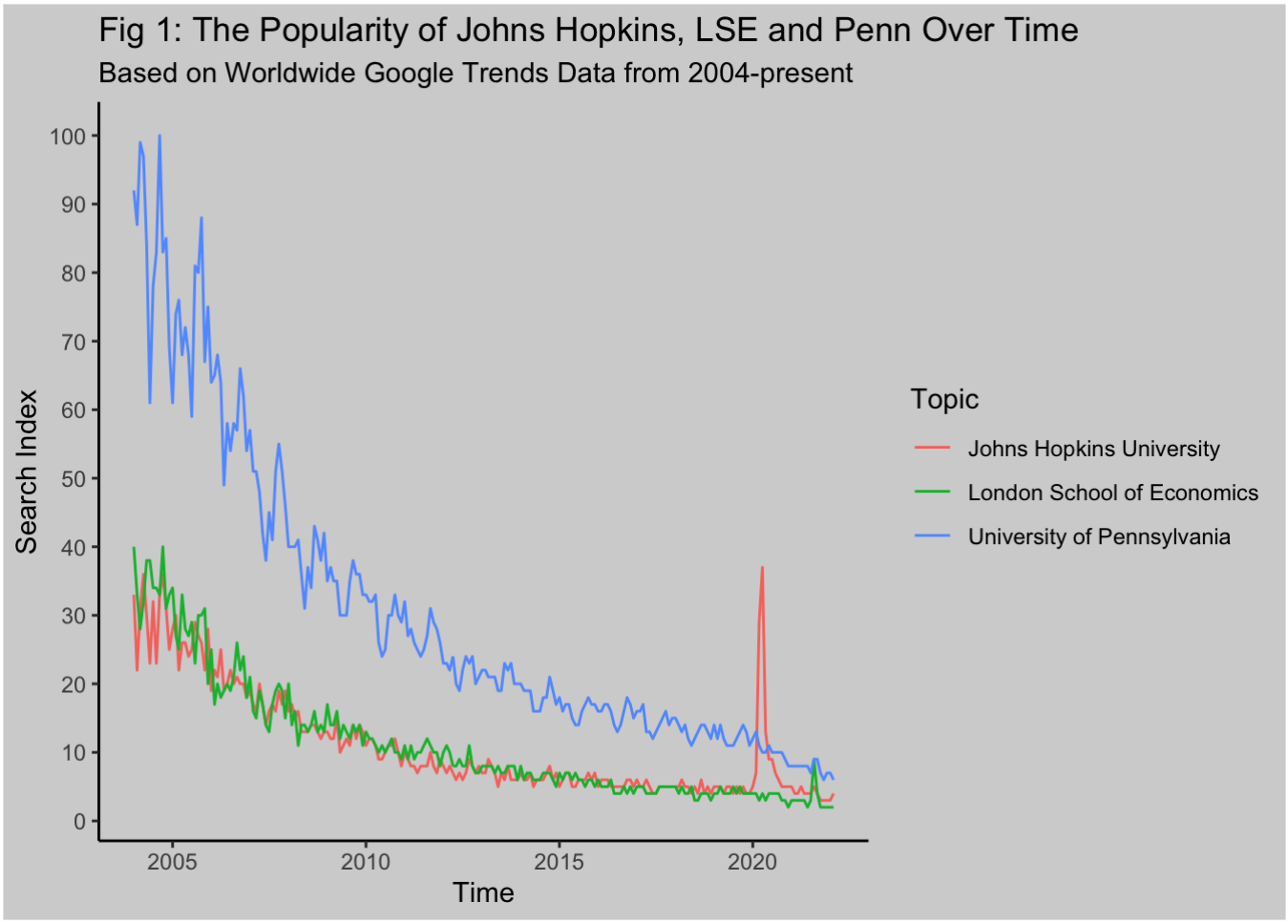
```
trends <- gtrends(c("University of Pennsylvania",
                    "Johns Hopkins University",
                    "London School of Economics"),
                  time = "all")

trend_over_time <- trends$interest_over_time %>%
  mutate(search_index=ifelse(hits=="<1",
                             0.5,
                             as.numeric(hits)),
               date=as.Date(date)) %>%
  rename(Topic = keyword)

trend_over_time
```

date	hits	Topic	geo	ti…	gpr…	category	search_index
<date>	<int>	<chr>	<chr>	<chr>	<chr>	<int>	<dbl>
2004-01-01	92	University of Pennsylvania	world	all	web	0	92
2004-02-01	87	University of Pennsylvania	world	all	web	0	87
2004-03-01	99	University of Pennsylvania	world	all	web	0	99
2004-04-01	97	University of Pennsylvania	world	all	web	0	97
2004-05-01	84	University of Pennsylvania	world	all	web	0	84
2004-06-01	61	University of Pennsylvania	world	all	web	0	61
2004-07-01	78	University of Pennsylvania	world	all	web	0	78

date	hits	Topic	geo	ti…	gpr…	category	search_index
<date>	<int>	<chr>	<chr>	<chr>	<chr>	<int>	<dbl>
2004-08-01	83	University of Pennsylvania	world	all	web	0	83
2004-09-01	100	University of Pennsylvania	world	all	web	0	100
2004-10-01	83	University of Pennsylvania	world	all	web	0	83

1-10 of 654 rows Previous **1** 2 3 4 5 6 … 66 Next

```
#Code for the Line Graph
trend_over_time %>% ggplot(aes(x=date, y=search_index, color= Topic))+
  geom_line() +
  labs(x = "Time",
       y = "Search Index",
       title = "Fig 1: The Popularity of Johns Hopkins, LSE and Penn Over Time",
       subtitle = "Based on Worldwide Google Trends Data from 2004-present") +
  theme_classic() +
  theme(plot.background = element_rect(fill ="lightgrey"),
        panel.background = element_rect(fill ="lightgrey"),
        legend.background = element_rect(fill ="lightgrey")) +
  scale_y_continuous(breaks = seq(from = 0, to = 100, by = 10))
```



Question 2

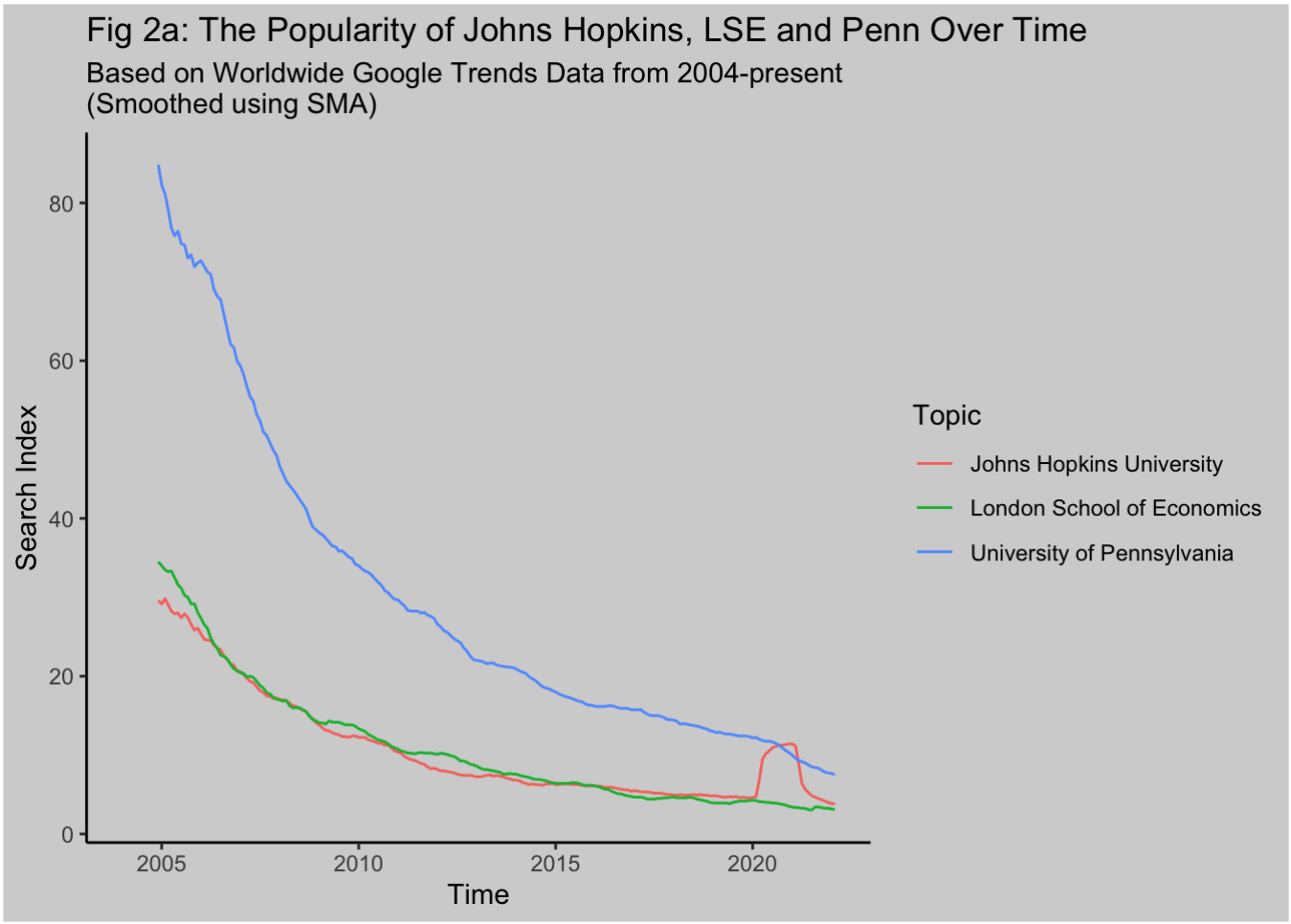
To plot the smoothed data, I will use the same code as above. However, I also installed a new package, `tidyquant`. I will use the `geom_ma` function of this package to plot the *simple moving average* of the search index instead of the particular value per month.

As seen below, this allows us to better observe the overall trend as opposed to the month-to-month fluctuations. Here, I've decided to use two ways of smoothing the data. The first (Fig 2a) uses the simple moving average, while the second (Fig 2b) uses the more sophisticated LOESS method. In both cases, the smoothing results in graphs of similar shapes. I would say that both methods work equally well, as unlike `method = "lm"` smoothing, they can accommodate for non-linear data.

In Fig 2a and 2b, we can observe that all three search terms show a downward trend across the years, with the steepest declines happening between 2004-2010. Does this mean that people are becoming less interested in higher education? This would be an interesting question to research, but because Google Trends data is limited, at this point we cannot draw broad conclusions.

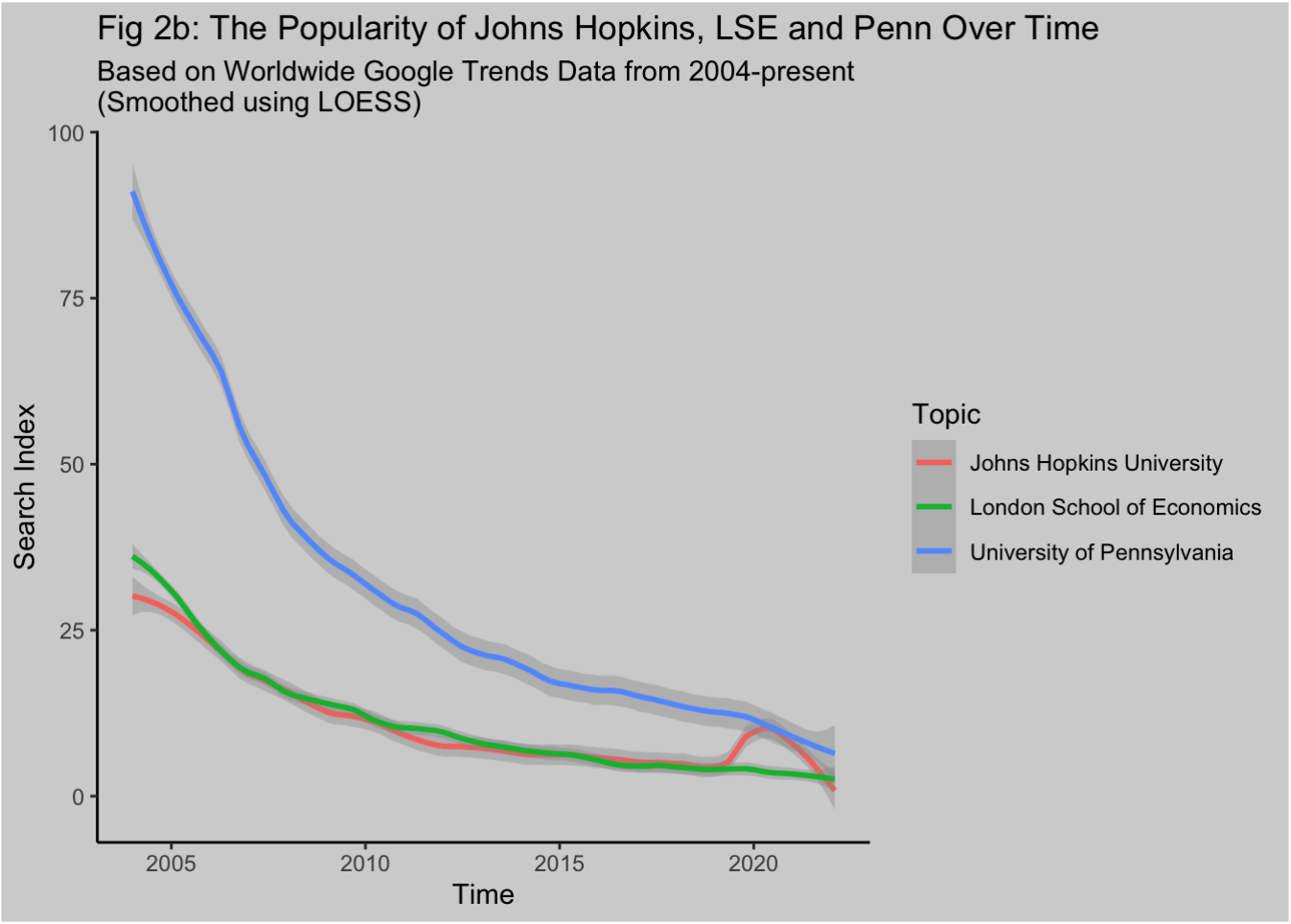
Another interesting observation is that despite either SMA or LOESS smoothing, Johns Hopkins University still shows a spike around 2020. This makes intuitive sense, as Johns Hopkins developed a Covid-19 dashboard which could explain why it was so popular during that time.

```
#Method 1: Smoothing the Graph by Calculating the Simple Moving Average (Fig 2a)
trend_over_time %>% ggplot(aes(x=date, y=search_index, color= Topic))+
  geom_ma(ma_fun = SMA,
          n = 12,
          linetype = "solid") +
  labs(x = "Time",
       y = "Search Index",
       title = "Fig 2a: The Popularity of Johns Hopkins, LSE and Penn Over Time",
       subtitle = "Based on Worldwide Google Trends Data from 2004-present \n(Smoothed u
sing SMA)") +
  theme_classic() +
  theme(plot.background = element_rect(fill ="lightgrey"),
        panel.background = element_rect(fill ="lightgrey"),
        legend.background = element_rect(fill ="lightgrey"))
```




```
#Method 2: Smoothing the Graph Using geom_smooth and LOESS (Fig 2b)
trend_over_time %>% ggplot(aes(x=date, y=search_index, color= Topic))+
  geom_smooth(method = "loess", span = 0.2, se=T) +
  labs(x = "Time",
       y = "Search Index",
       title = "Fig 2b: The Popularity of Johns Hopkins, LSE and Penn Over Time",
       subtitle = "Based on Worldwide Google Trends Data from 2004-present \n(Smoothed u
sing LOESS)") +
  theme_classic() +
  theme(plot.background = element_rect(fill ="lightgrey"),
        panel.background = element_rect(fill ="lightgrey"),
        legend.background = element_rect(fill ="lightgrey"))
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Question 3

To analyze seasonality, I will take the average search index by month for all years where data is available. In R, I do this by grouping the data by month and then plotting the average to visualize seasonal trends.

I believe that the best way to visualize seasonality is by a line graph with month plotted on the x-axis and average search index on the y-axis (Figures 3a(i), 3a(ii), 3a(iii)). But some people are more visual learners, so I'm also creating radar charts (Figures 3b(i), 3b(ii), 3b(iii)), which represent seasonality as how far from a symmetrical shape the web representing each search term is. What visualization is best depends on your audience.

I am plotting each university separately because their overall popularities are different, so we want to have customized scales in each visualization to more easily observe differences by month.

We can see that Johns Hopkins and Penn seem to have higher hits around March and September-October. In Figures 3a(i) and 3a(ii), we see this as a peak in the line. In Figures 3b(i) and 3b(ii) , this same information is represented as a "stretching out" of the web in the parts of the graph that correspond to these months.

These findings are interesting as these are times of the year that are important in the college application process. Most universities open applications in the Fall and release final decisions around March. Note that Johns Hopkins in particular shows a large peak around March, which I hypothesize is related to March 2020, when Covid-19 first hit. We can also observe dips around summer time, which makes intuitive sense for universities.

The LSE visualizations in Figures 3a(iii) and 3b(iii) show a dip around December, which again makes intuitive sense as this is during Winter break. The Spring/Fall peaks observed with Penn and Johns Hopkins data were not observed in the LSE data. If I had to hypothesize, this is probably because LSE follows a different academic timeline from the two American schools.

Creating a Data Frame for Seasonal Trends

```
seasonal_trends <- trend_over_time %>%
  separate(date, into = c("year", "month", "day"), sep = "-", convert = TRUE) %>%
  select(year, month, Topic, hits) %>%
  group_by(month, Topic) %>%
  summarize(hits = mean(hits))
```

```
## `summarise()` has grouped output by 'month'. You can override using the `.groups` arg
ument.
```

```
seasonal_trends
```

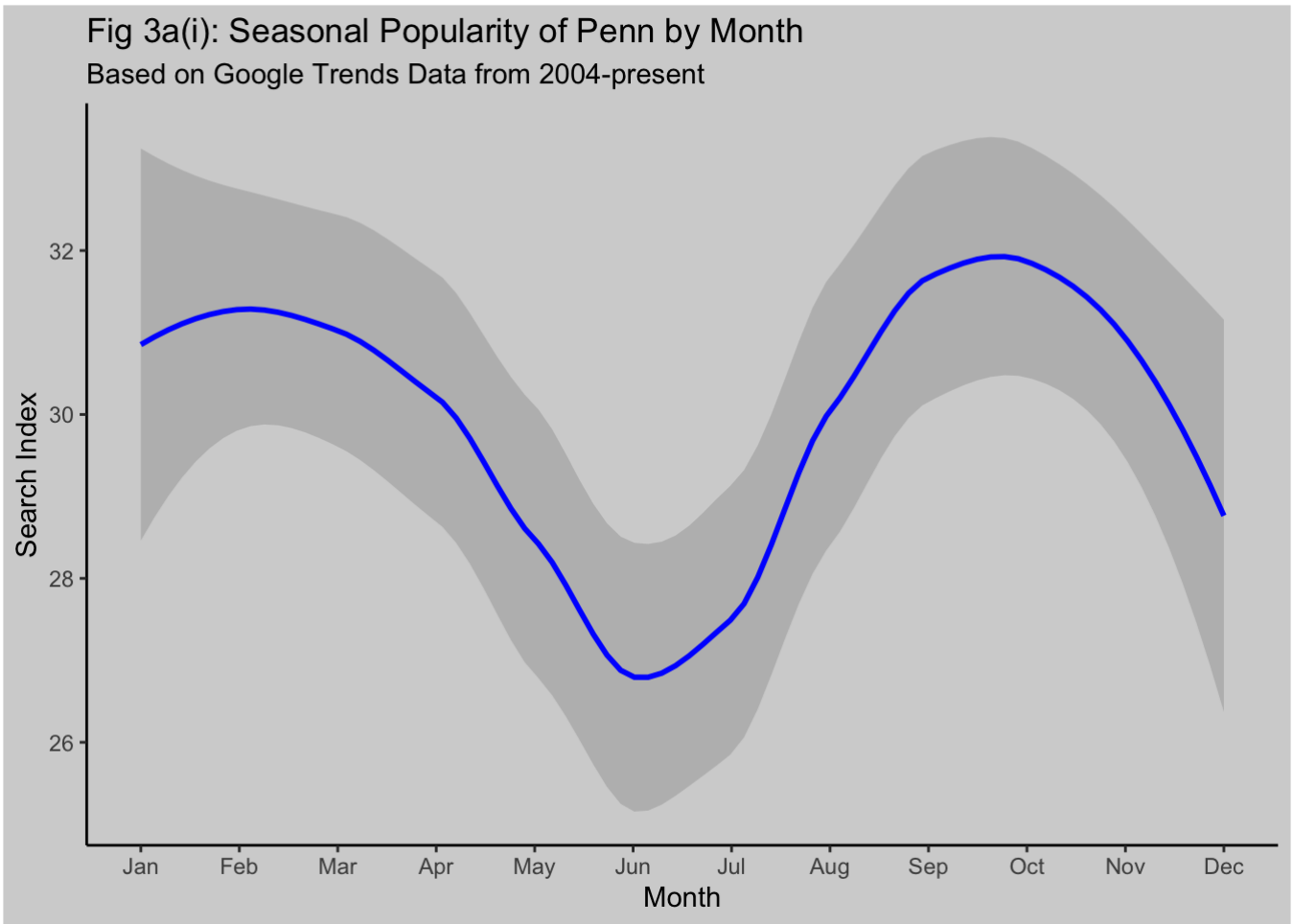
month	Topic	hits
<int>	<chr>	<dbl>
1	Johns Hopkins University	11.052632
1	London School of Economics	12.789474
1	University of Pennsylvania	30.789474
2	Johns Hopkins University	10.684211
2	London School of Economics	10.526316
2	University of Pennsylvania	30.421053
3	Johns Hopkins University	12.333333
3	London School of Economics	11.055556
3	University of Pennsylvania	32.666667
4	Johns Hopkins University	13.833333

1-10 of 36 rows Previous **1** 2 3 4 Next

Seasonality of Each University Visualized on Line Graphs

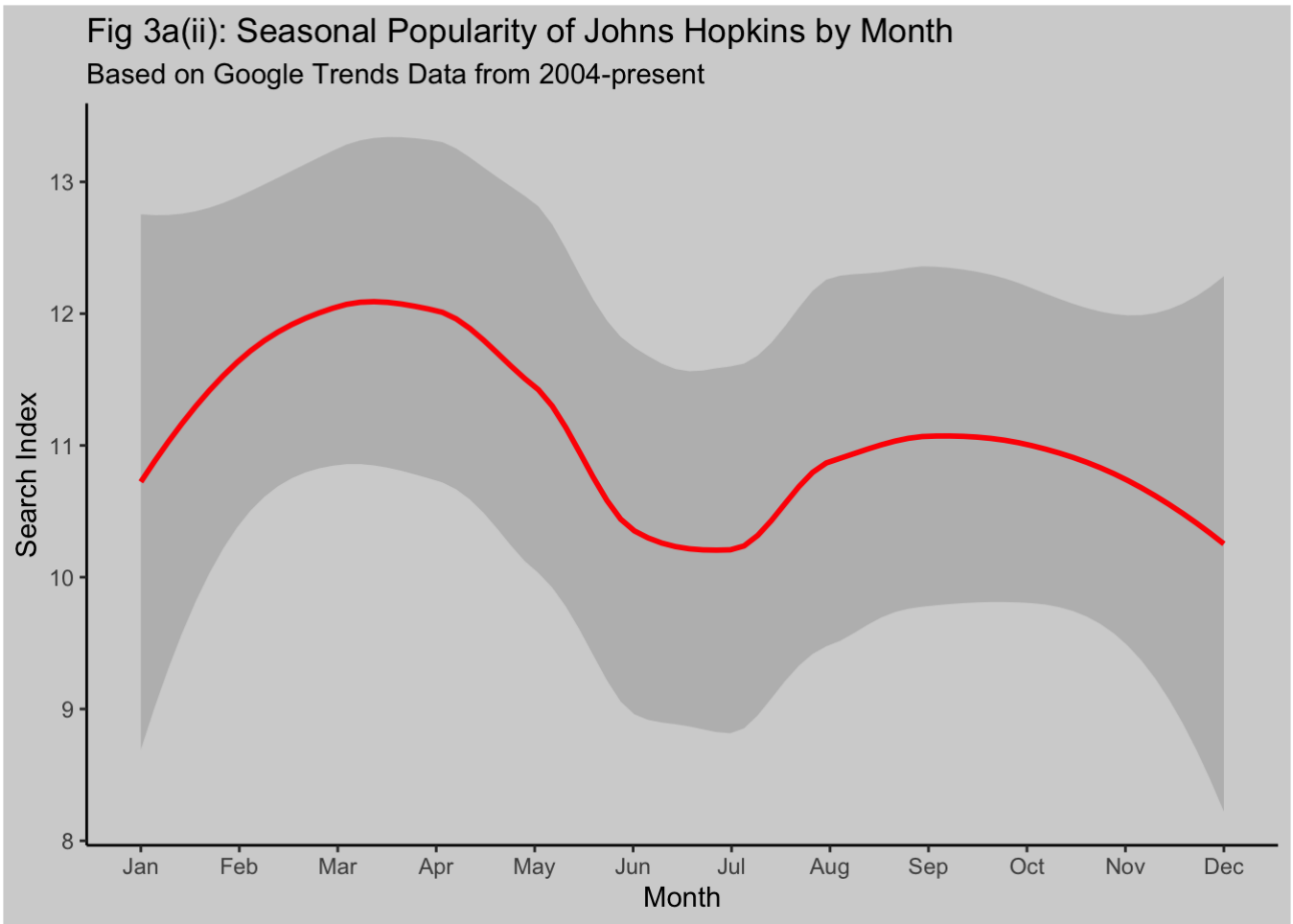
```
#Fig 3a(i): Seasonal Popularity of Penn by Month Visualized Through Line Graph
seasonal_trends %>%
  filter(Topic == "University of Pennsylvania") %>%
  ggplot(aes(x = month, y = hits)) +
  geom_smooth(method = loess, span = 0.1, color = "blue") +
  scale_x_continuous(breaks = 1:12, labels =  month.abb[1:12]) +
  labs(x = "Month",
       y = "Search Index",
       title = "Fig 3a(i): Seasonal Popularity of Penn by Month",
       subtitle = "Based on Google Trends Data from 2004-present") +
  theme_classic() +
  theme(plot.background = element_rect(fill ="lightgrey"),
        panel.background = element_rect(fill ="lightgrey"),
        legend.background = element_rect(fill ="lightgrey"))
```

```
## `geom_smooth()` using formula 'y ~ x'
```



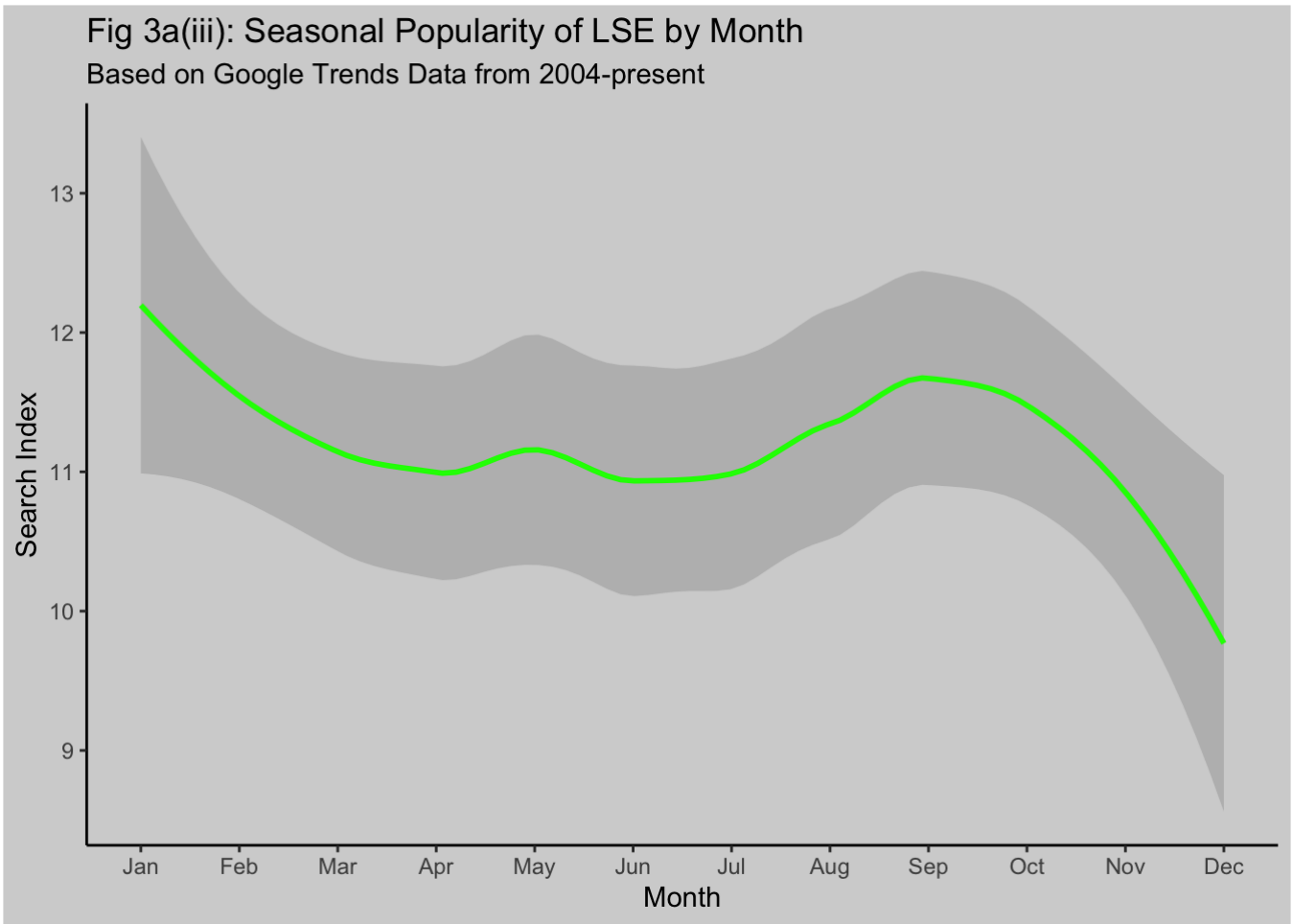
```
#Fig 3a(ii): Seasonal Popularity of Johns Hopkins by Month Visualized Through Line Graph
seasonal_trends %>%
  filter(Topic == "Johns Hopkins University") %>%
  ggplot(aes(x = month, y = hits)) +
  geom_smooth(method = loess, span = 0.1, color = "red") +
  scale_x_continuous(breaks = 1:12, labels =  month.abb[1:12]) +
  labs(x = "Month",
       y = "Search Index",
       title = "Fig 3a(ii): Seasonal Popularity of Johns Hopkins by Month",
       subtitle = "Based on Google Trends Data from 2004-present") +
  theme_classic() +
  theme(plot.background = element_rect(fill ="lightgrey"),
        panel.background = element_rect(fill ="lightgrey"),
        legend.background = element_rect(fill ="lightgrey"))
```

```
## `geom_smooth()` using formula 'y ~ x'
```



```
#Fig 3a(iii): Seasonal Popularity of LSE by Month Visualized Through Line Graph
seasonal_trends %>%
  filter(Topic == "London School of Economics") %>%
  ggplot(aes(x = month, y = hits)) +
  geom_smooth(method = loess, span = 0.1, color = "green") +
  scale_x_continuous(breaks = 1:12, labels =  month.abb[1:12]) +
  labs(x = "Month",
       y = "Search Index",
       title = "Fig 3a(iii): Seasonal Popularity of LSE by Month",
       subtitle = "Based on Google Trends Data from 2004-present") +
  theme_classic() +
  theme(plot.background = element_rect(fill ="lightgrey"),
        panel.background = element_rect(fill ="lightgrey"),
        legend.background = element_rect(fill ="lightgrey"))
```

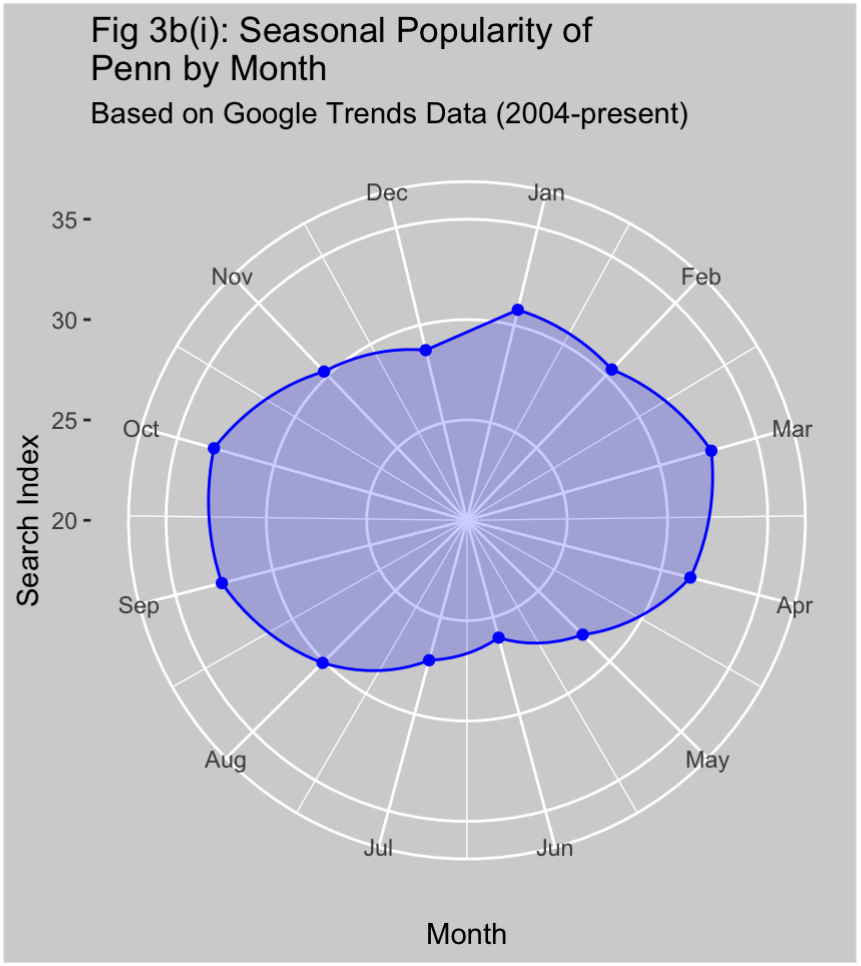
```
## `geom_smooth()` using formula 'y ~ x'
```



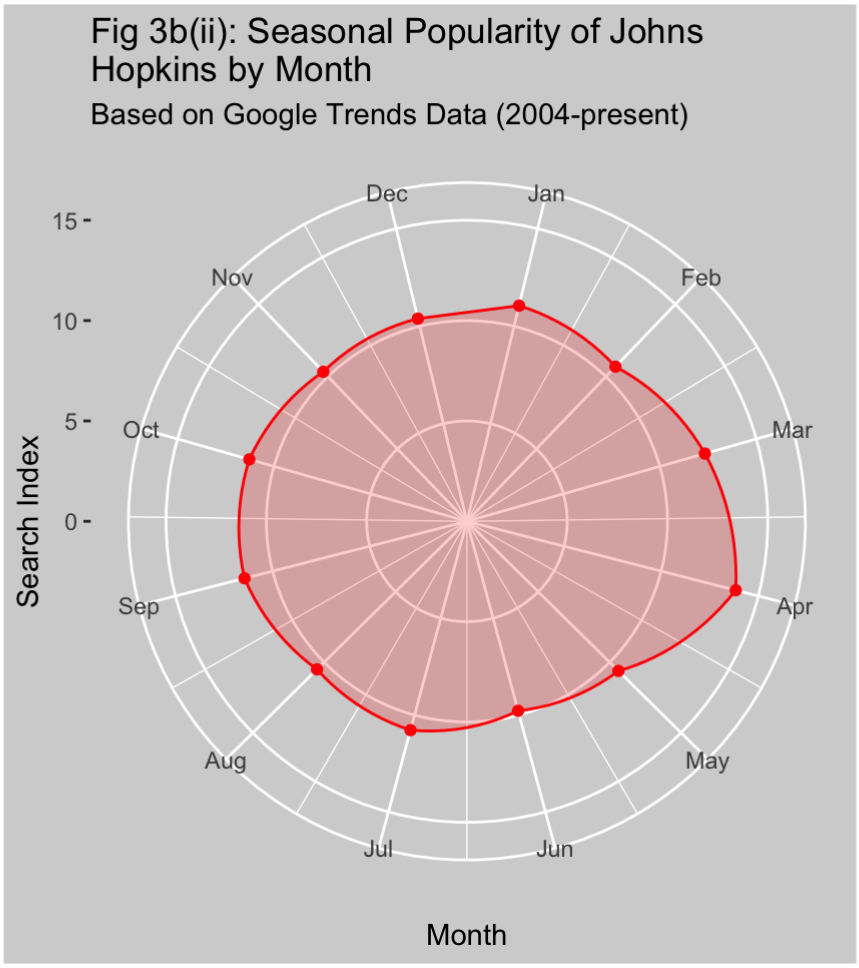
Seasonality of Each University Visualized on Radar Charts

```
#Fig 3b(i) Radar Chart for Penn
seasonal_trends %>%
  filter(Topic == "University of Pennsylvania") %>%
  ggplot(aes(x = month, y = hits)) +
  geom_col(position = "dodge", alpha = 0) +
  geom_point(color = "blue") +
  geom_polygon(fill = "blue", alpha = 0.2, color = "blue") +
  scale_x_continuous(breaks = 1:12, labels =  month.abb[1:12]) +
    labs(x = "Month",
       y = "Search Index",
       title = "Fig 3b(i): Seasonal Popularity of \nPenn by Month",
       subtitle = "Based on Google Trends Data (2004-present)") +
  scale_y_continuous(limits = c(20,35)) +
  coord_polar() +
    theme(plot.background = element_rect(fill ="lightgrey"),
        panel.background = element_rect(fill ="lightgrey"),
        legend.background = element_rect(fill ="lightgrey"))
```

```
## Warning: Removed 12 rows containing missing values (geom_col).
```



```
#Fig 3b(ii) Radar Chart for Johns Hopkins
seasonal_trends %>%
  filter(Topic == "Johns Hopkins University") %>%
  ggplot(aes(x = month, y = hits)) +
  geom_col(position = "dodge", alpha = 0) +
  geom_point(color = "red") +
  geom_polygon(fill = "red", alpha = 0.2, color = "red") +
  scale_x_continuous(breaks = 1:12, labels =  month.abb[1:12]) +
    labs(x = "Month",
       y = "Search Index",
       title = "Fig 3b(ii): Seasonal Popularity of Johns \nHopkins by Month",
       subtitle = "Based on Google Trends Data (2004-present)") +
  scale_y_continuous(limits = c(0,15)) +
  coord_polar() +
    theme(plot.background = element_rect(fill ="lightgrey"),
        panel.background = element_rect(fill ="lightgrey"),
        legend.background = element_rect(fill ="lightgrey"))
```



```
#Fig 3b(iii) Radar Chart for LSE
seasonal_trends %>%
  filter(Topic == "London School of Economics") %>%
  ggplot(aes(x = month, y = hits)) +
  geom_col(position = "dodge", alpha = 0) +
  geom_point(color = "darkgreen") +
  geom_polygon(fill = "darkgreen", alpha = 0.2, color = "darkgreen") +
  scale_x_continuous(breaks = 1:12, labels =  month.abb[1:12]) +
    labs(x = "Month",
       y = "Search Index",
       title = "Fig 3b(iii): Seasonal Popularity of LSE \n by Month",
       subtitle = "Based on Google Trends Data (2004-present)") +
  scale_y_continuous(limits = c(0,15)) +
  coord_polar() +
    theme(plot.background = element_rect(fill ="lightgrey"),
        panel.background = element_rect(fill ="lightgrey"),
        legend.background = element_rect(fill ="lightgrey"))
```