cobec CONSULTING

QUANTIFYING THE FUTURE

When Data Isn't Enough

Kellie Scarbrough, ICEAA 2019, Tampa

May 2019

Agenda

- Bad Data
- Government Data 2.0
- Pre-Processing and Exploring
- Text Mining in R
- Visualizing Data

What makes data “bad”?

BAD DATA

"Bad" = Inaccurate

Inaccurate "big data" sets can still have predictive power

Messy ≠ "Bad"

Analysts already have the skills to clean messy data:

- Normalization
- Removal of duplicates
- Correction of errors and blanks
- Correction of spelling
- Standardization
- Transformation

How government data has evolved

GOVERNMENT DATA 2.0

Big Data 1.0 and 2.0

1.0: The original adoption of internet technologies and electronic commerce

Businesses establish their online presence

2.0: The establishment of infrastructure and processes to collect large volumes of data

Businesses shift to data-analytic thinking

Foster, Provost and Tom Fawcett. *Data Science for Business: What You Need to Know About Data Mining and Analytics Thinking*. O'Reilly, 2013.

Gov. Data 1.0 and 2.0

1.0: The adoption of systems with streamlined data collection and reporting

Government establishes access to data

2.0: Government adopts best practices centered around data analytics

Government uses data to improve the estimating, budgeting, and execution of its programs

Foster, Provost and Tom Fawcett. *Data Science for Business: What You Need to Know About Data Mining and Analytics Thinking*. O'Reilly, 2013.

Government Big Data Initiatives

- California Natural Resources Agency's shared services data lake
- Joint Improvised Thread Defense Operations operations research and process improvement
- Department of Homeland Security integration of various departments
- GSA's Data Center Optimization Initiative (DCOI)
- DOE's Scalable Data Management, Analysis, and Visualization (SDAV) institute
- USGS Big Data for Earth System Science

Going Rogue

Data science is a broad field that can be described as[1]:

$$\begin{aligned}\text{data science} = \{\ &\text{statistics} \cap \text{informatics} \cap \\ &\text{computing} \cap \text{communication} \cap \text{sociology} \cap \\ &\text{management} \mid \text{data} \cap \text{domain} \cap \text{thinking}\ \}\end{aligned}$$

Analysts who do not adapt to big data and the evolution of data analytics will produce inferior estimates and models.

1. Longbing, Cao, "Data science: challenges and directions." In: *Communications of the ACM* 60.8 (Aug. 2017), pp. 59-68.

What techniques can analysts use?

PRE-PROCESSING AND EXPLORING

Visual Aids and R

- Visualization can be used during pre-processing to:
  - Identify inaccurate values
  - Find missing values
  - Find duplicate values
  - Establish candidates for bins
  - Identify places to consolidate data
  - Find outliers
  - Assess relationships

R and Resources

- R is a command line interface and was chosen because it is free, available to any analyst
- R is a powerful statistical computing environment
- Requires learning the R programming language
- Can be used with GUI like RStudio to facilitate use, reduce learning curve
- Code for the examples shown here can be found in the corresponding paper for this presentation and in many online forums – you can easily adapt it for your own needs

Inspecting Data

When reading in files to R, be sure to set argument stringsAsFactors equal to FALSE:

orders.df <- read.csv("C:/Users/…/orders.csv", stringsAsFactors = False)

head() returns the first six rows of data

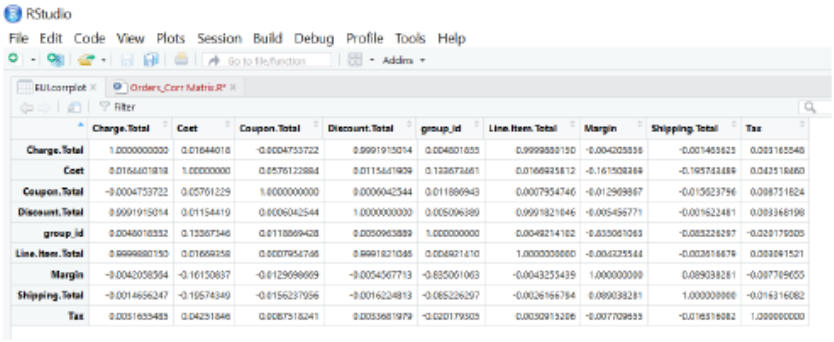
```
head(orders.df)
str(orders.df)
summary(orders.df)
```

```
> str(orders.df)
'data.frame':   275932 obs. of  68 variables:
 $ ï..Number.of.Records  : int  1 1 1 1 1 1 1 1 1 1 ...
 $ Status..group.        : chr  "active" "active" "active" "active" ...
 $ Billing.Id            : int  42 9090 15702 9090 8918 8 44 44 20 8 ...
 $ braintree_customer_id : chr  "" "" "" "" ...
 $ business_name         : chr  "ClearSight Studio" "ClearSight Studio" "ClearSight Studi
 $ Charge.Total          : num  10.19 19.63 12.12 39.62 7.12 ...
 $ contact_me            : chr  "False" "False" "False" "False" ...
 $ Cost                  : num  1.65 7.7 1.65 23.1 1.9 0 0 0 0 0 ...
 $ Coupon.Id             : int  9 NA NA NA 15 NA 9 9 NA NA ...
 $ Coupon.Total          : num  3.16 0 0 0 5 0 3.14 3.16 0 0 ...
```

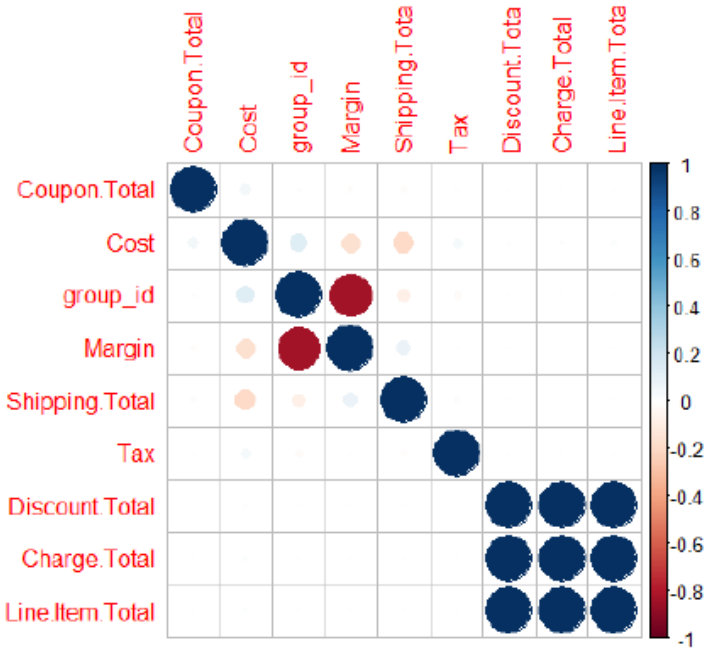
```
        :213974   (Other)        :210820
Discount..Groups.     discount       Discount.Total
Min.   :0.33      Min.   :0        Min.   :   -170.0
1st Qu.:0.33      1st Qu.:0        1st Qu.:      1.0
Median :0.33      Median :0        Median :      6.0
Mean   :0.33      Mean   :0        Mean   :     21.9
3rd Qu.:0.33      3rd Qu.:0        3rd Qu.:     17.9
Max.   :0.33      Max.   :0        Max.   :1100115.0
NA's   :275155    NA's   :275905
                  Email
```

Exploring Data

There are several functions available to generate correlation matrices and visualize the results



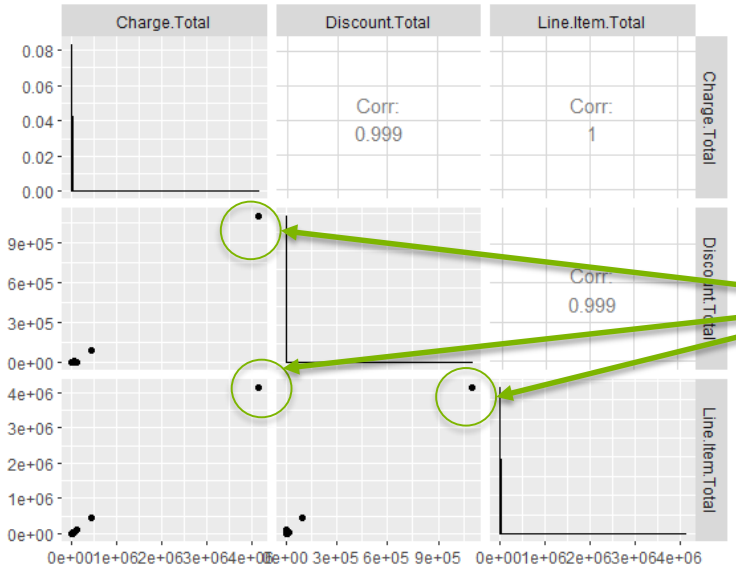
(a) Correlation matrix from EUI.corrplot



(b) EUI corrplot() result

Explore with ggpairs()

```
#explore potential drivers of total charge
ggpairs(orders.df[,c(6,20,38)])
```



There is an obvious outlier.

Pull the max value to see what the value is for further troubleshooting

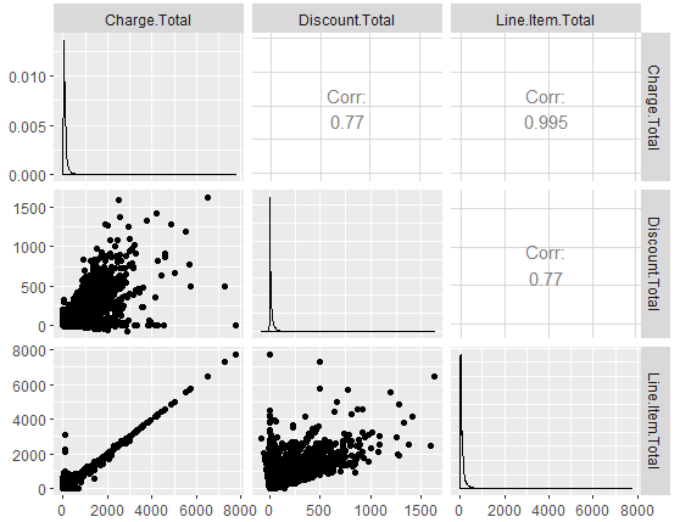
```
#Find max value for troubleshooting
max(orders.df$Charge.Total,   na.rm = FALSE, dims = 1, n = NULL)
```

Result: $4.1M – We know this is an error!

Error Correction: Apply Filter

```
orders.df <-orders.df %>% filter(Status..Orders. == "placed" |
      Status..Orders. == "manually_placed" |
      Status..Orders. == "shipped" |
      Status..Orders. == "shipping")

summary(orders.df$Status..Orders.)
```

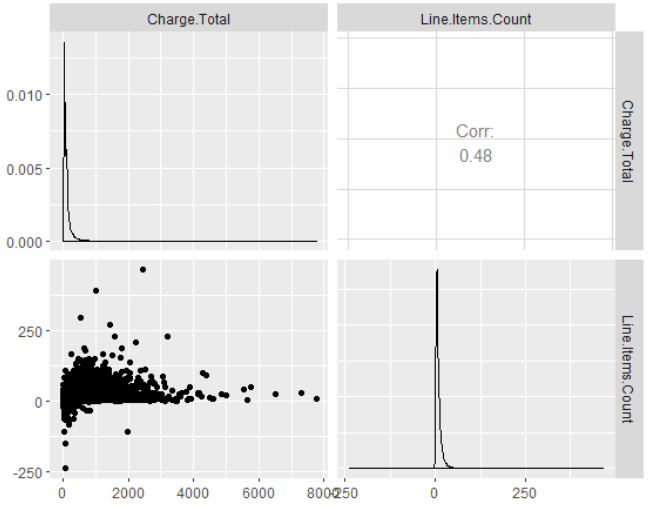
```
> summary(orders.df$Status..Orders.)
      canceled      cancelled           cart      fraudulent manually_placed         placed     processing
             0              0              0              0           1641           3004              0
    saved_cart        shipped       shipping
             0         224210           4306
```



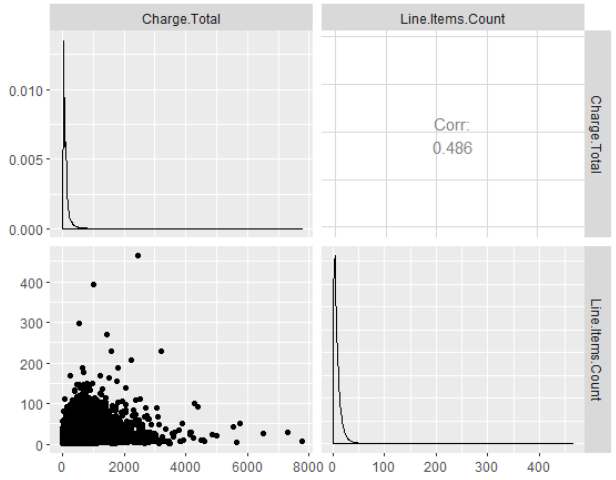
Much better!

Another Example: Negative Values

```
#explore line item count and total charge
ggpairs(orders.df[,c(6,39)])
```




```
#remove negative line item counts and re-run
orders_licorr.df <- orders.df %>% filter(Line.Items.Count > 0)
ggpairs(orders_licorr.df[,c(6,39)])
```



How to extract value from text fields

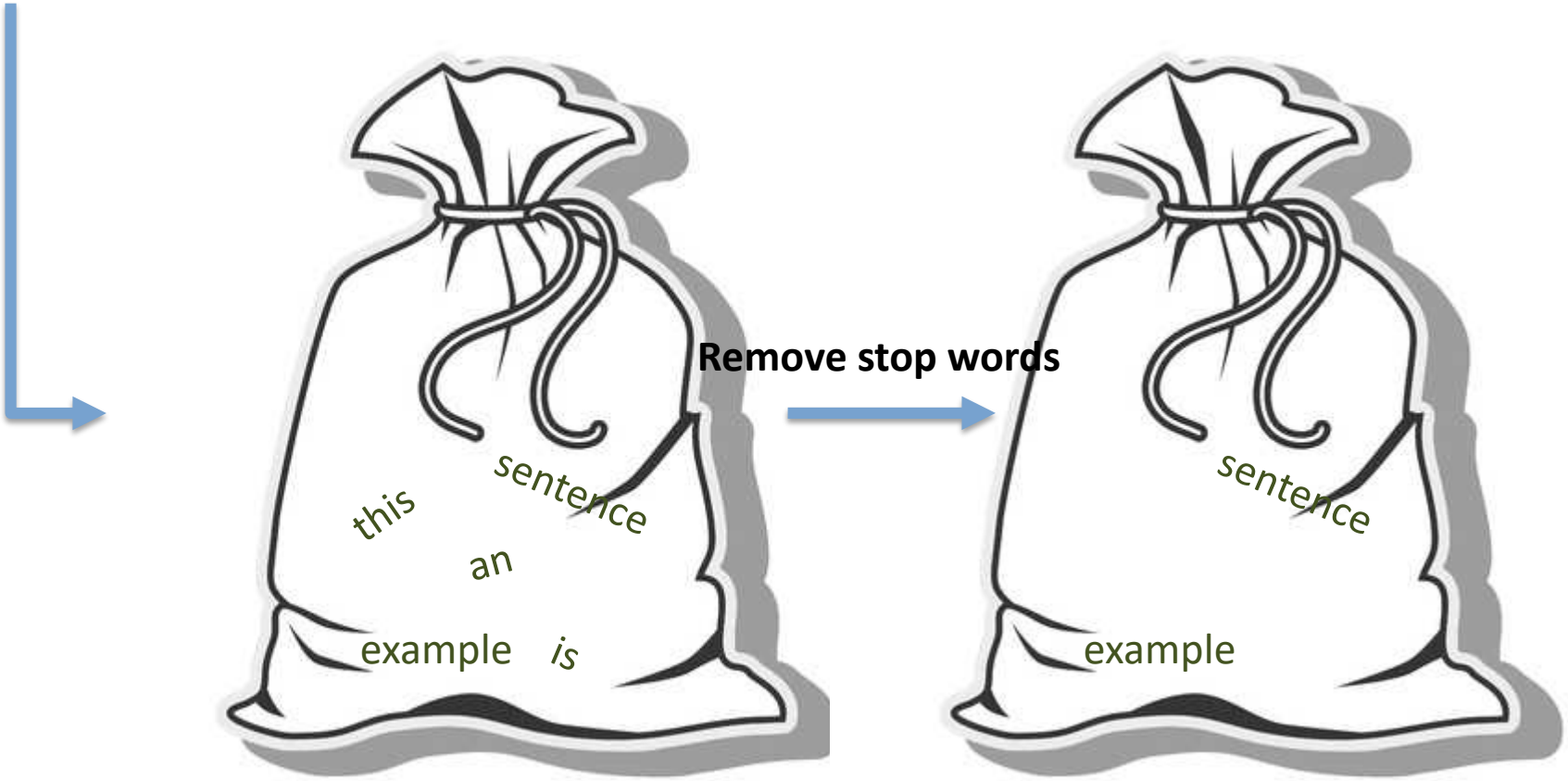
TEXT MINING

Text mining overview

- Text fields are common in government data
- We will use “bag-of-words” text mining
  - Each word is treated as an individual item
  - Disregards sentence structure
  - Requires removal of stop words

Bag-of-Words

This sentence is an example.



The Code: Create a Corpus

The library Rweka contains a function which allows you to view words in groups

```
library(RWeka)
tokenizer <- function(x) {
NGramTokenizer(x, Weka_control(min = 1, max = 2))
}
```

The primary tools for text mining are vectors and corpora

```
#Create vector
parts_source <- VectorSource(parts_desc_clean)
#Create corpus
parts_corpus <- VCorpus(parts_source)
```

Clean the Corpus

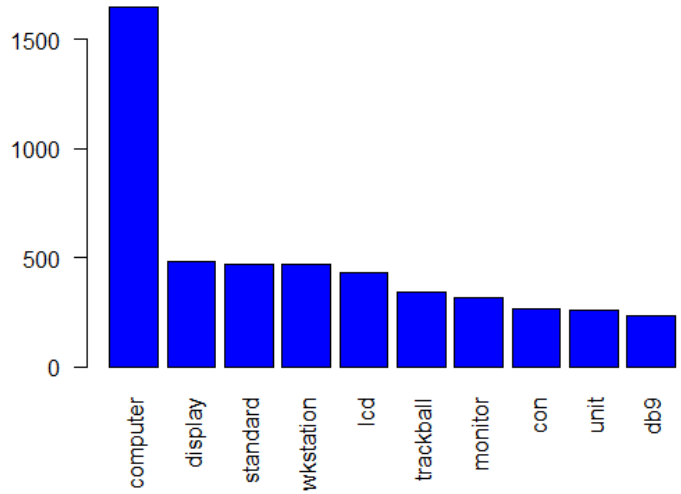
```
# Alter the function code to match the instructions
clean_corpus <- function(corpus) {
# Remove punctuation
corpus <- tm_map(corpus, removePunctuation)
# Transform to lower case
corpus <- tm_map(corpus, content_transformer(tolower))
# Remove stop words using common English stop words
corpus <- tm_map(corpus, removeWords, c(stopwords("en")))
# Strip whitespace
corpus <- tm_map(corpus, stripWhitespace)
return(corpus)
}

parts_clean <- clean_corpus(parts_corpus)
parts_tdm <- TermDocumentMatrix(parts_clean)
parts_m <- as.matrix(parts_tdm)
term_frequency <- rowSums(parts_m)
term_frequency<- sort(term_frequency, decreasing = TRUE)
```

The Code: View the Results

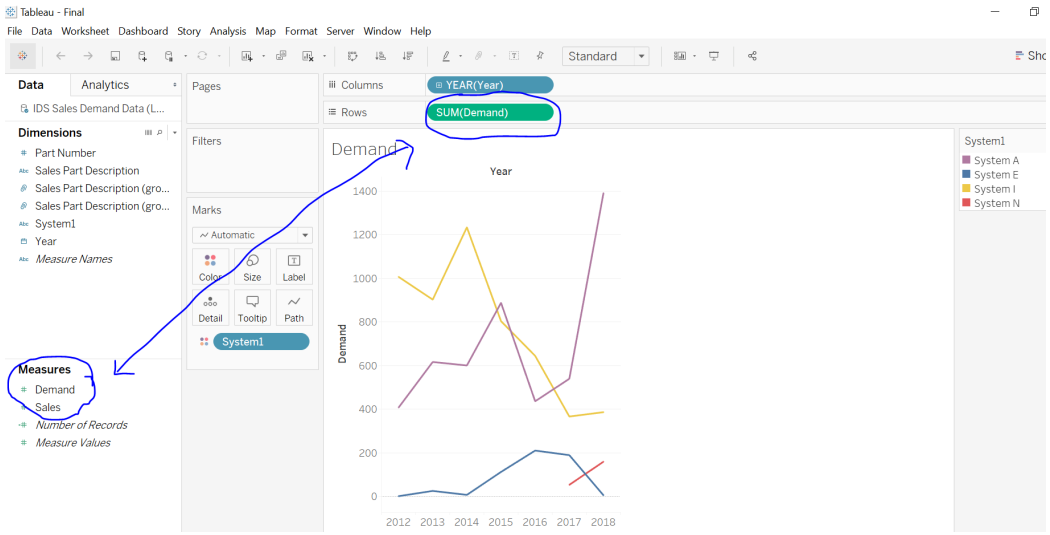
```
# View the top 10 most common words
term_frequency[1:10]

# Plot a barchart of the 10 most common words
barplot(term_frequency[1:10], col ="blue", las=2)

# Load wordcloud package
library(wordcloud)

terms_vec <- names(term_frequency)

wordcloud(terms_vec, term_frequency, max.words = 50, colors = "red")

wordcloud(terms_vec, term_frequency, max.words =50,
colors = c("grey80","darkgoldenrod1","tomato"))
```

Text Mining Results

```
> # View the top 10 most common words
> term_frequency[1:10]
 computer   display  standard wkstation       lcd trackball   monitor       con      unit       db9
     1648       486       469       469       435       347       316       266       260       234
>
```

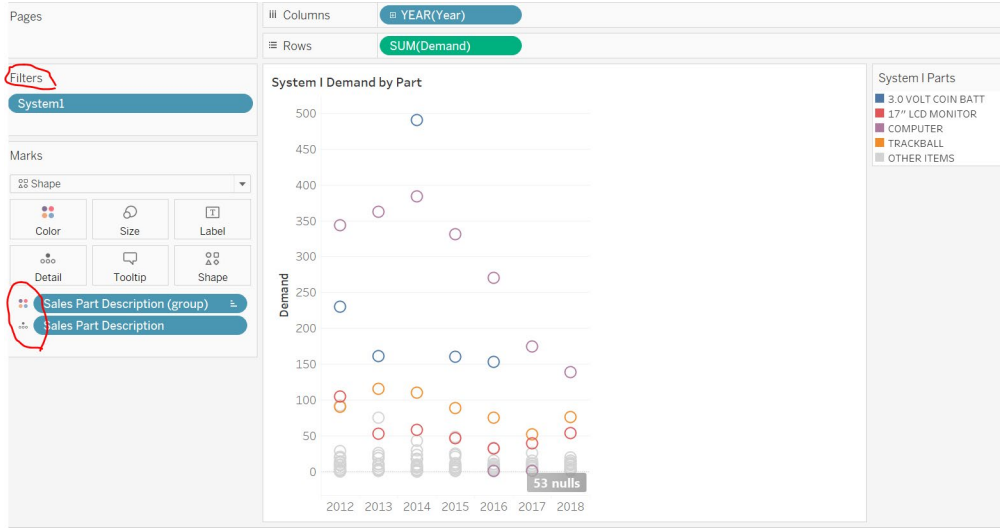


Using Tableau for Exploration



In Tableau, we can quickly interchange variables and break out data by various dimensions

We can also quickly group data and filter to drill down and extract meaningful insights



cobec CONSULTING

How to Communicate Visually

VISUALIZING RESULTS

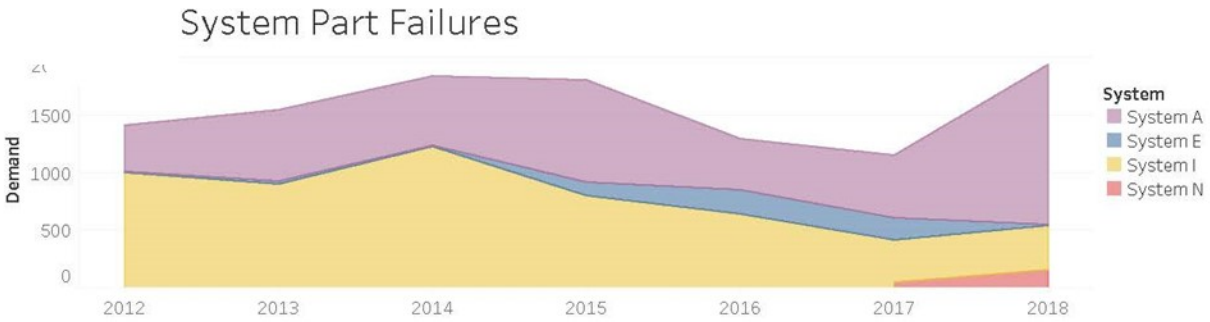
Visualizing the Results

If you can't communicate and defend your analysis, your efforts are wasted.

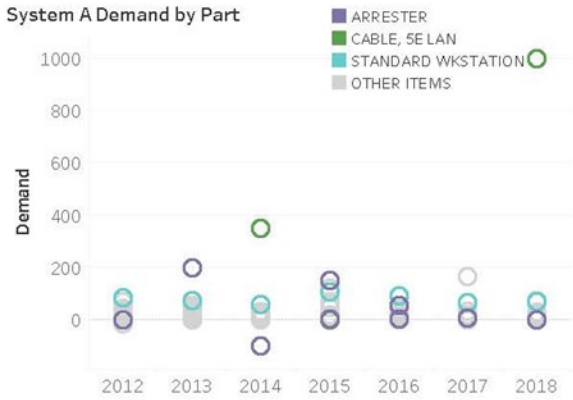
Provide results that are:

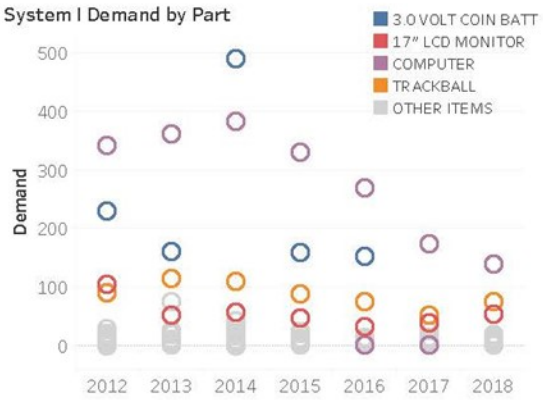
- Clear
- Meaningful
- Actionable

Logistics Dashboard Example



Most demand for parts come from Systems A and I. The highest demand for parts are computers for System I, indicating complete system failures are occurring. Failures spiked in 2014 and have been declining steadily since. Steady failures of 17" monitors, 3.0 volt coin batteries, and trackballs are also occurring. System A experiences a moderate but steady level of workstation replacement, and periodic spikes in 5E LAN cable and arrester part demands.





Summary

- Bad vs messy
- Role of data science in data analytics
- Explored some techniques for inspecting large data sets
- Explored text mining
- Visualizing the results

QUESTIONS?