NY Open Statistical Programming Meetup - Feb 2019

Time Series, Two Ways:
Anomaly Detection & Forecasting

Catherine Zhou

codecademy

twitter @catherinezh

#rstats

#rstatsnyc

#rladies

@codecademy

About Me_

- Proud New YorkeR
- Currently @ Codecademy
- Formerly @ JetBlue &
  New York Sports Clubs

codecademy

Agenda

By the end of this talk you will be able to:

- Assess **data quality** in time series
- Forecast **seasonal trends**
- Manage **holidays + special events** in your forecast
- **Plot and visualize** *Google Trends* data
- Explore different **anomaly detection algorithms**
- Explain **case studies** for forecasting & anomaly detection

codecademy

PART ONE

The Origin Story_

... how I got roped into talking about time-series

codecademy

Jared Lander

oooh, show how you do time series forecasting

Jared Lander

oooh, show how you do time series forecasting

I have a complicated relationship with forecasting lol

Jared Lander

oooh, show how you do time series forecasting

I have a complicated relationship with forecasting lol

facebook

Basic Information

Relationship Status	It's complicated

Jared Lander

oooh, show how you do time series forecasting

I have a complicated relationship with forecasting lol

Ppl don't like hearing they don't have enough quality data to forecast well

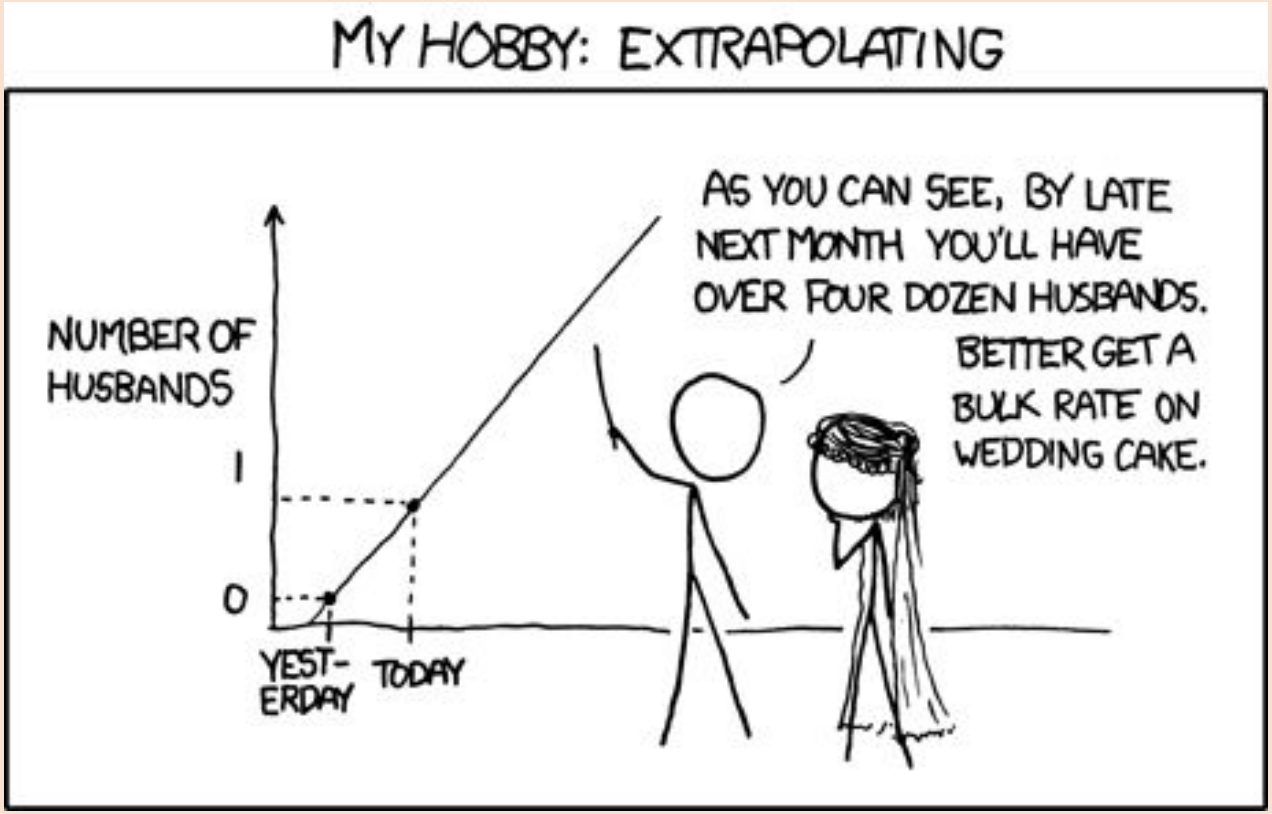
PART TWO

Forecasting with Time-Series

... what to do when your company confuses data science with fortune-telling

catherine zhou @catherinezh · 5h

one of my fave stories is speaking at a middle school career day in 2016. by the end of the day, the 6th graders were convinced i was a fortune teller and started asking me questions like: will the warriors make playoffs? will trump become pres?


MY HOBBY: EXTRAPOLATING
NUMBER OF HUSBANDS
1
0
YEST-ERDAY
TODAY
AS YOU CAN SEE, BY LATE NEXT MONTH YOU'LL HAVE OVER FOUR DOZEN HUSBANDS.
BETTER GET A BULK RATE ON WEDDING CAKE.

EXPECTATION

We want to work with data that is:

- Clean and well-organized
- Daily or weekly patterns
- Clear seasonal trends
- Key metrics to monitor
- Actionable insights

EXPECTATION

We want to work with data that is:

- Clean and well-organized
- Daily or weekly patterns
- Clear seasonal trends
- Key metrics to monitor
- Actionable insights

VS.

REALITY

We often work with data that has:

- Inconsistent trends and patterns
- Terabytes in size
- Missing or unclean data
- Multiple key metrics
  - Difficult to monitor
  - Difficult to interpret

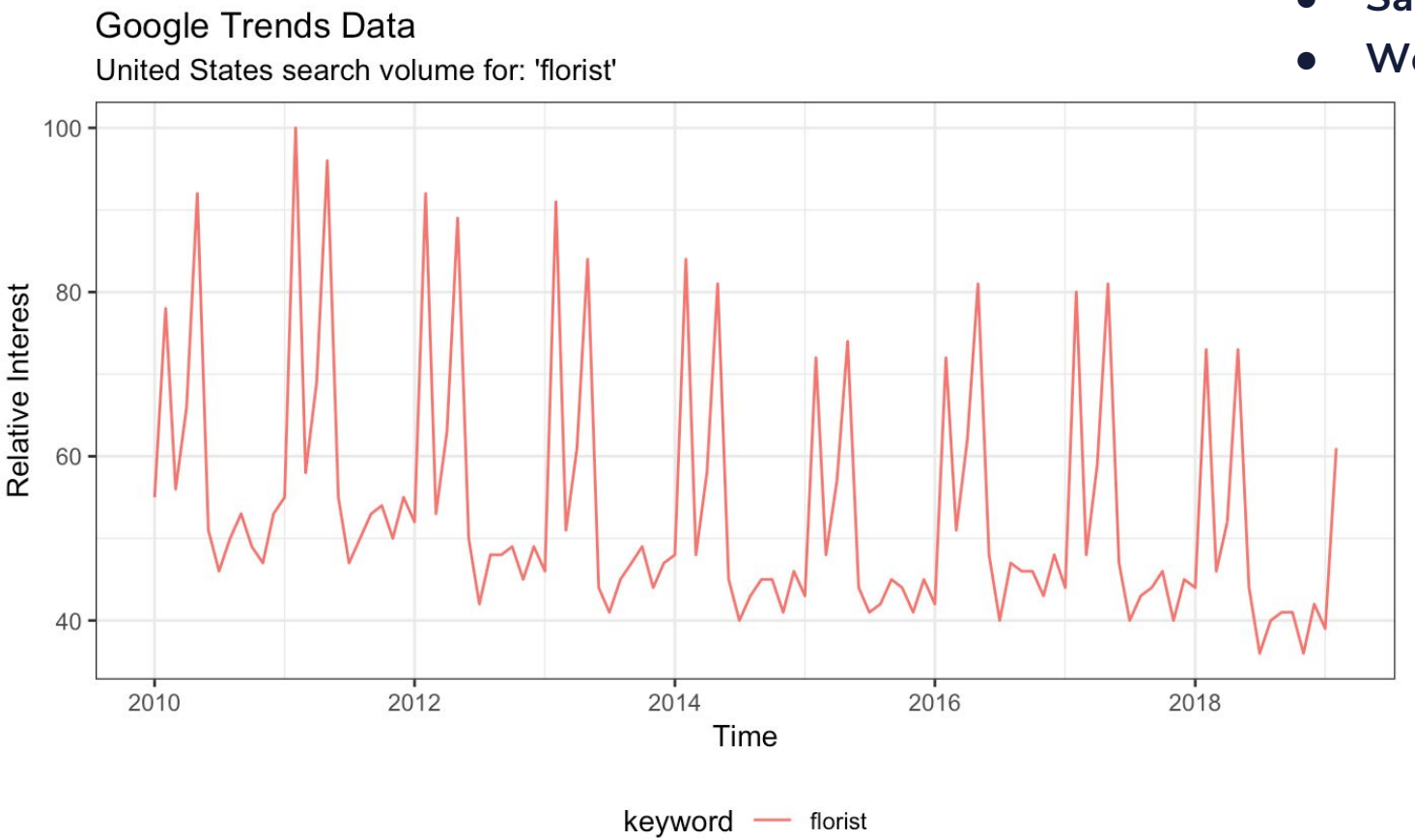
codecademy

When should we forecast our time series data?

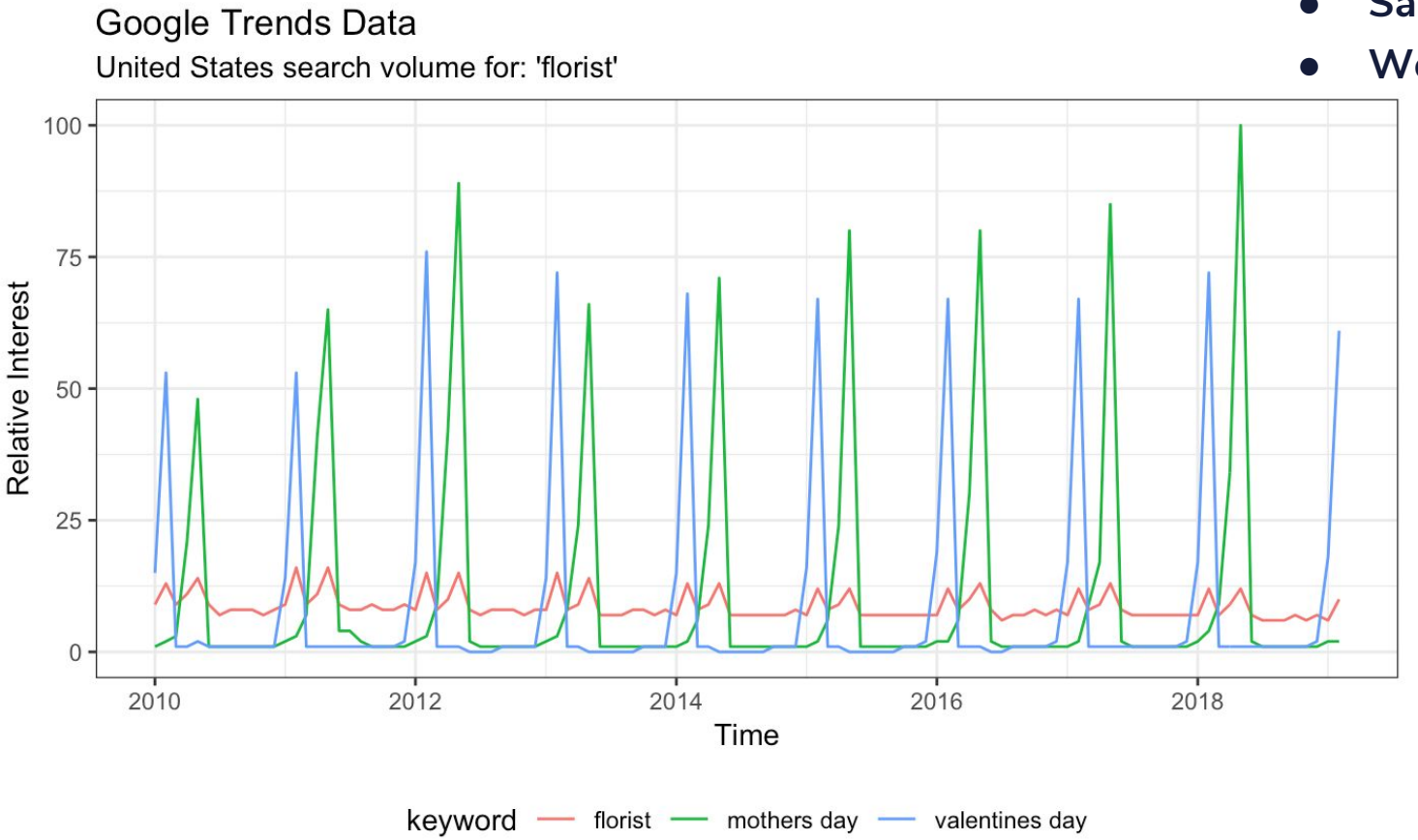
Seasonal trends: holidays and known events

Use cases:

- Sales + Promotions
- Workforce Planning



Seasonal trends: holidays and known events

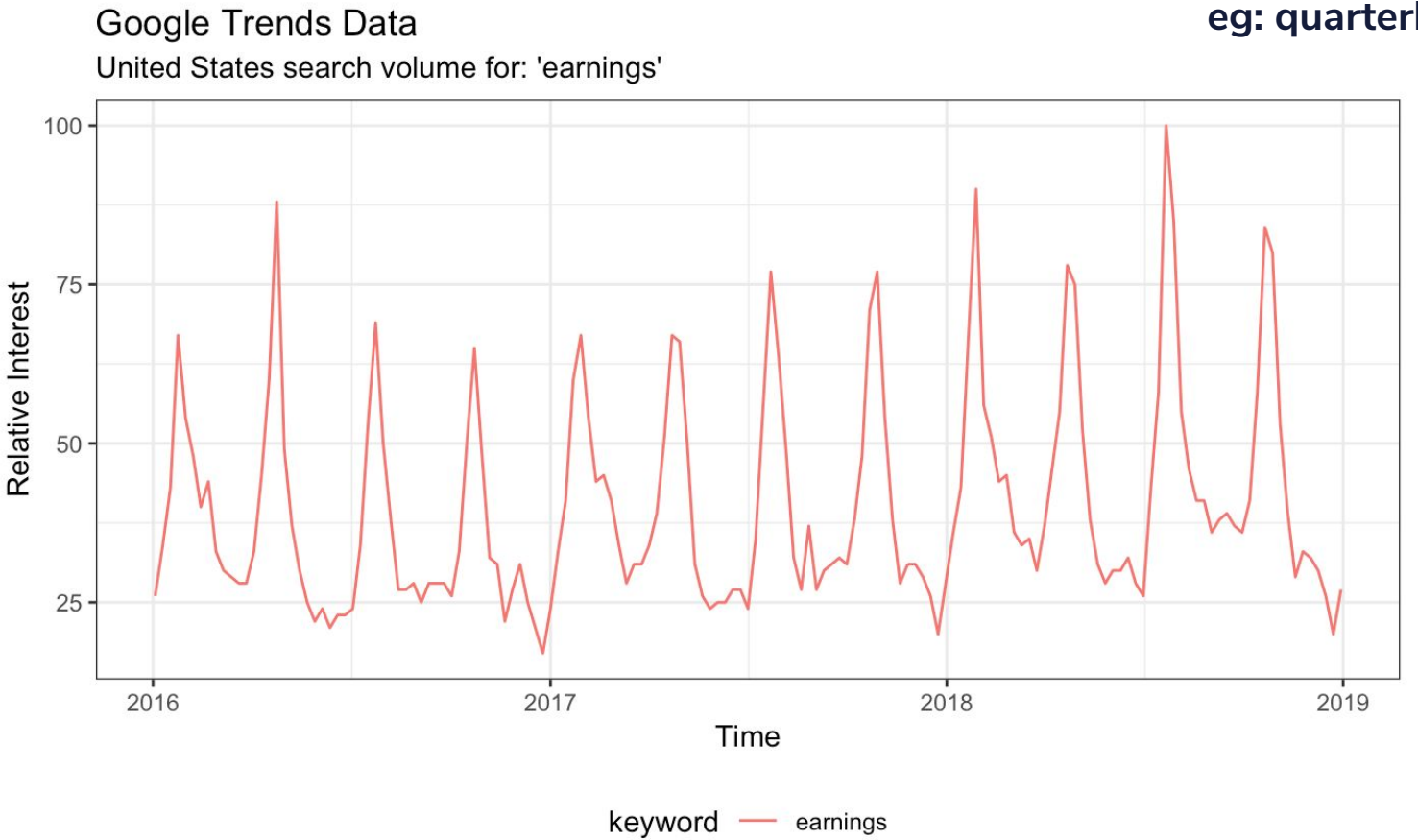


Use cases:

- **Sales + Promotions**
- **Workforce Planning**

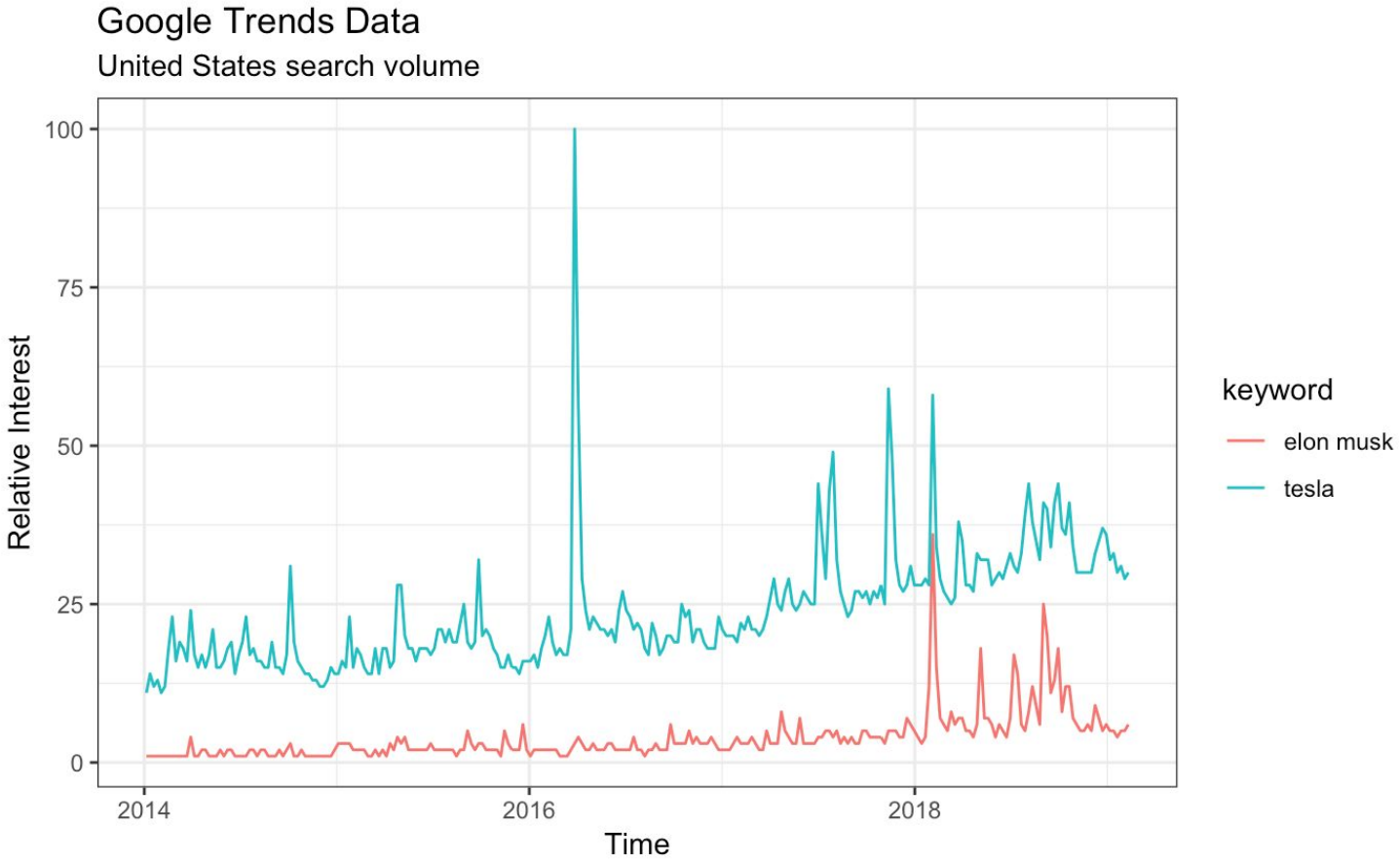
Enough historical data and domain knowledge

eg: quarterly earnings calls

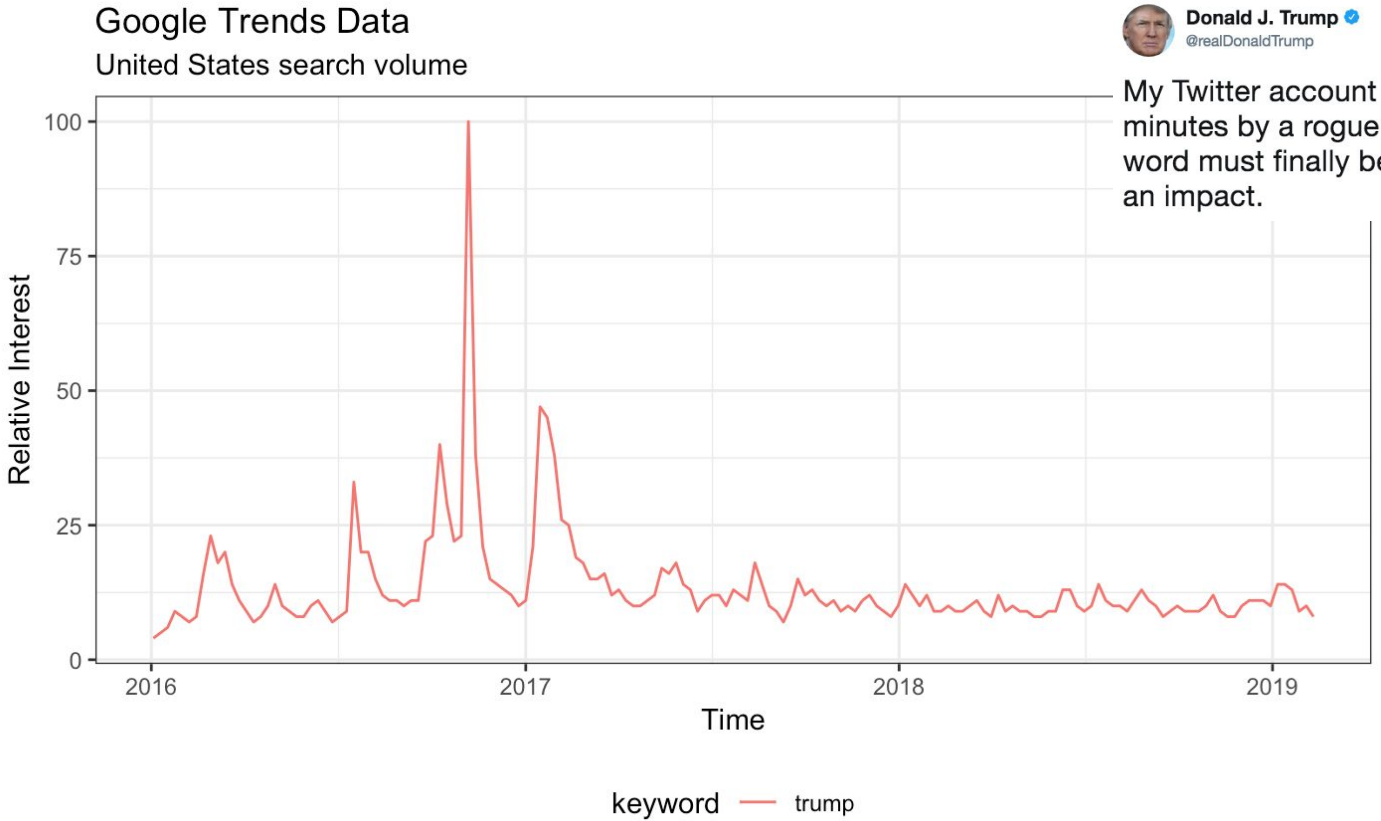


When shouldn’t we forecast our time series data?

External factors influence your primary metric.



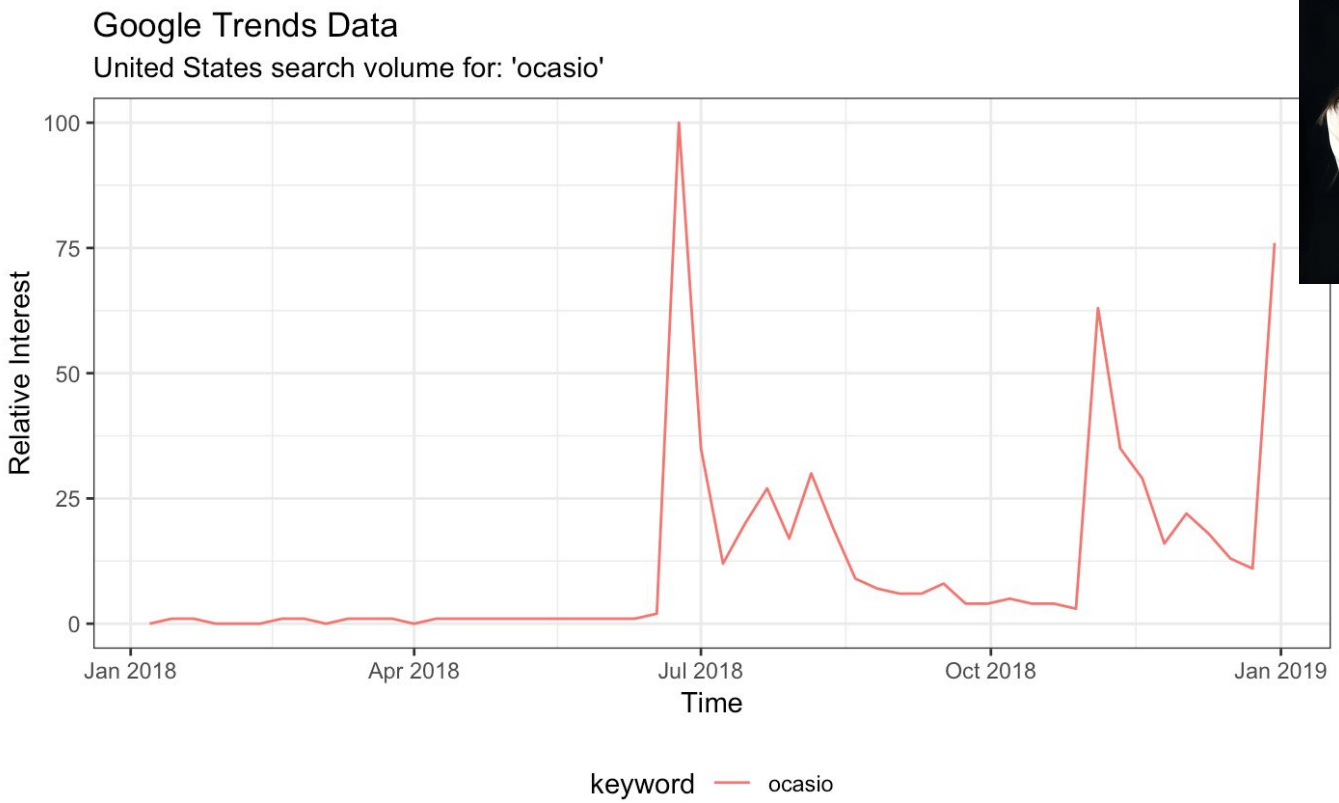
Behavior is too unpredictable to forecast.



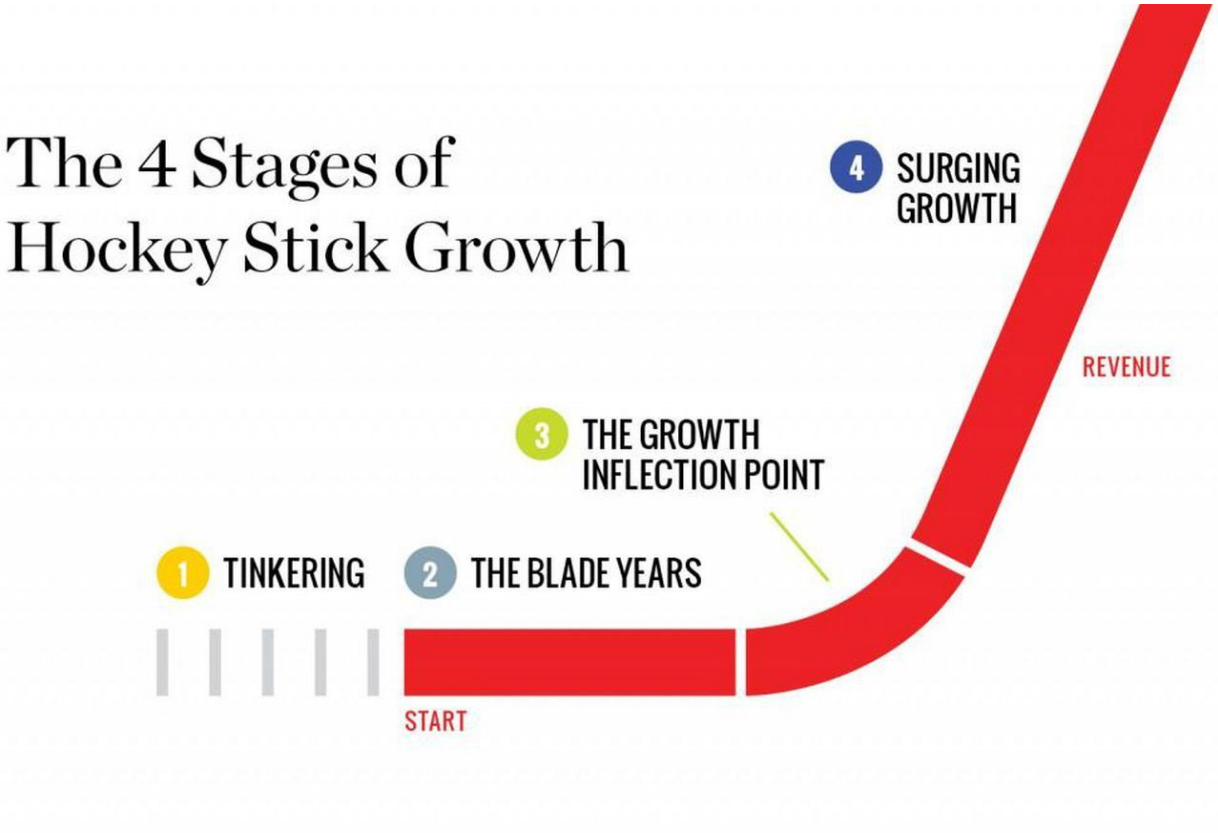
Donald J. Trump
@realDonaldTrump

My Twitter account was taken down for 11 minutes by a rogue employee. I guess the word must finally be getting out-and having an impact.

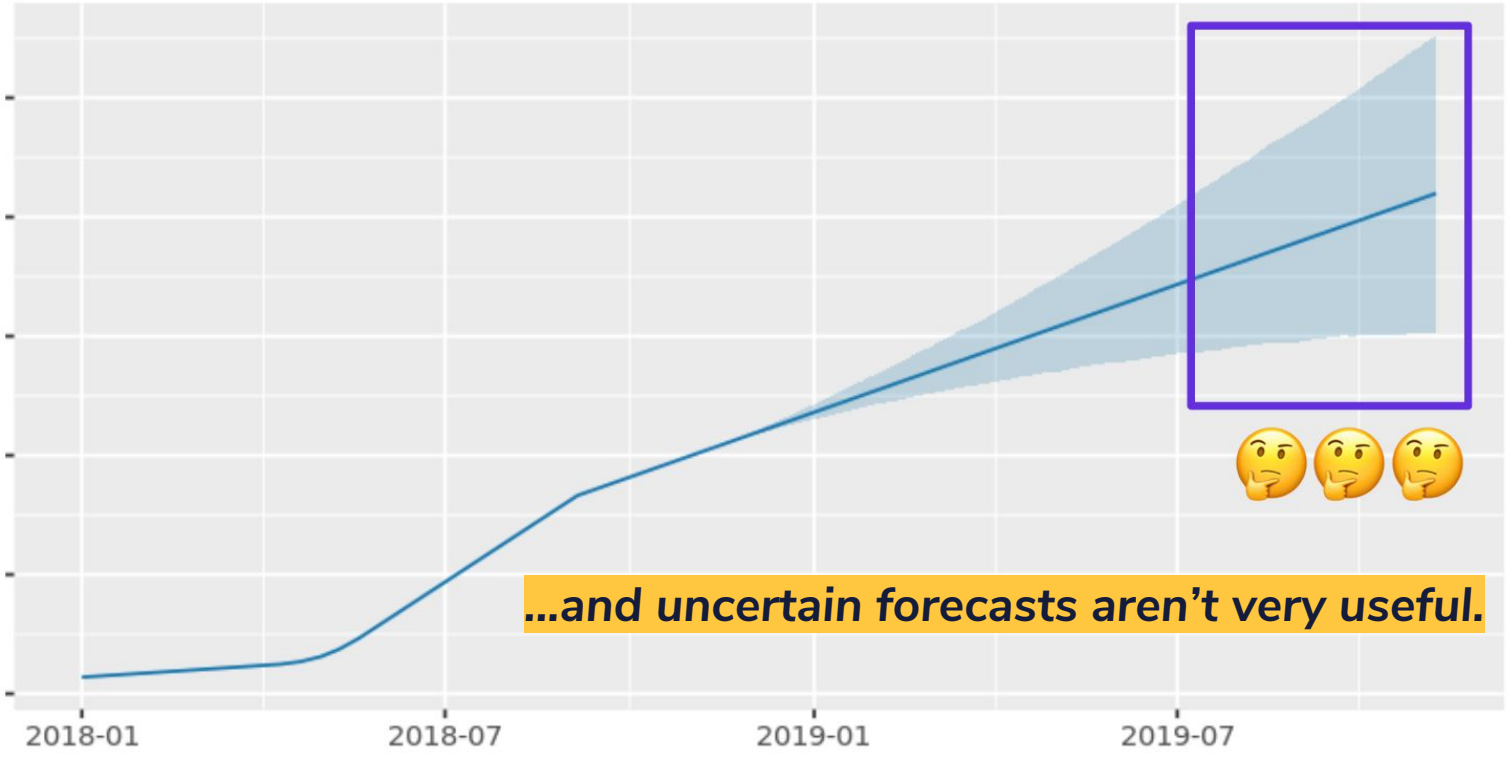
We only started collecting data recently (or a change of events) = not enough data to forecast.



For data scientists, growth is a double-edged sword.



Growth creates uncertainty in time series forecasting...



codecademy

LIVE CODE SESSION: PROPHET

Let's get started!

Follow along:

twitter @catherinezh

github @cattystats

https://github.com/cattystats/

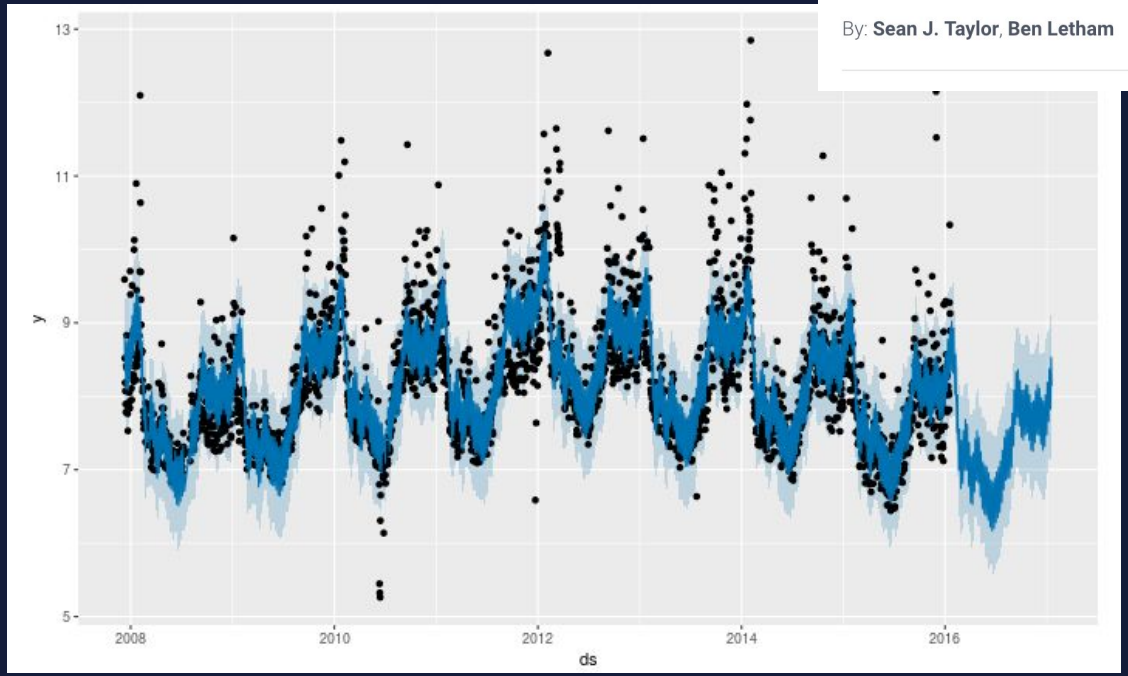
codecademy

facebook research

February 23, 2017

Prophet: forecasting at scale

By: **Sean J. Taylor**, **Ben Letham**



Where Prophet shines

Not all forecasting problems can be solved by the same procedure. Prophet is optimized for the business forecast tasks we have encountered at Facebook, which typically have any of the following characteristics:

- hourly, daily, or weekly observations with at least a few months (preferably a year) of history
- strong multiple “human-scale” seasonalities: day of week and time of year
- important holidays that occur at irregular intervals that are known in advance (e.g. the Super Bowl)
- a reasonable number of missing observations or large outliers
- historical trend changes, for instance due to product launches or logging changes
- trends that are non-linear growth curves, where a trend hits a natural limit or saturates

How Prophet works

At its core, the Prophet procedure is an additive regression model with four main components:

- A piecewise linear or logistic growth curve trend. Prophet automatically detects changes in trends by selecting changepoints from the data.
- A yearly seasonal component modeled using Fourier series.
- A weekly seasonal component using dummy variables.
- A user-provided list of important holidays.

$$y(t) = g(t) + s(t) + h(t) + \varepsilon t$$

y = forecast output

g = growth trend (directional trend)

s = seasonality (weekly, monthly, etc)

h = holidays (user-provided)


```r
#install.packages("gtrendsR")
library(gtrendsR)
google_trends_df = gtrends(
                    c("Vote"), #keywords -- start with one
                    gprop = "web", #choose: web, news, images, froogle, youtube
                    geo = c("US"), #only pull results for US
                    time = "2004-01-01 2018-11-08")[[1]] #timeframe
```

```
> as.tibble(google_trends_df)
# A tibble: 179 x 6
   date                hits keyword geo   gprop category
   <dttm>             <int> <chr>   <chr> <chr>     <int>
 1 2004-01-01 00:00:00     5 Vote    US    web           0
 2 2004-02-01 00:00:00     7 Vote    US    web           0
 3 2004-03-01 00:00:00     7 Vote    US    web           0
 4 2004-04-01 00:00:00     5 Vote    US    web           0
 5 2004-05-01 00:00:00     5 Vote    US    web           0
 6 2004-06-01 00:00:00     5 Vote    US    web           0
 7 2004-07-01 00:00:00    10 Vote    US    web           0
 8 2004-08-01 00:00:00    14 Vote    US    web           0
 9 2004-09-01 00:00:00    21 Vote    US    web           0
10 2004-10-01 00:00:00    46 Vote    US    web           0
# ... with 169 more rows
>
```

codecademy

1. LOAD PROPHET + PREPARE DATA

```
df <- google_trends_df %>%
      mutate(date=lubridate::ymd(date)) %>% #parse date
      tbl_df() %>%
      mutate(ds=date,y=hits) %>%
      select(ds,y) #format for prophet

library(prophet) #load package
```

codecademy

2. SPECIFY CEILING AND FLOOR USING DOMAIN KNOWLEDGE

```
#specify ceiling and floor using domain knowledge
df$cap <- 100 #google trend hits is a normalized value
df$floor <- 0 #ranges from 0-100
m <- prophet(df,growth = 'logistic')
```

codecademy

3. MAKE FUTURE DATA FRAME. NOW FORECAST!

```
#specify ceiling and floor using domain knowledge
df$cap <- 100 #google trend hits is a normalized value
df$floor <- 0 #ranges from 0-100
m <- prophet(df,growth = 'logistic')

future <- make_future_dataframe(m, periods = 52,freq = 'week')
future$cap <- 100
future$floor <- 0

forecast <- predict(m, future)
```

codecademy

FORECASTING WEEKLY GOOGLE SEARCHES FOR ‘EARNINGS’

FORECAST OUTPUT

```
forecast <- predict(m, future)
forecast %>%
  select(ds,yhat,yhat_lower,yhat_upper)%>%
  rename("week"="ds",
         "forecast"="yhat",
         "forecast_lower"="yhat_lower",
         "forecast_upper"="yhat_upper")%>%
  filter(week>="2019-03-25")
```

Predicts Increased Search Volume for April 2019 Q1 Earnings Call Season

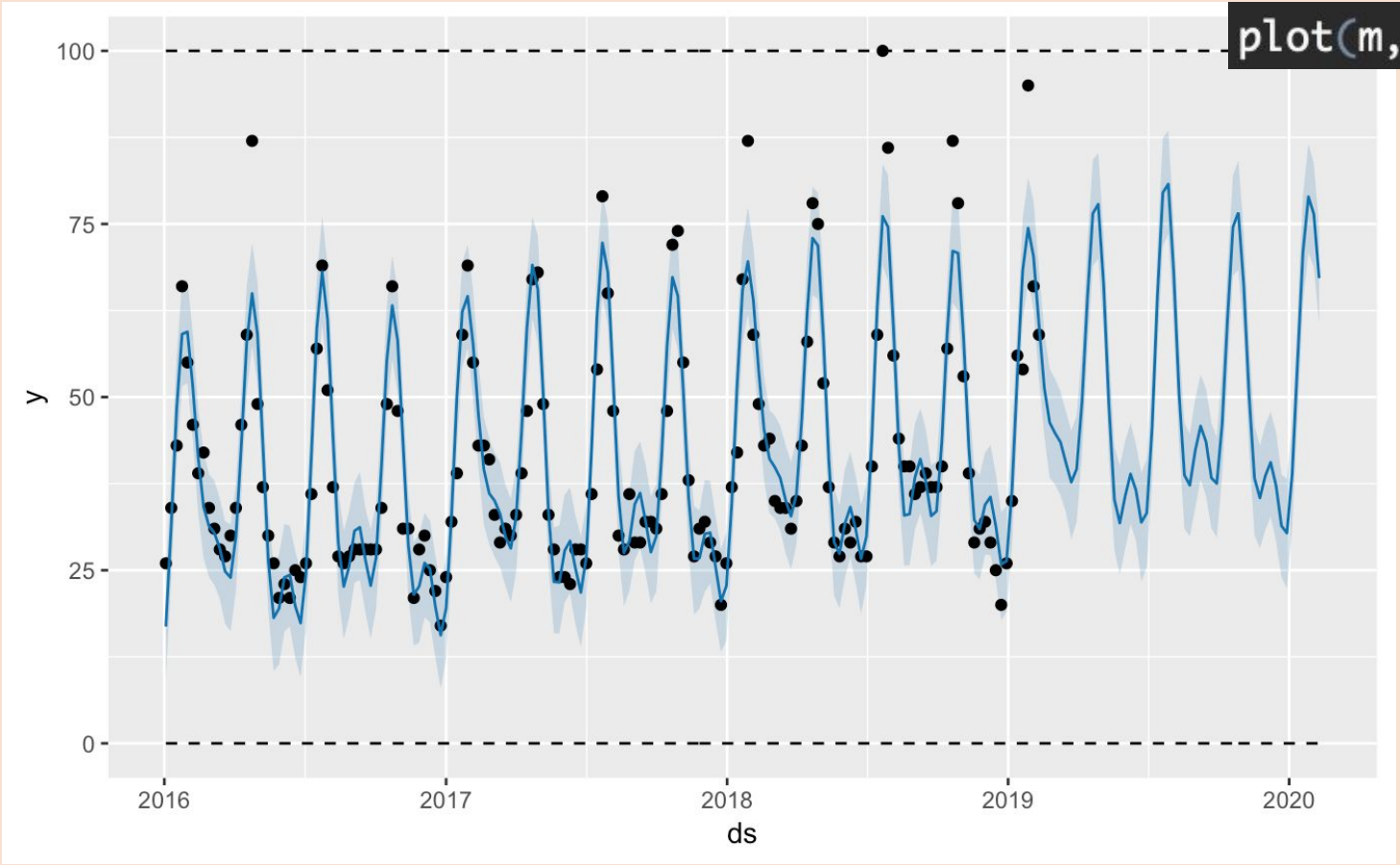
week <S3: POSIXct>	forecast <dbl>	forecast_lower <dbl>	forecast_upper <dbl>
2019-03-31	41.32850	33.57433	49.07475
2019-04-07	51.08769	43.87153	58.85880
2019-04-14	65.89501	58.54957	73.64711
2019-04-21	78.04951	70.47922	86.16771
2019-04-28	79.49348	71.68262	87.01388
2019-05-05	68.19601	60.62778	76.30620
2019-05-12	50.53473	42.87641	58.21047
2019-05-19	36.81497	28.86804	44.24385

codecademy

FORECASTING WEEKLY GOOGLE SEARCHES FOR ‘EARNINGS’

PLOT FORECAST

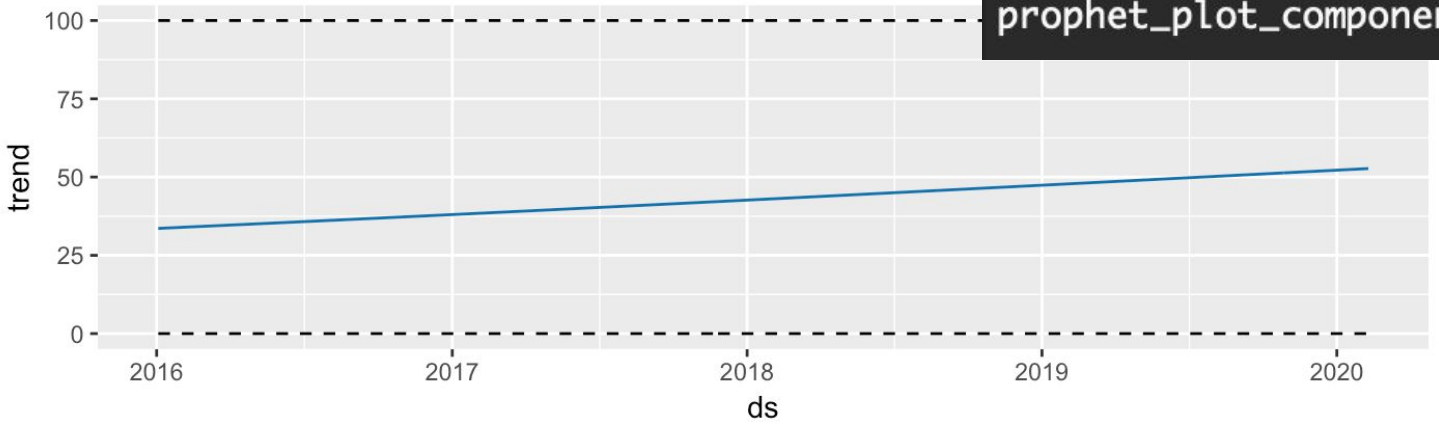
```
plot(m, forecast)
```

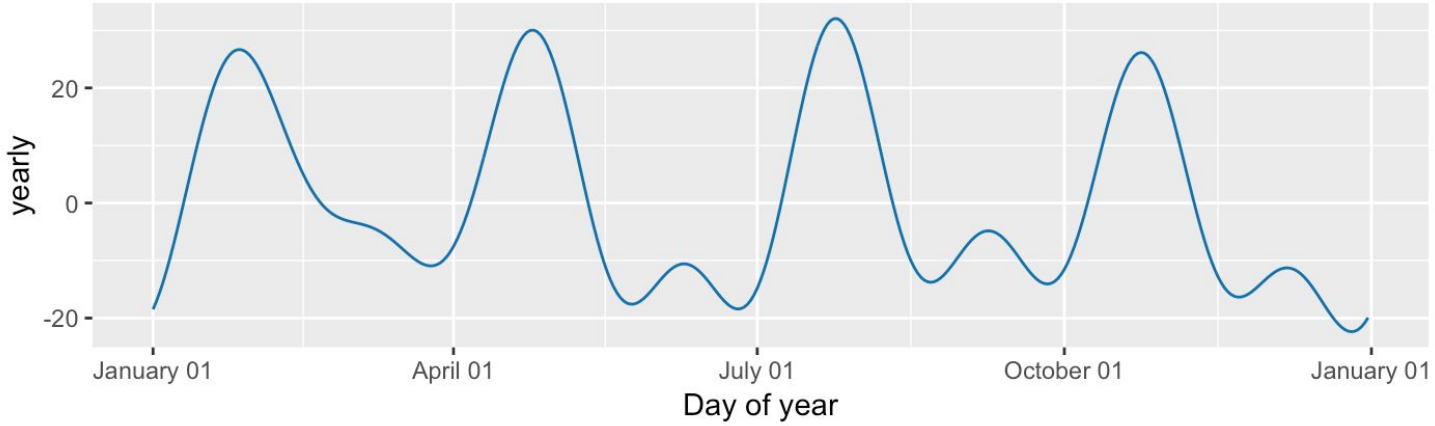


FORECASTING WEEKLY GOOGLE SEARCHES FOR ‘EARNINGS’

FORECAST COMPONENTS

```
prophet_plot_components(m, forecast)
```





INPUT RELEVANT HOLIDAYS, COMPANY EARNINGS CALLS, PRODUCT RELEASES

Past Events

Jan 30, 2019 - 2:00 PM PT

Facebook Q4 2018 Earnings

Earnings Release (pdf) - Slides (pdf) - Earnings Call Transcript (pdf) - Follow Up Call Tran

Oct 30, 2018 - 2:00 PM PT

Facebook Q3 2018 Earnings

2019 Holidays

The most common (Federal) holidays of the United States (USA) in 2019 are listed

Date	Holiday
January 1	New Year's Day 2019
January 21	Martin Luther King Day 2019
January 24	Belly Laugh Day 2019
February 2	Groundhog Day 2019
February 12	Lincoln's Birthday 2019
February 14	Valentine's Day 2019

```
holiday_post <- data_frame(
  holiday = 'holiday_post',
  ds = as.Date(c('2017-01-01', '2018-01-01', '2019-01-01',
                 '2017-11-23', '2018-11-22', '2019-11-28',
                 '2017-12-25', '2018-12-25', '2019-12-25',
                 '2017-12-31', '2018-12-31', '2019-12-31' )),
  lower_window = 0,
  upper_window = 1
)
holiday_pre <- data_frame(
  holiday = 'holiday_pre',
  ds = as.Date(c('2017-07-04', '2018-07-04', '2019-07-04',
                 '2017-09-04', '2018-09-03', '2019-09-02',
                 '2017-12-25', '2018-12-25', '2019-12-25')),
  lower_window = -1,
  upper_window = 0
)
holidays <- bind_rows(holiday_pre, holiday_post)

m <- prophet(df,holidays = holidays,growth = 'logistic')
```

INPUT RELEVANT HOLIDAYS, COMPANY EARNINGS CALLS, PRODUCT RELEASES

Past Events

Jan 30, 2019 - 2:00 PM PT

Facebook Q4 2018 Earnings

Earnings Release (pdf) - Slides (pdf) - Earnings Cal

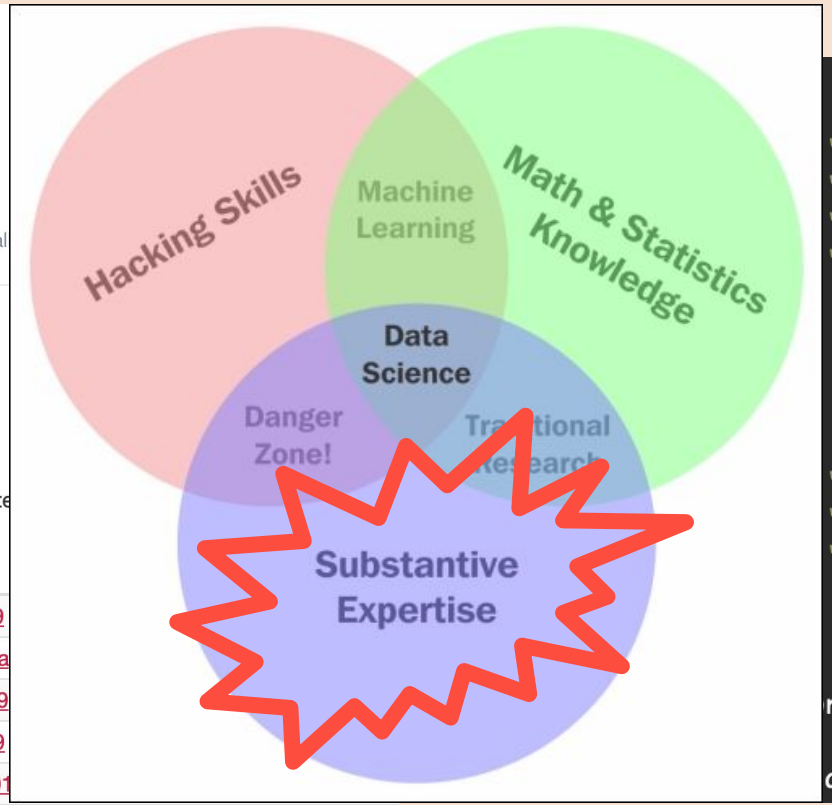
Oct 30, 2018 - 2:00 PM PT

Facebook Q3 2018 Earnings

2019 Holidays

The most common (Federal) holidays of the Unite

Date	Holiday
January 1	New Year's Day 2019
January 21	Martin Luther King Da
January 24	Belly Laugh Day 2019
February 2	Groundhog Day 2019
February 12	Lincoln's Birthday 201
February 14	Valentine's Day 2019




```
'2018-01-01', '2019-01-01',
'2018-11-22', '2019-11-28',
'2018-12-25', '2019-12-25',
'2018-12-31', '2019-12-31' )),

'2018-07-04', '2019-07-04',
'2018-09-03', '2019-09-02',
'2018-12-25', '2019-12-25')),

re, holiday_post)

days,growth = 'logistic')
```

Forecasting Best Practices_

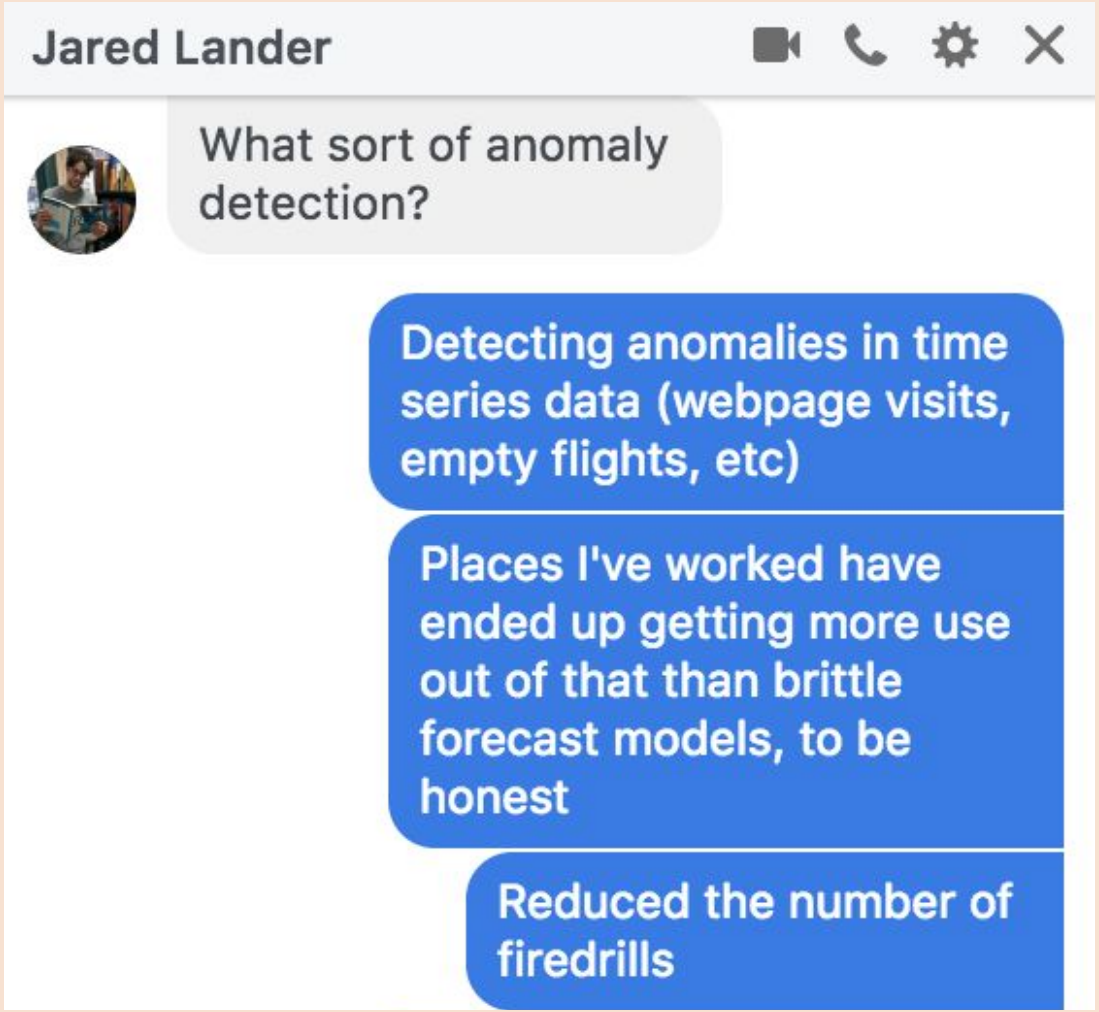
- **Set clear expectations**
  - What level of certainty can a forecast provide given the data at hand?
- **Share upper and lower ranges**
  - Sharing just the midpoint might lead to speculation if the upper and lower ranges are broad.
- **Update your priors (capacity, floor, holidays/launches, etc)**
- **Differentiate company forecasts vs targets**
- **Domain knowledge is valuable!** *(Use your 'spreadsheet people')*

codecademy

PART THREE

Anomaly Detection_

codecademy

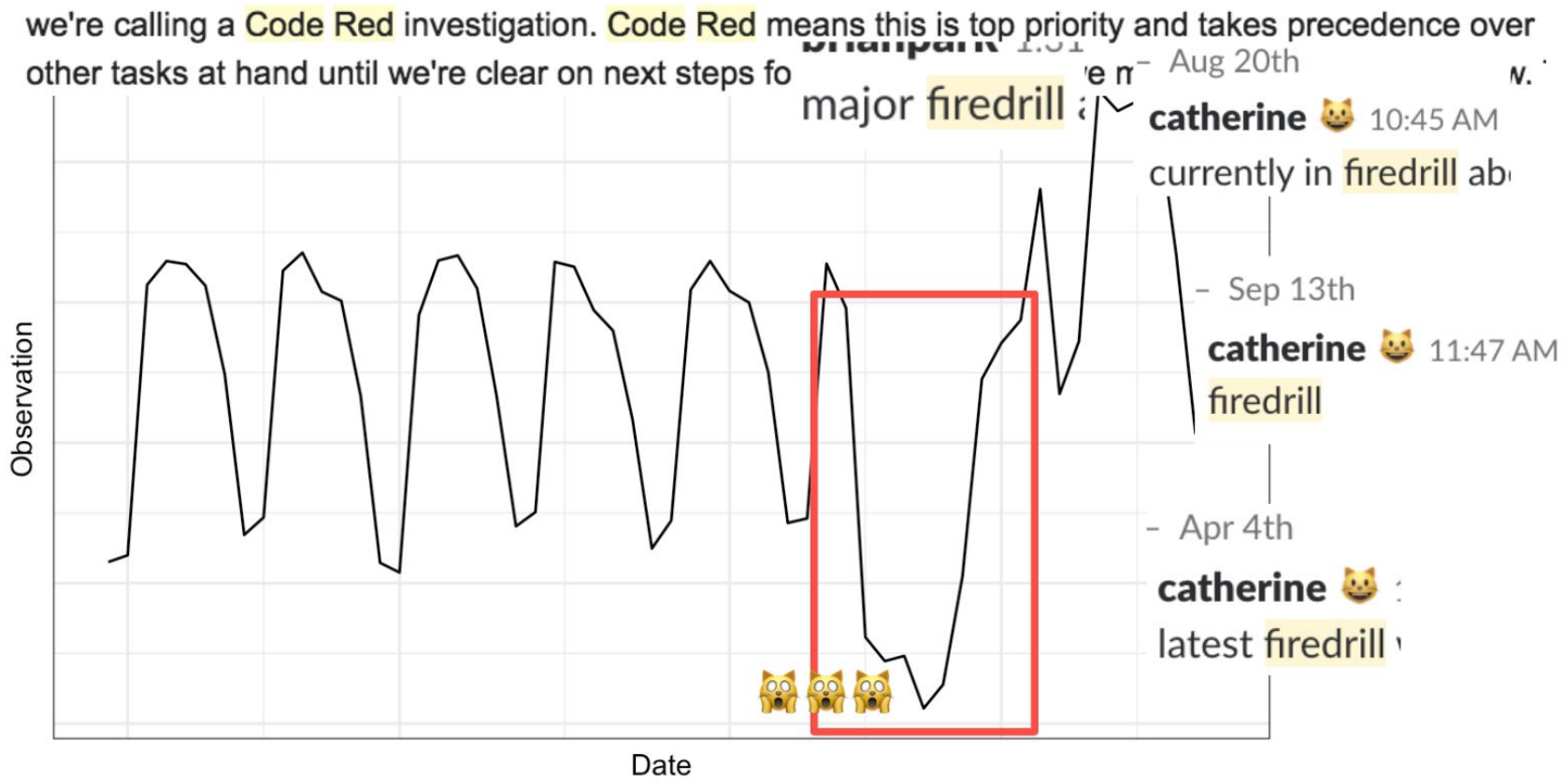


Anomaly detection on key metrics can lead to earlier detection of irregularities and reduce the number of fire drills.

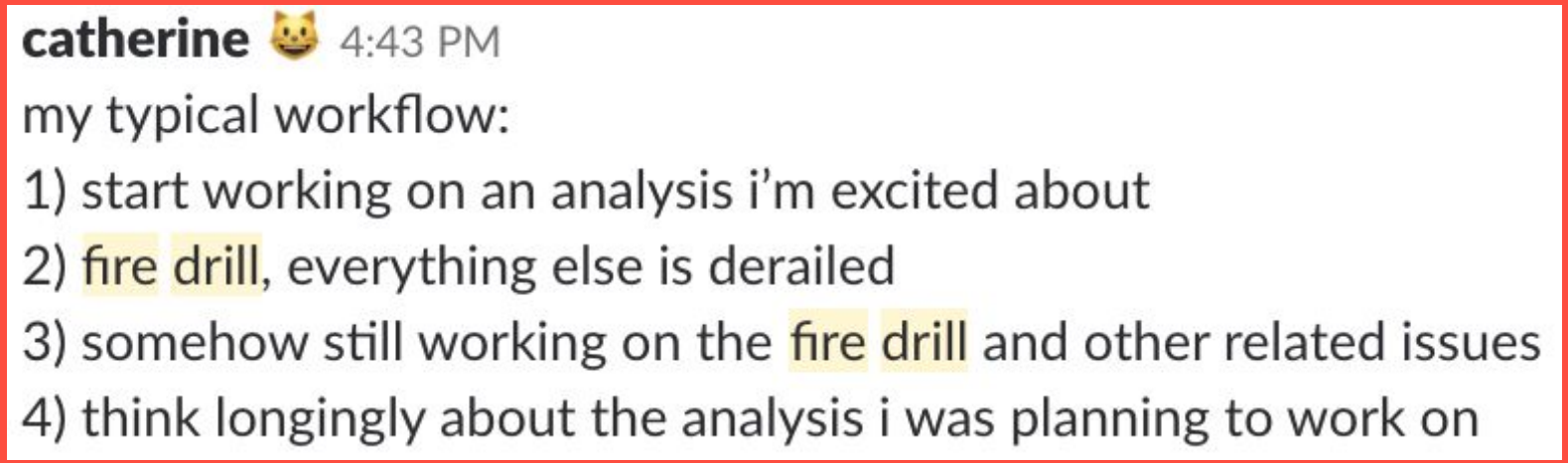
We can be proactive instead of reactive.

codecademy

something is terribly wrong 🔥🔥🔥



DATA SCIENCE FIRE DRILLS



PART THREE

Anomaly Detection with Time Series

... or how to know when something is terribly wrong 🔥🔥🔥

Applications of Anomaly Detection_

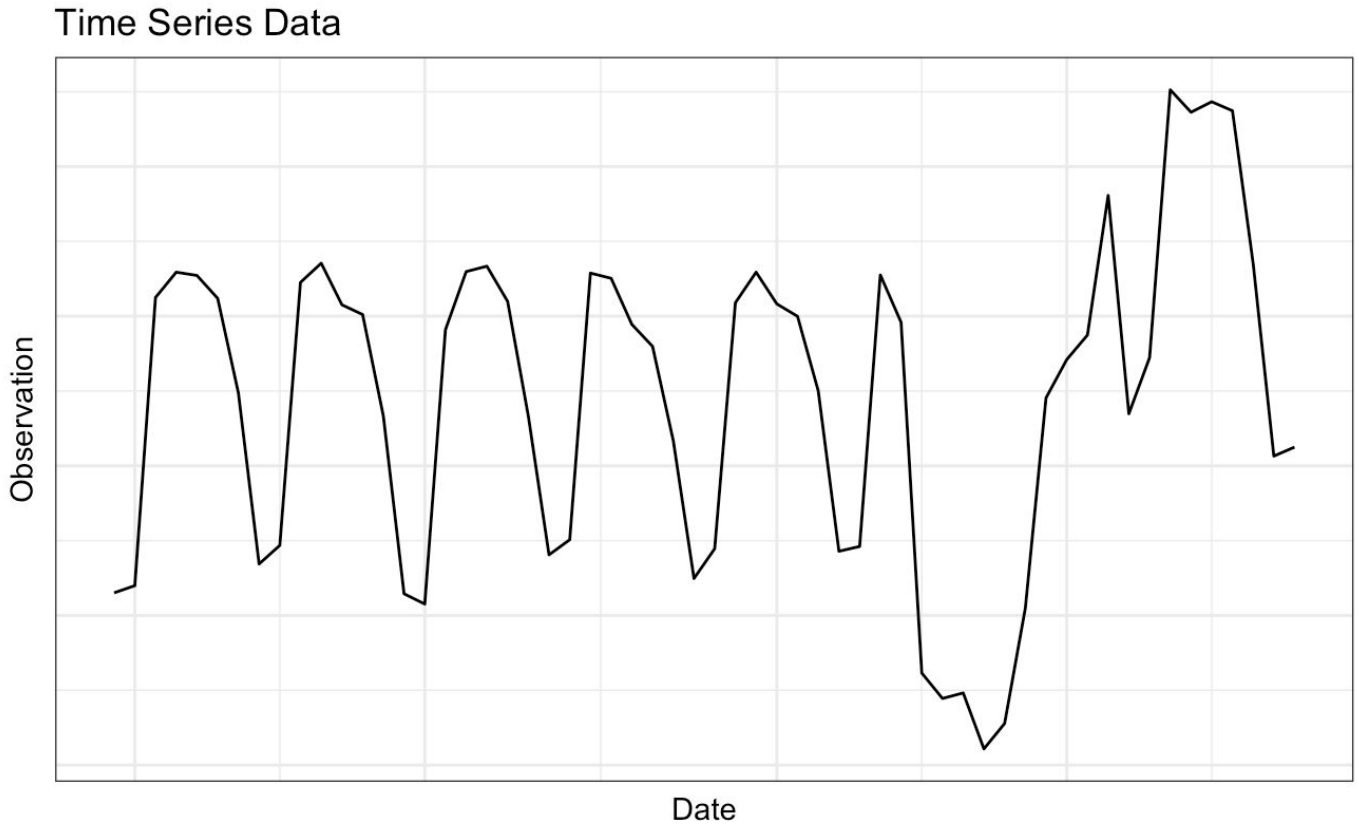
- Fraud Detection
- KPI Monitoring
- Identify Breakage
- Workforce Planning
- Nature (e.g. weather)
- ... and more!

"monitor key metrics, website breakage, and fraudulent activity... we can build a system for anomaly detection to uncover blind spots in large datasets and reduce fire drills at work"

- me talking about why anomaly detection matters

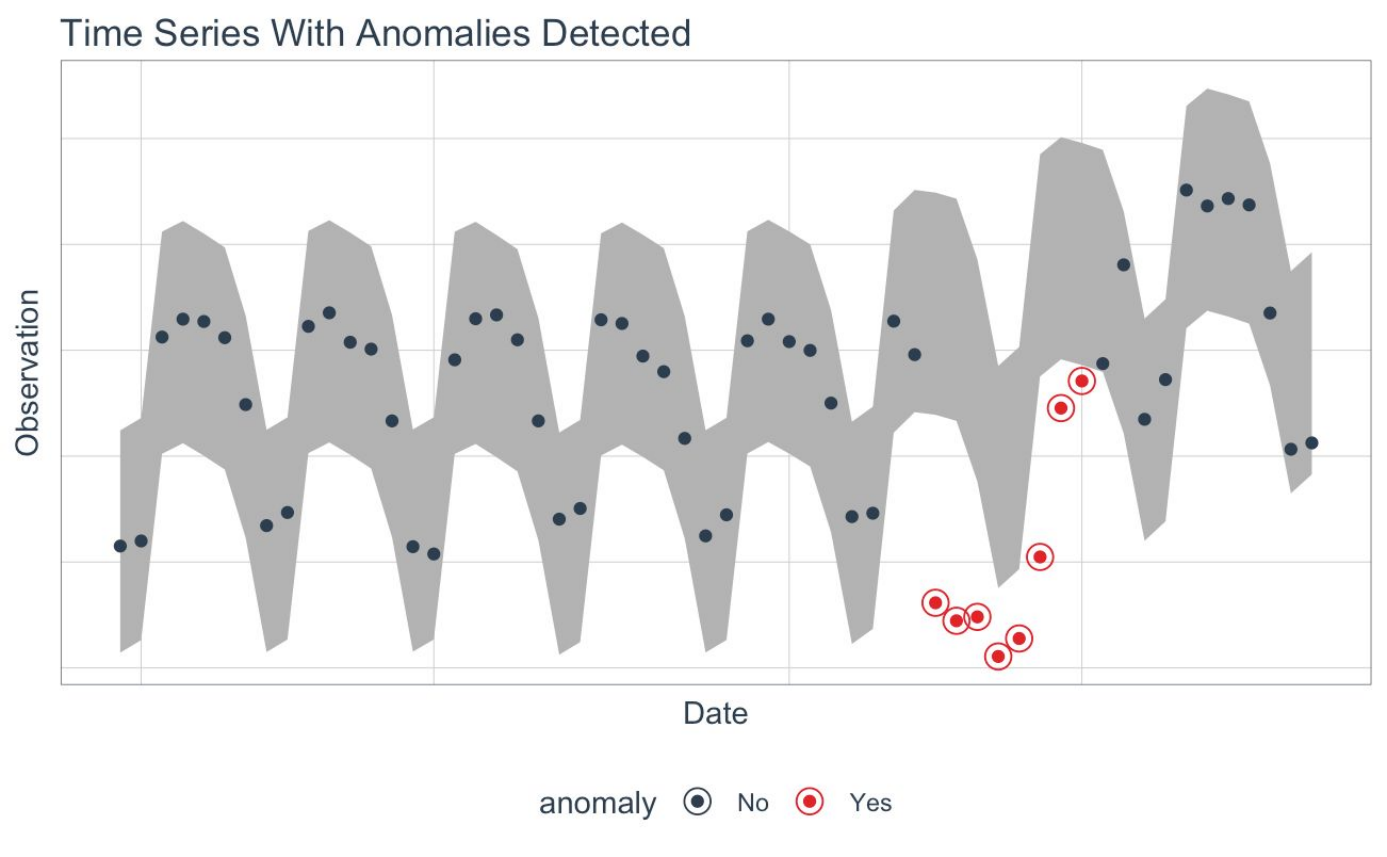
codecademy

Before...



codecademy

AFTER!



codecademy

LIVE CODE SESSION: ANOMALIZE

Let's get started!

Follow along:

twitter @catherinezh

github @cattystats

bit.ly/anomaly-r

codecademy

1. CREATE A DATA FRAME


```
#install.packages("gtrendsR")
library(gtrendsR)
google_trends_df = gtrends(
                    c("Vote"), #keywords -- start with one
                    gprop = "web", #choose: web, news, images, froogle, youtube
                    geo = c("US"), #only pull results for US
                    time = "2004-01-01 2018-11-08")[[1]] #timeframe
```

```
> as.tibble(google_trends_df)
# A tibble: 179 x 6
   date                hits keyword geo   gprop category
   <dttm>             <int> <chr>   <chr> <chr>     <int>
 1 2004-01-01 00:00:00     5 Vote    US    web           0
 2 2004-02-01 00:00:00     7 Vote    US    web           0
 3 2004-03-01 00:00:00     7 Vote    US    web           0
 4 2004-04-01 00:00:00     5 Vote    US    web           0
 5 2004-05-01 00:00:00     5 Vote    US    web           0
 6 2004-06-01 00:00:00     5 Vote    US    web           0
 7 2004-07-01 00:00:00    10 Vote    US    web           0
 8 2004-08-01 00:00:00    14 Vote    US    web           0
 9 2004-09-01 00:00:00    21 Vote    US    web           0
10 2004-10-01 00:00:00    46 Vote    US    web           0
# ... with 169 more rows
>
```

install + load gtrendsR: choose a keyword that interests you

codecademy

2. PREPARE DATA

install + load tidyverse and anomalize

```
#install.packages("anomalize")
library(tidyverse)
library(anomalize)

google_trends_df_tbl = google_trends_df %>%
                       mutate(date=lubridate::ymd(date)) %>%
                       tbl_df()
```

codecademy

2. PREPARE DATA

install + load tidyverse and anomalize

```
#install.packages("anomalize")
library(tidyverse)
library(anomalize)

google_trends_df_tbl = google_trends_df %>%
                       mutate(date=lubridate::ymd(date)) %>%
                       tbl_df()
```

codecademy

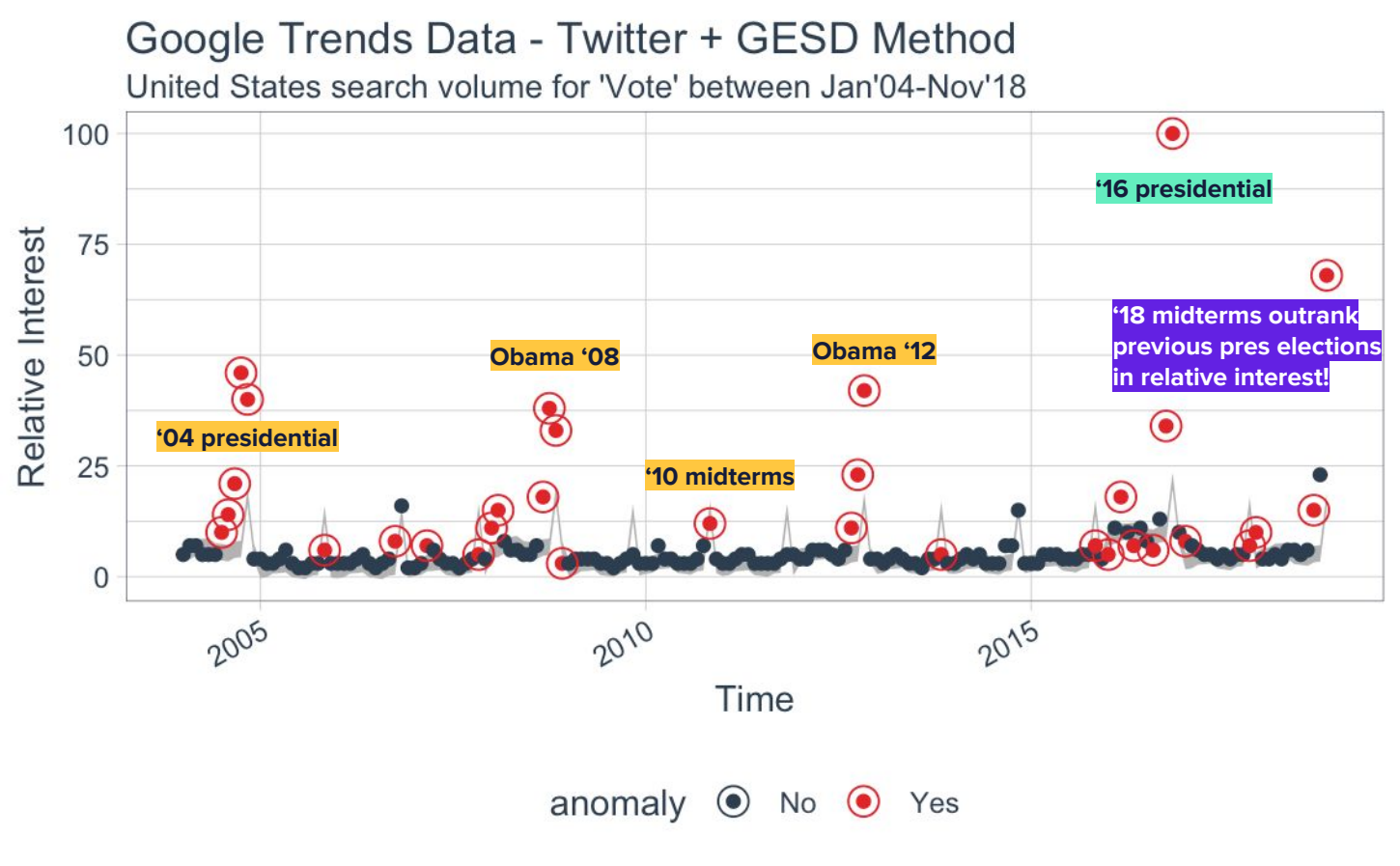
3. ANOMALIZE!

anomalize!

```
google_trends_df_tbl %>%   # Twitter and GESD
    time_decompose(hits, method = "twitter",trend = "1 year") %>%
    anomalize(remainder, method = "gesd") %>%
    time_recompose() %>%
    # Anomaly Visualization
    plot_anomalies(time_recomposed = TRUE) +
    labs(title = "Google Trends Data - Twitter + GESD
Method",x="Time",y="Relative Interest", subtitle = "United States search volume
for 'Vote' between Jan'04-Nov'18"
         )
```

codecademy

3. ANOMALIZE... TADA! KEYWORD: VOTE

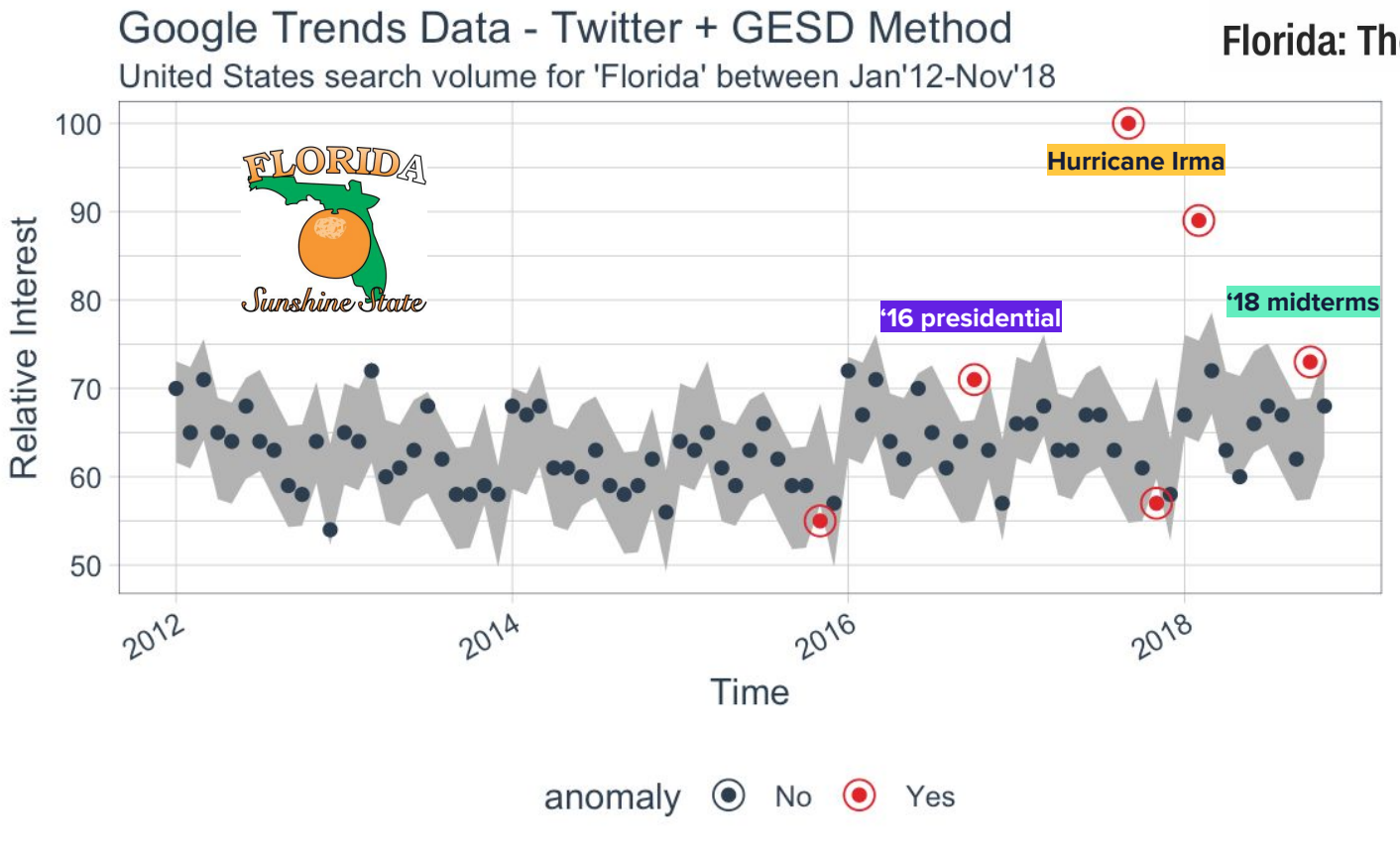


codecademy

LET'S TRY THIS WITH... **KEYWORD: FLORIDA**

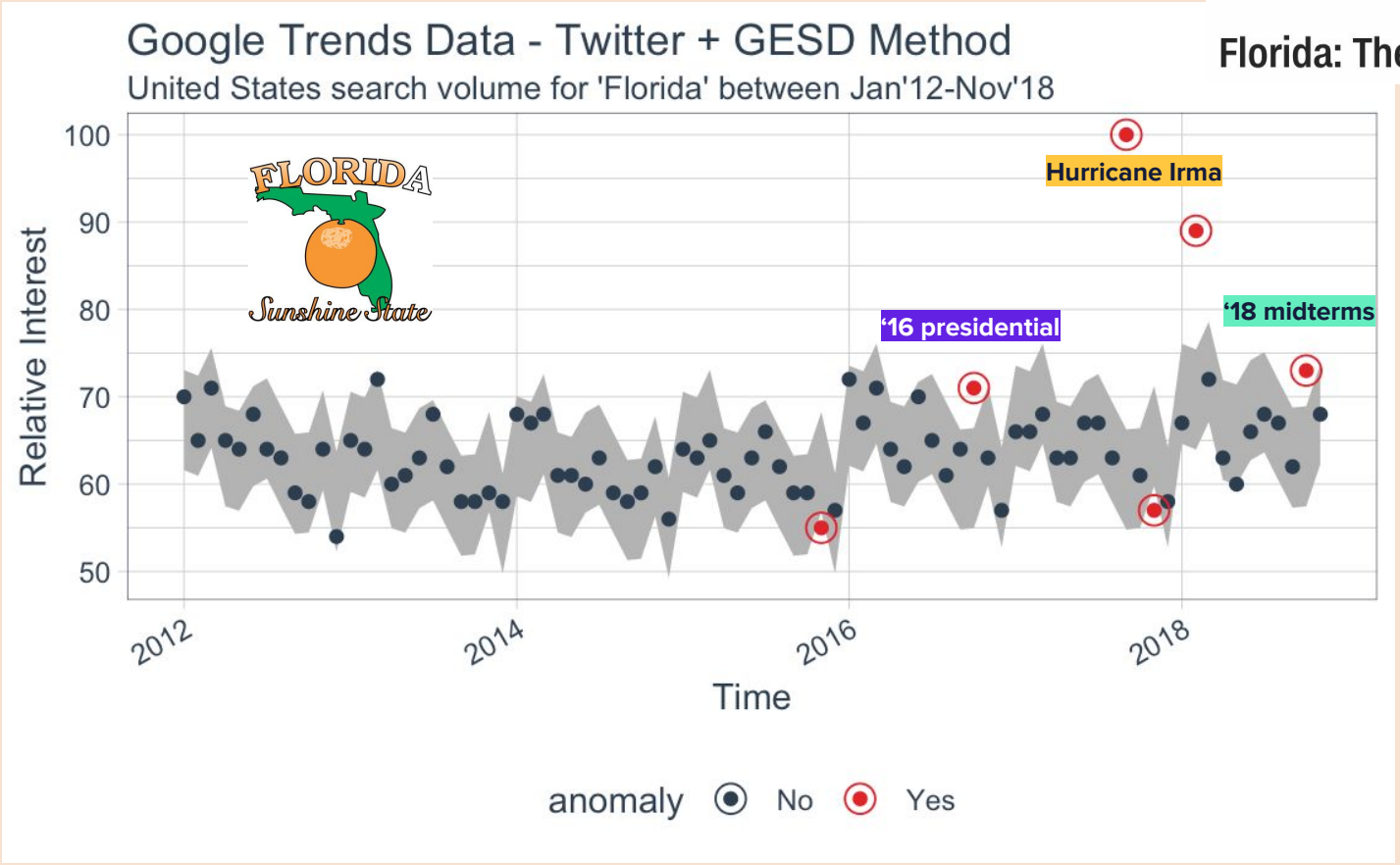
CNN politics

Florida: The swingiest swing state



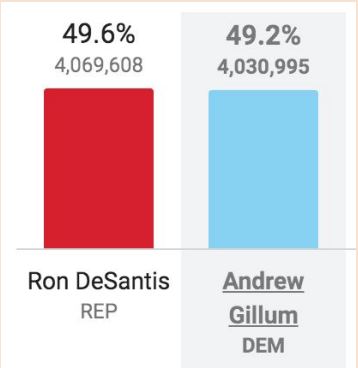
codecademy

LET'S TRY THIS WITH... **KEYWORD: FLORIDA**



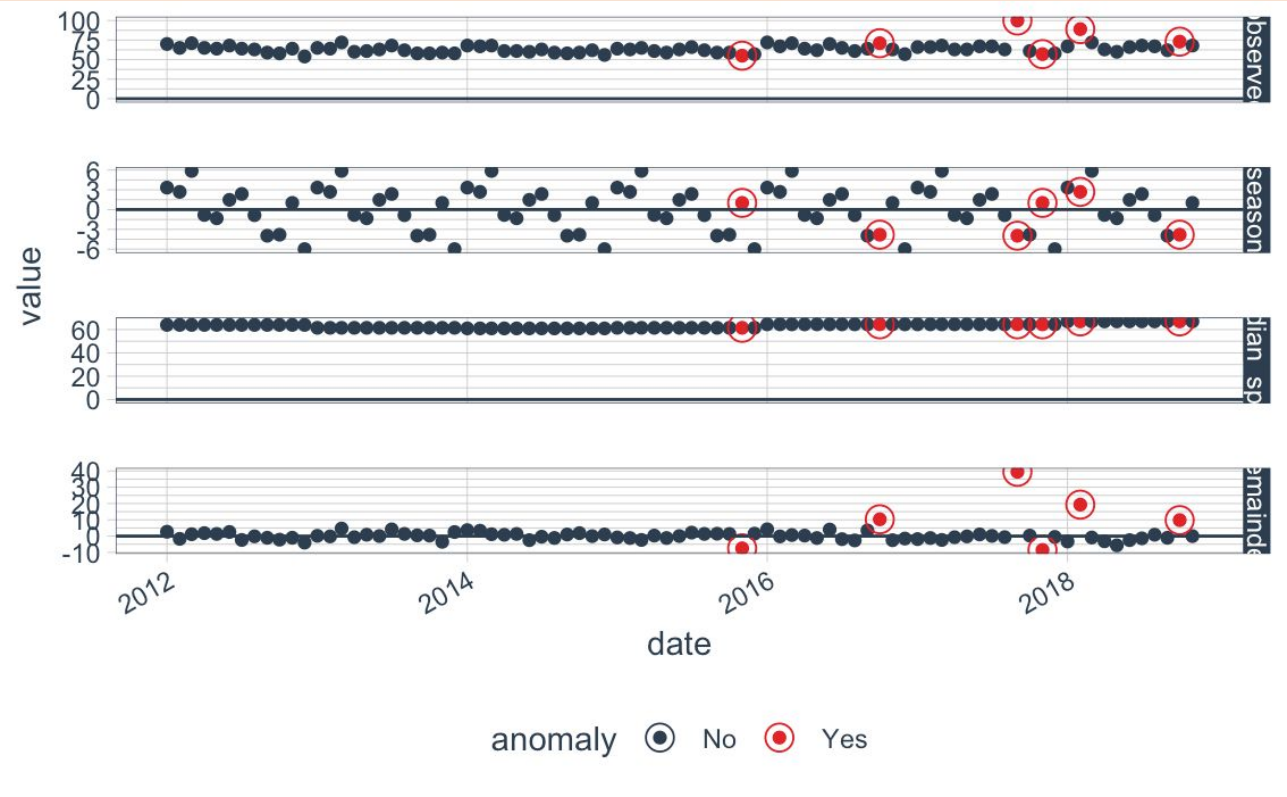
CNN politics

Florida: The swingiest swing state



KEYWORD: FLORIDA

```
google_trends_df_tbl %>%
  time_decompose(hits, method = "twitter",
                 frequency = "1 year", trend = "1 year") %>%
  anomalize(remainder, method = "gesd", alpha = 0.05, max_anoms = 0.2) %>%
  plot_anomaly_decomposition()
```



plot_anomaly_decomposition()

visualize inner workings of how algorithm detects anomalies in the "remainder"

codecademy

4. EXPLORE METHODS BASED ON TIME SERIES ATTRIBUTES

anomalize cheat sheet:

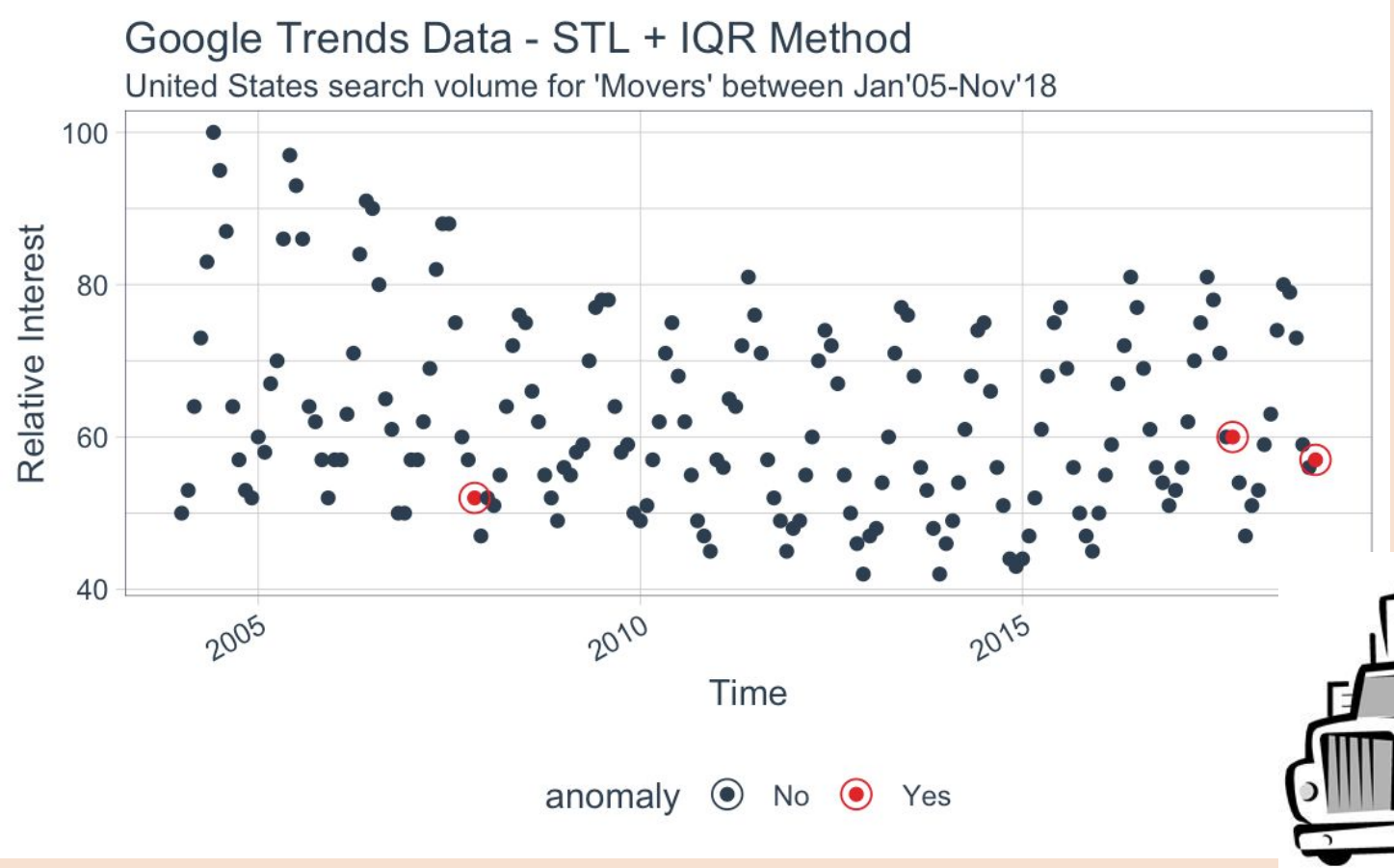
Twitter + GESD better for highly seasonal data

STL + IQR if seasonality is not a major factor

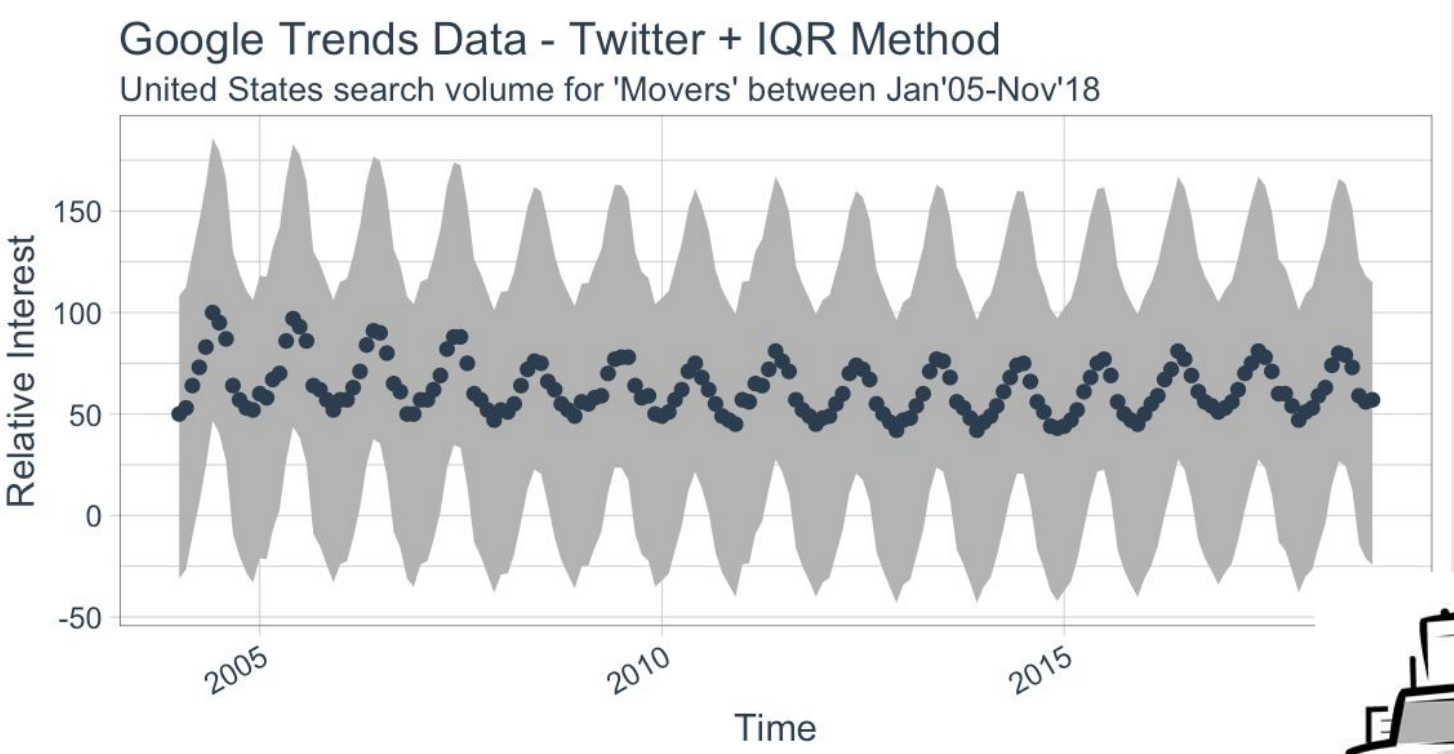
adjust trend period using domain knowledge

STL + IQR

KEYWORD: MOVERS

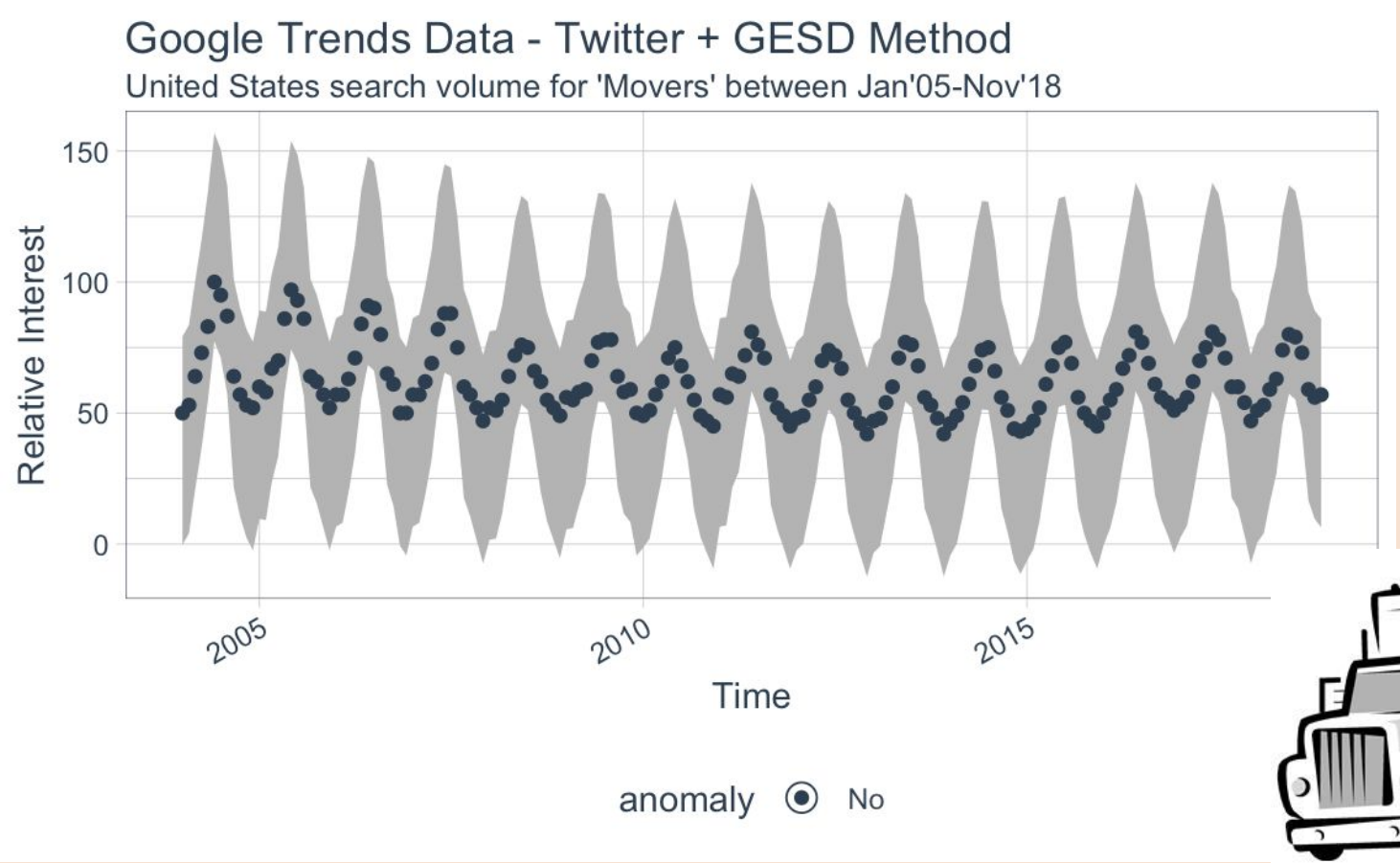


TWITTER DECOMPOSE CONTROLS FOR SEASONALITY **KEYWORD: MOVERS**



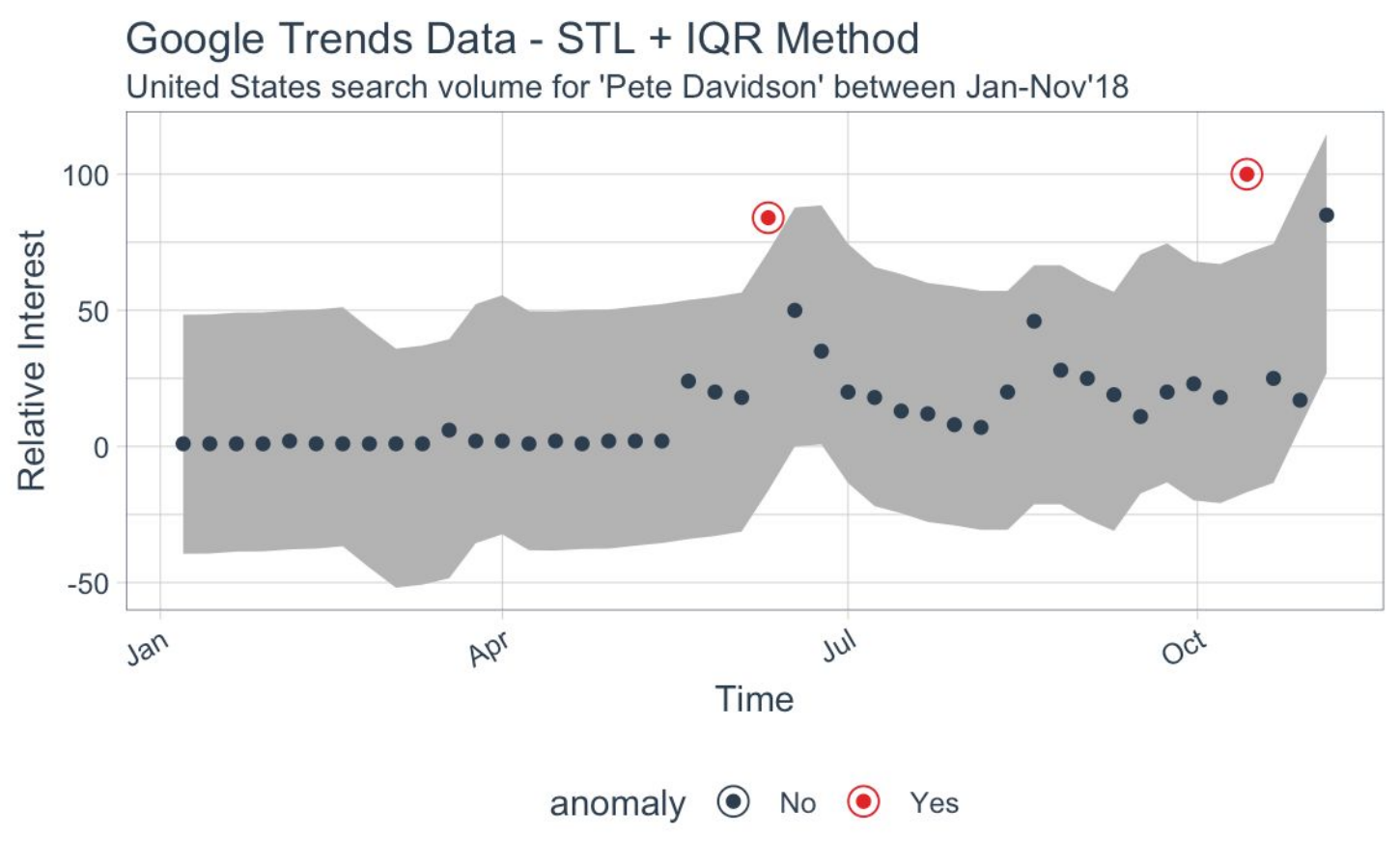
TWITTER + GESD PULLS IN THE MEDIAN/MEAN

KEYWORD: MOVERS



TRY THIS ON DIFFERENT KEYWORDS

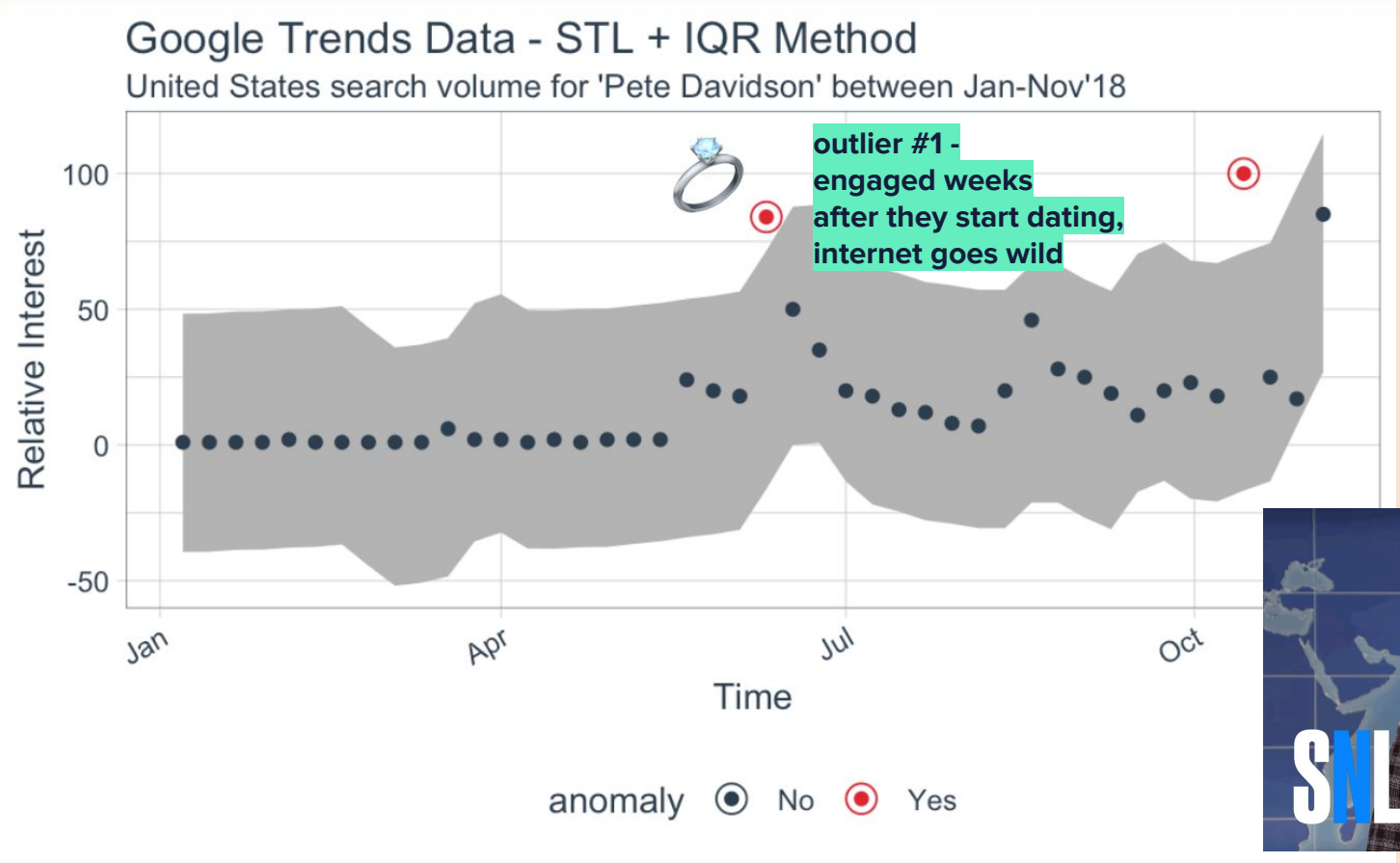
PETE DAVIDSON



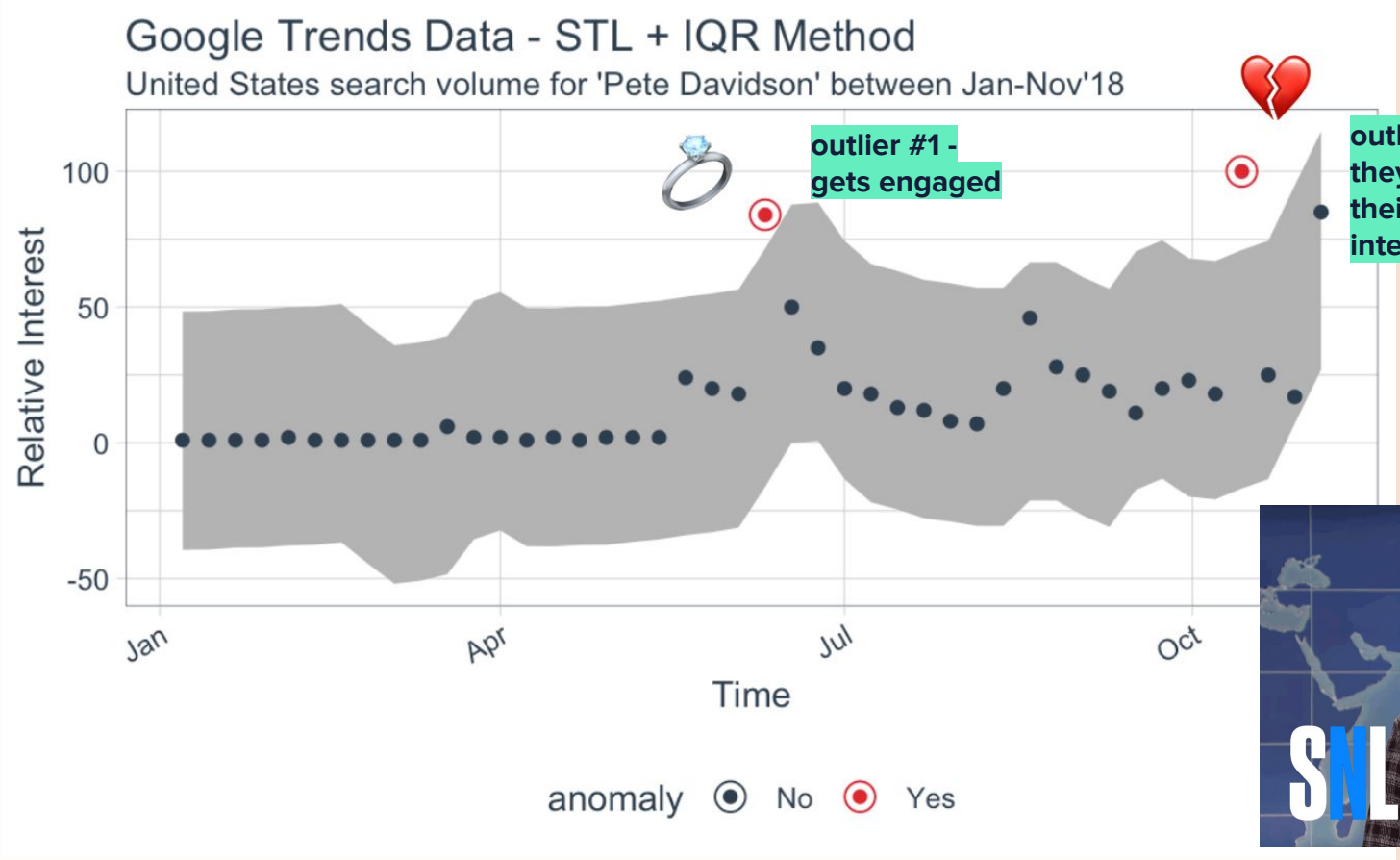
recent news, and 2018 data only -- seasonality is not really a factor, so we go back to using STL + IQR

codecademy

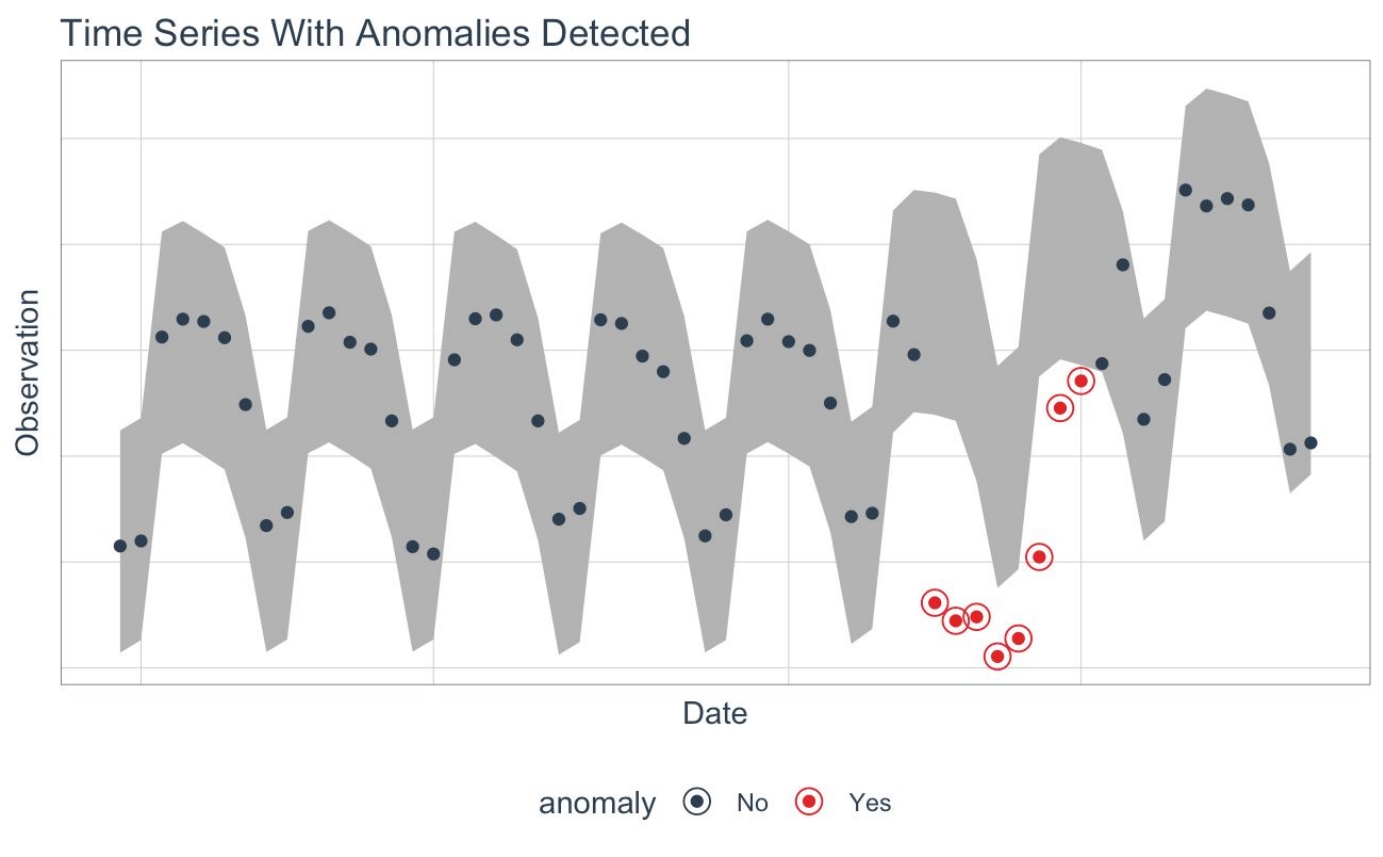
PETE DAVIDSON



PETE DAVIDSON



TRY THIS AT HOME!



Keywords To Try:

- NBA
- Weather
- Politicians
- Elon Musk
- Bitcoin
- Holidays
- Memes
- Events

... and anything else you might think of!

codecademy

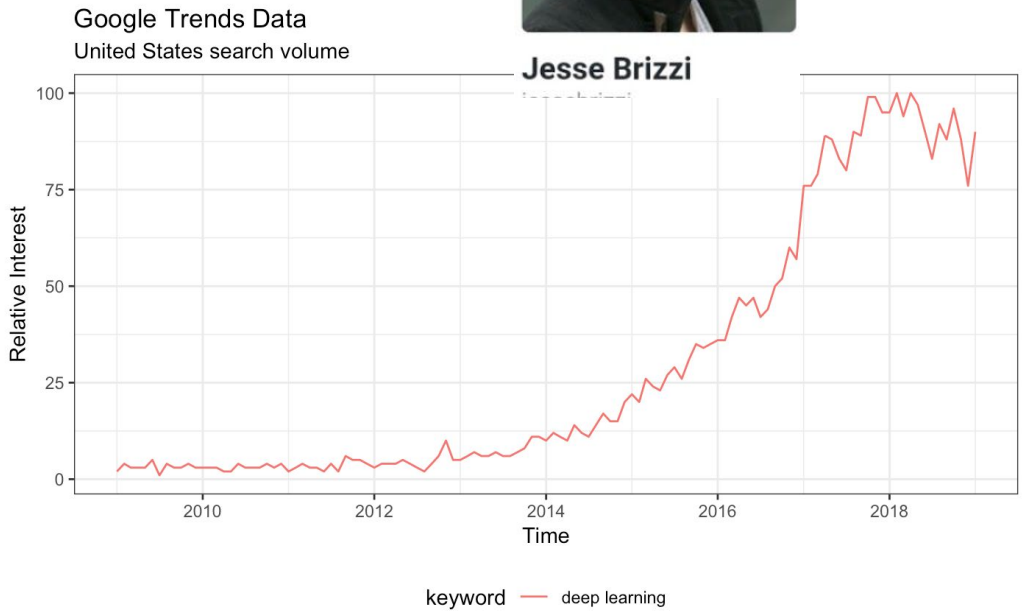
Additional Resources

- R Code + Notebook
- Introducing Anomalize
- Github: Anomalize
- Facebook Prophet
- NSSD Podcast: Sean Taylor (on prophet)
- Codecademy (Learn R coming soon!)

codecademy

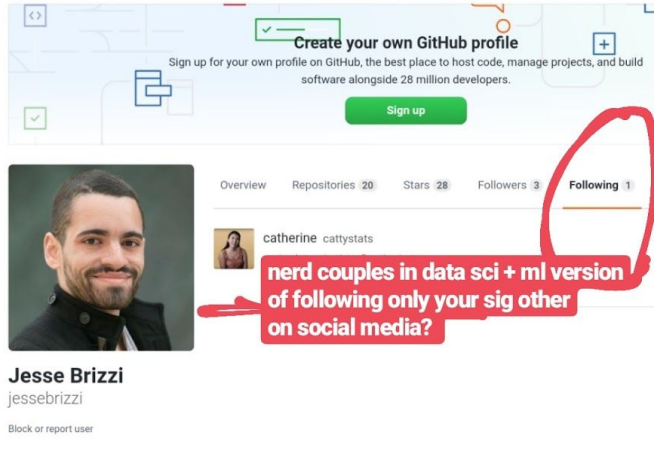
March Meetup:
Choosing A Deep Learning Library

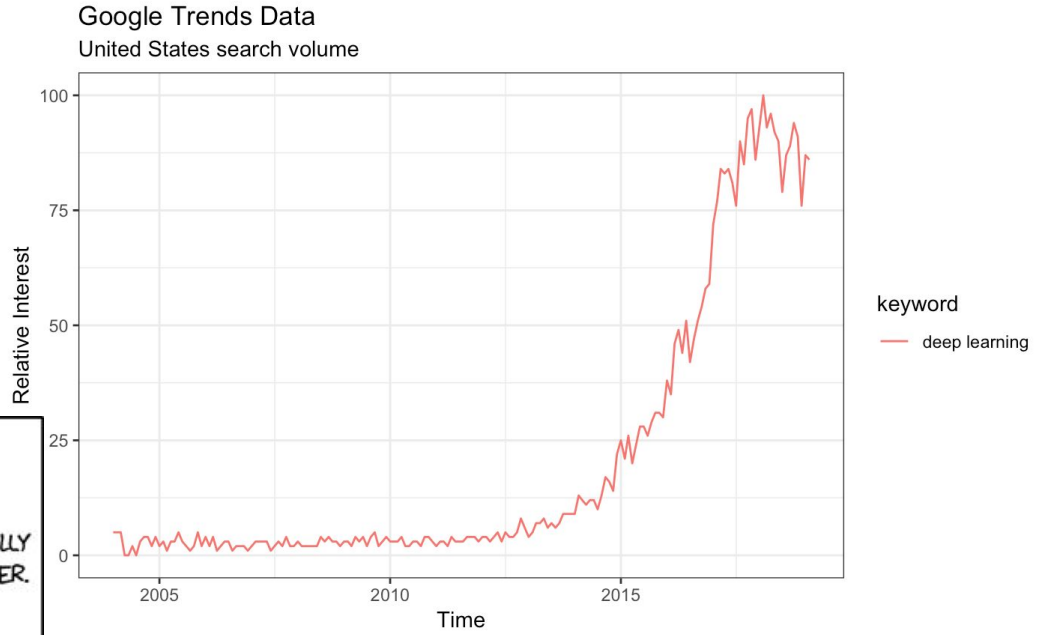
Jesse Brizzi

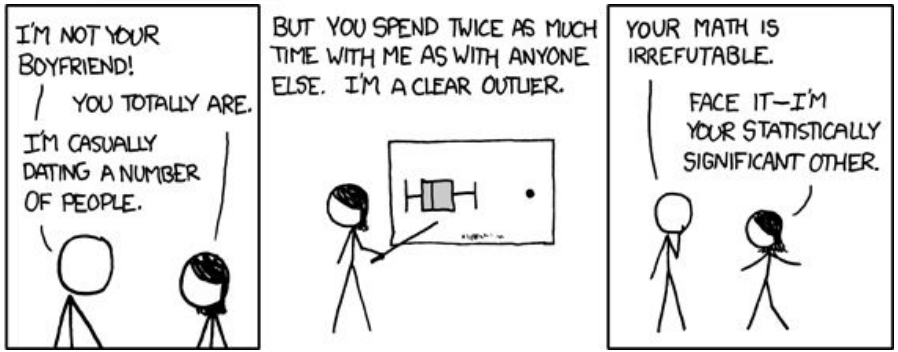


WIRED Curalate makes social sell with AI using Apache MXNet on AWS

March Meetup:
Choosing A Deep Learning Library







SUPPORT EACH OTHER.
GOOD LUCK AND HAVE FUN!

twitter @catherinezh

github @cattystats

#rstats

#rstatsnyc

#rladies

codecademy