R PROGRAMMING

Text Mining

INTRODUCTIONS

Who is this guy?

- bradleyboehmke.github.io
- bradleyboehmke@gmail.com
- @bradleyboehmke
- bradleyboehmke
- bradleyboehmke

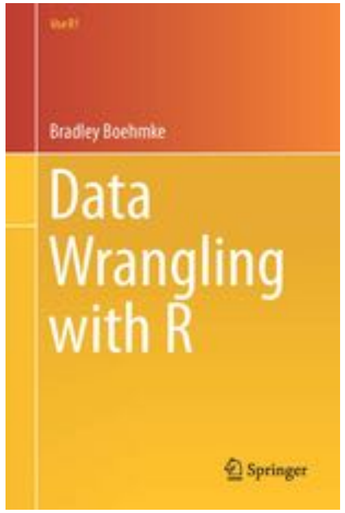

Dayton


AF INSTITUTE OF TECHNOLOGY

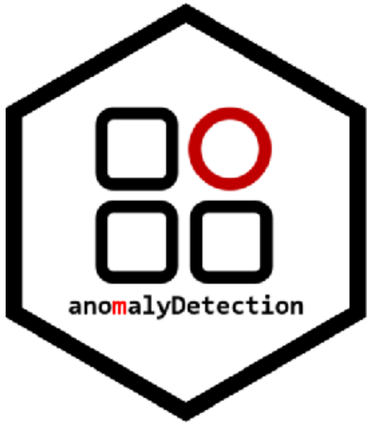

WAKE FOREST
UNIVERSITY


3451°

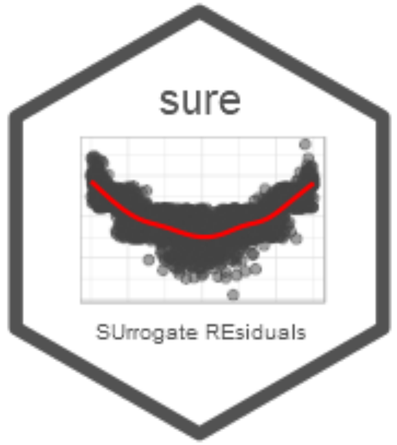

Kroger


Bradley Boehmke
Data Wrangling with R
Springer


DATA
SCIENCE LAB


anomalyDetection

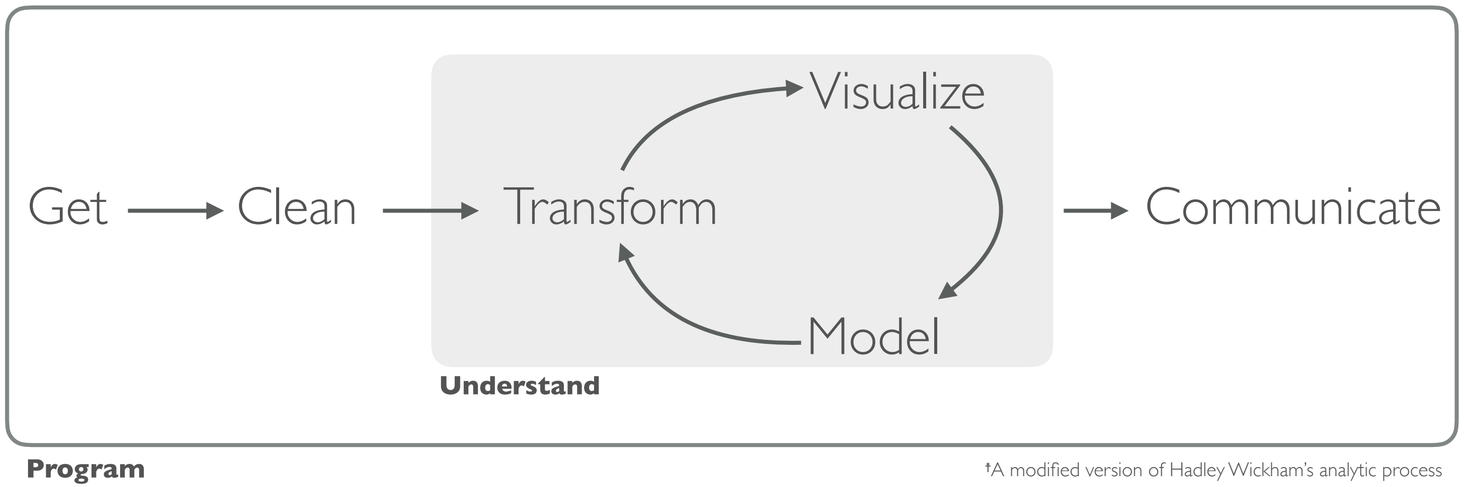

Leverage Products
Strategic Products
Routine Products
Bottleneck Products
KraljicMatrix


sure
SUrrogate REsiduals


vip

SETTING THE EXPECTATIONS

- Introduction to R
- Intermediate R
- **Text Mining with R**
- Applied Analytics with R
- Machine Learning with R (May 14-15)



†A modified version of Hadley Wickham's analytic process

SETTING THE EXPECTATIONS

Day 1

- Understanding the tidyverse
- Regular expressions
- Organizing unstructured text
- Frequency analysis

Day 2

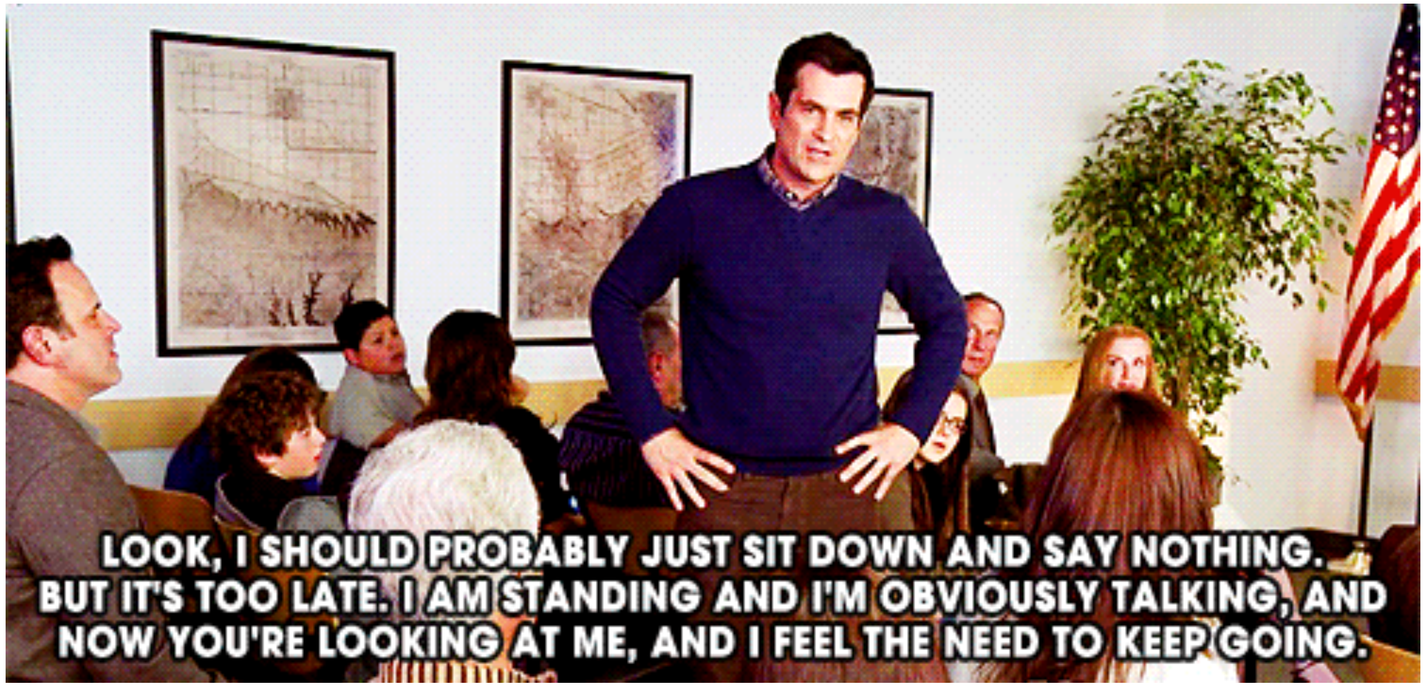
- Sentiment analysis
- Word association
- Topic modeling
- Predictive modeling

organize & clean ⇢ ***describe & predict***

You will be overwhelmed!

My Teaching Philosophy

THIS IS MEANT TO BE A DISCUSSION



YOUR TURN!

Lots of hands-on coding exercises

Strong proponent of collaborative work!

YOUR TURN!

Introduce yourself to your neighbors:

Who are you and what is your experience with R?

Regarding the topics that will be covered what are your strengths? Weaknesses?

WARM-UPS

Enough chit-chatting, time to code!

VECTOR EXERCISES

1. *check out the built-in vector* `state.region`

2. *what type of data does this contain?*

3. *Subset for only north central states. How many north central states are there?*

4. *Change state.name to a character variable, add* `state.region` *values as names to the* `state.name` *vector, subset for north central states.*

SOLUTIONS

```
# 1
state.region

# 2
class(state.region)
[1] “factor"

# 3
nc_states <- state.region[state.region == "North Central"]
length(nc_states)
[1] 12

# 4
state.name <- as.character(state.name)
names(state.name) <- state.region
state.name[names(state.name) == "North Central"]
```

MATRIX EXERCISES

1. check out the built-in **`VADeaths`** *matrix data*

2. what attributes does **`VADeaths`** *have?*

3. Calculate averages for each column and row

4. Can you figure out how to add these averages to your table so the output looks like:

	Rural Male	Rural Female	Urban Male	Urban Female	Avg_by_Age
50-54	11.70	8.70	15.40	8.40	11.050
55-59	18.10	11.70	24.30	13.60	16.925
60-64	26.90	20.30	37.00	19.30	25.875
65-69	41.00	30.90	54.60	35.10	40.400
70-74	66.00	54.30	71.10	50.00	60.350
Avg_by_Local	32.74	25.18	40.48	25.28	30.920

SOLUTIONS

```
# 1. check out the built-in VADeaths
VADeaths

# 2. what attributes does VADeaths have?
attributes(VADeaths)

# Calculate averages for each column and row
Avg_by_Age <- rowMeans(VADeaths)
Avg_by_Local <- colMeans(VADeaths)

# 4. Can you figure out how to add these averages to your table
addmargins(VADeaths, FUN = mean)
      Rural Male Rural Female Urban Male Urban Female   mean
50-54      11.70         8.70      15.40          8.40 11.050
55-59      18.10        11.70      24.30         13.60 16.925
60-64      26.90        20.30      37.00         19.30 25.875
65-69      41.00        30.90      54.60         35.10 40.400
70-74      66.00        54.30      71.10         50.00 60.350
mean       32.74        25.18      40.48         25.28 30.920
```

DATA FRAME EXERCISES

1. *Load the* `nycflights13` *package*

2. *Using the* `flights` *data, select the first 1000 rows and the following columns:* `month, dep_delay, carrier, distance, time_hour`. *Save this as* `small_flights`

3. *Look at the structure and summary of* `small_flights`

4. *Rename the columns of small_flights to* `"Month", "Delay", "Carrier", "Distance", “Date-Time”`

5. *Look at the first and last 15 rows*

SOLUTIONS

```
# 1. Load the nycflights13 package
library(nycflights13)

# 2. select the first 1000 rows and the following columns: month, dep_delay, carrier, distance, time_hour.
# Save this as small_flights
small_flights <- flights[1:1000, c("month", "dep_delay", "carrier", "distance", "time_hour")]

# 3. Look at the structure and summary of small_flights
str(small_flights)
summary(small_flights)

# 4. Rename the columns of small_flights to c("Month", "Delay", "Carrier", "Distance", “Date-Time”)
names(small_flights) <- c("Month", "Delay", "Carrier", "Distance", "Date-Time")

# 5. Look at the first and last 15 rows
head(small_flights, 15)
tail(small_flights, 15)
```

LIST EXERCISES

1. *Create this this regression model:*

```
flight_lm <- lm(arr_delay ~ dep_delay + month + carrier,
                data = flights)
```

2. *What list items does* `flight_lm` *contain?*

3. *Extract the residuals from the* `flight_lm` *list*

4. *What is the min, max, median, and mean of these residuals?*

SOLUTIONS

```
# 1. Create this this regression model:
flight_lm <- lm(arr_delay ~ dep_delay + month + carrier, data = flights)

# 2. what list items does flight_lm contain?
names(flight_lm)
 [1] "coefficients"  "residuals"     "effects"       "rank"          "fitted.values"
 [6] "assign"        "qr"            "df.residual"   "na.action"     "contrasts"
[11] "xlevels"       "call"          "terms"         "model"

# extract residuals from flight_lm list
residuals <- flight_lm$residuals

# compute summary statistics
summary(residuals)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-109.000  -10.980   -2.066    0.000    8.602  207.200
```


LET'S GET STARTED!