Welcome

IBF PRESENTS
PREDICTIVE ANALYTICS & MINING BIG DATA

Sara Brumbaugh

sara@ceresAnalytics.com
617-519-3151

Institute of Business Forecasting & Planning

OVERALL AGENDA

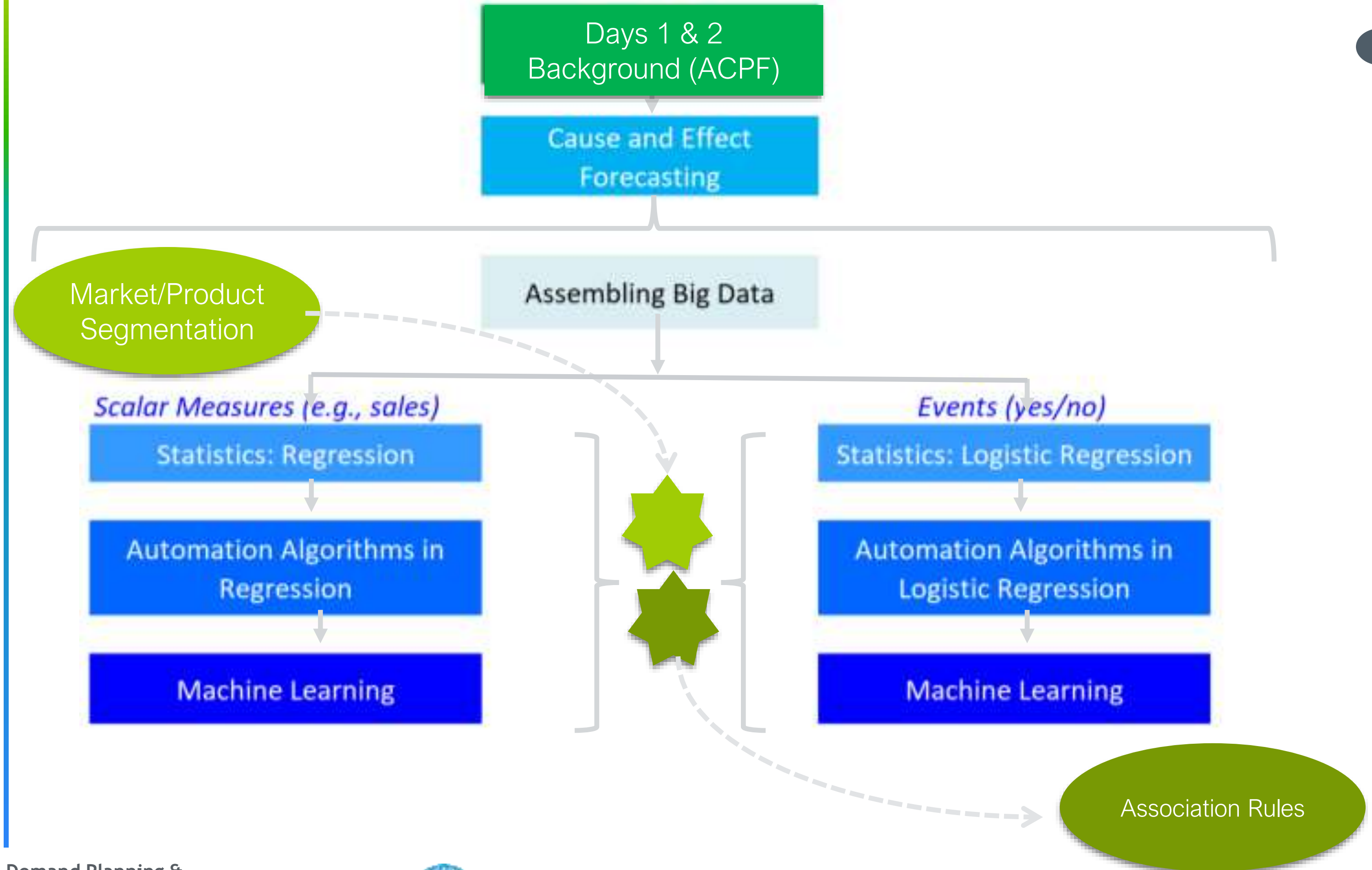
A.M.

1. The Quantitative Skill Set
2. Databases, Data Extraction & Data Assembly
3. Scripting (a survey approach)

P.M.

4. Descriptive Techniques
5. Predictive Techniques (including Association Rules)

ANATOMY OF TODAY'S TOPICS



THE QUANTITATIVE SKILL SET

An Overview

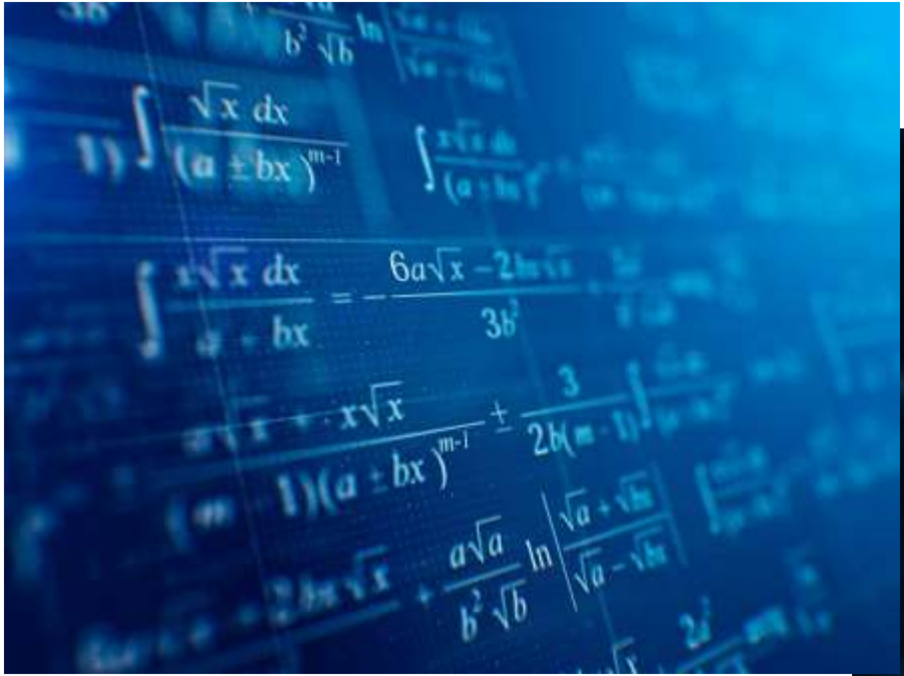
1

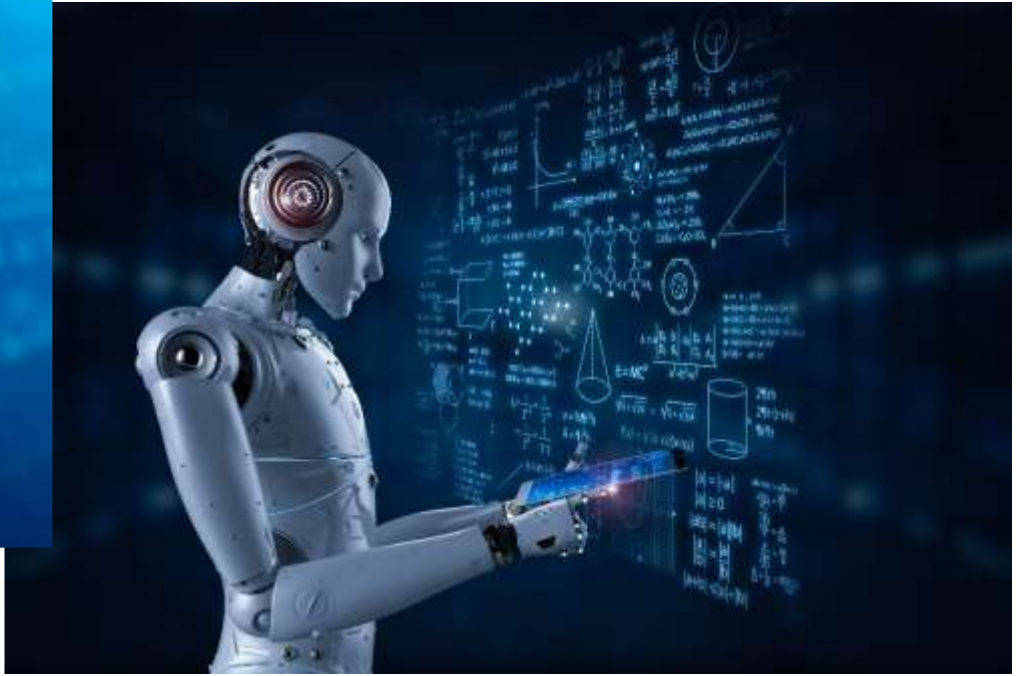
FORECASTING

THE BEST JOB IN AN ORGANIZATION

- Opportunities for skills
- Data access
- Working relationships
- Answerable questions

STATISTICS VS. MACHINE LEARNING





QUANTITATIVE ANALYSIS

EXAMPLES FROM STATISTICS

- Quantitative Analysis
  - Descriptive
    - Univariate
      - Distributions (Mean, Standard Deviation)
      - Comparisons (of means, medians, or proportions)
      - Coincidence (of categories)
    - Multivariate
      - Cluster Analysis
      - Linear Discriminant Analysis
      - Factor Analysis
      - Principal Components
      - Independent Components
  - Predictive
    - Univariate
      - Time Series Analysis
    - Multivariate
      - Linear Regression
      - Logistic Regression

QUANTITATIVE ANALYSIS

VOCABULARY FOR STATISTICS VS. MACHINE LEARNING

Statistics	Machine Learning
Prediction	Supervised Learning
Description	Unsupervised Learning
Independent Variables (x) *(predictors)*	Features, Inputs
Dependent Variable (y)	Target or Training Value
Predicted (Fitted) Dependent Variable ($\hat{y}$)	Output
Constant	Bias
Residuals	Errors

Some say that many of machine learning's techniques are simply "borrowed" from statistics…

GIVEN A WEALTH OF DATA…

How do I predict a metric?
(sales, market return, or weight loss)

How do I predict a yes/no event?
(response to a promotion, probability of attrition)

How do I know what will drive or predict a result?
(… all of the above!)

How can I construct market segments?
(customers or patients or portfolios)

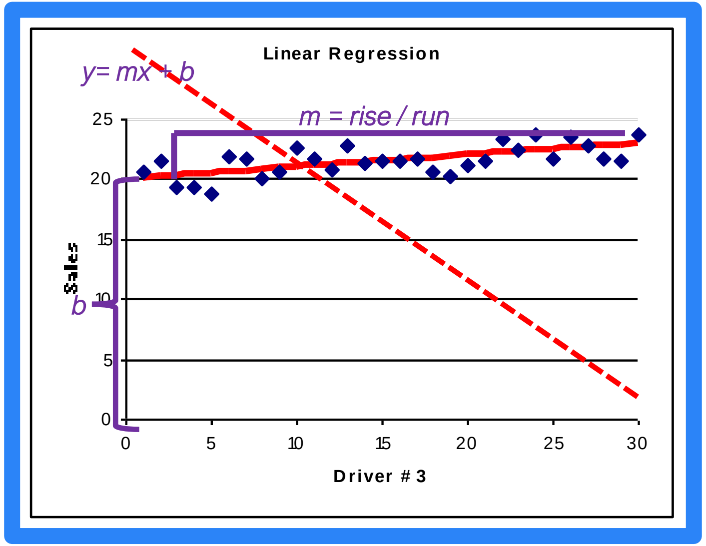
What products are purchased together?
(or events occur together)

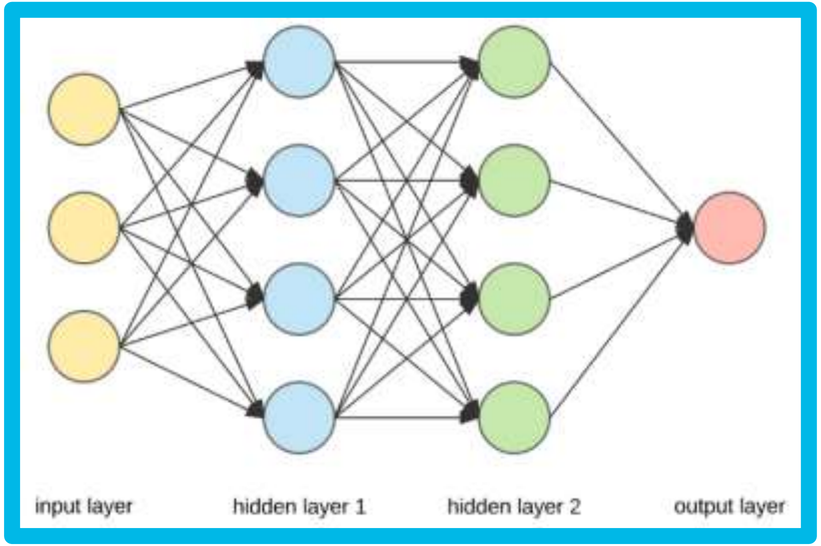
HOW DO I PREDICT A METRIC?

Examples

Statistics: Linear Regression Analysis

Machine Learning: Neural Network



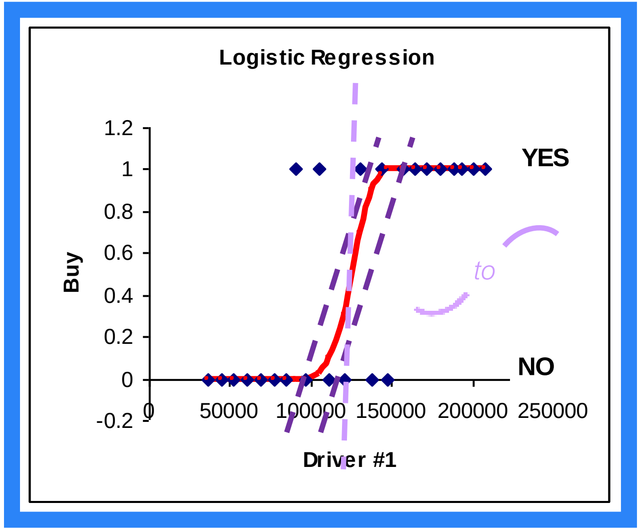


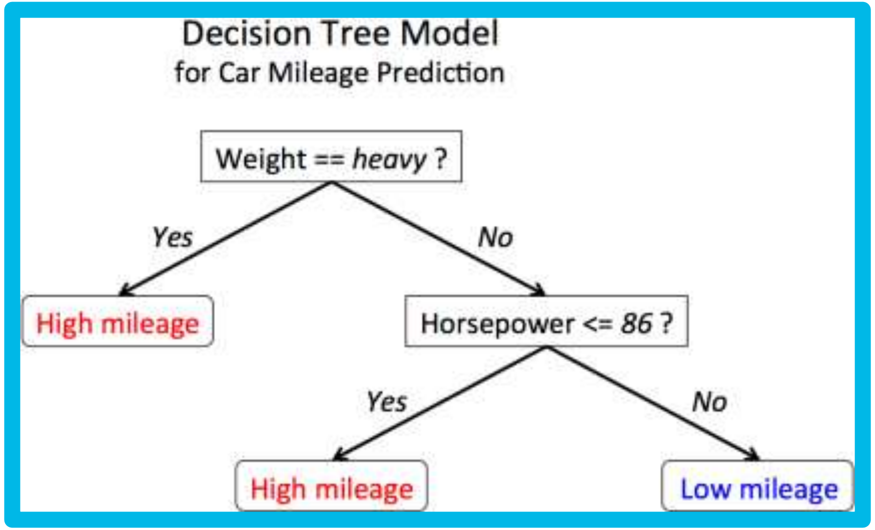
https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6

HOW DO I PREDICT A YES/NO RESULT?

Examples

Statistics: Logistic Regression Analysis
Machine Learning: Neural Network, CHAID(Decision Tree), Support Vector Machine



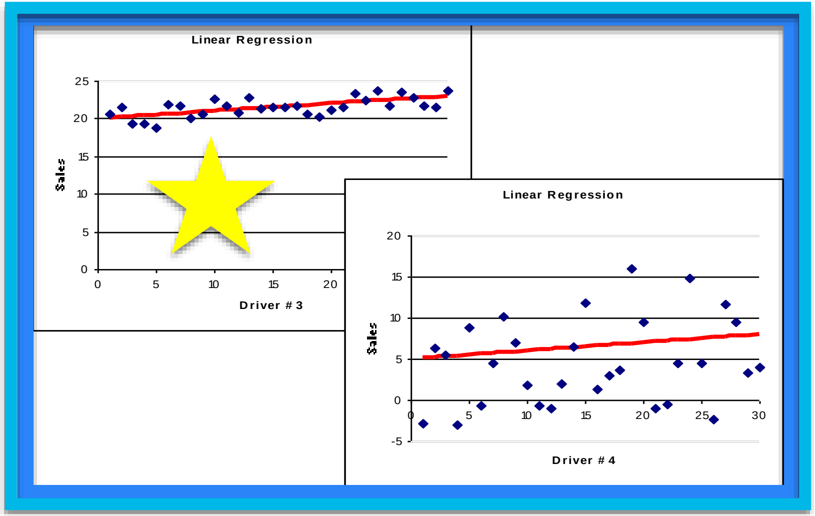


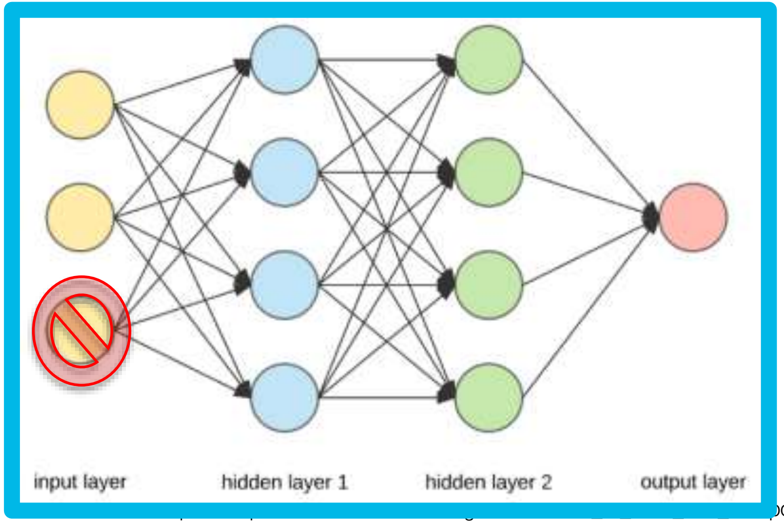
https://www.crondose.com/2016/07/easy-way-understand-decision-trees/

HOW DO I KNOW WHAT WILL DRIVE A RESULT?

Examples

Statistics: Automated variable selection (stepwise, all possible subsets)
Machine Learning: Filter Methods, Wrapper Methods, Embedded Methods





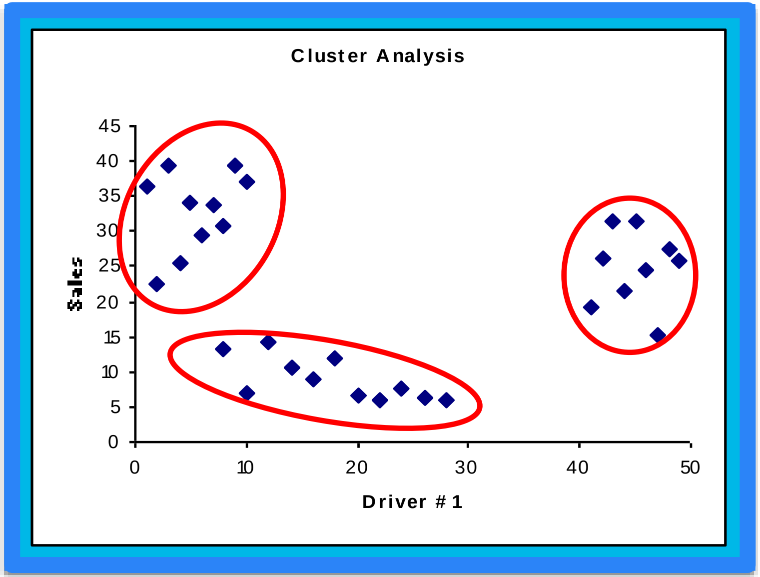
Institute of Business Forecasting & Planning

HOW CAN I CONSTRUCT MARKET SEGMENTS?

Examples

Statistics: Cluster Analysis

Machine Learning: Cluster Analysis



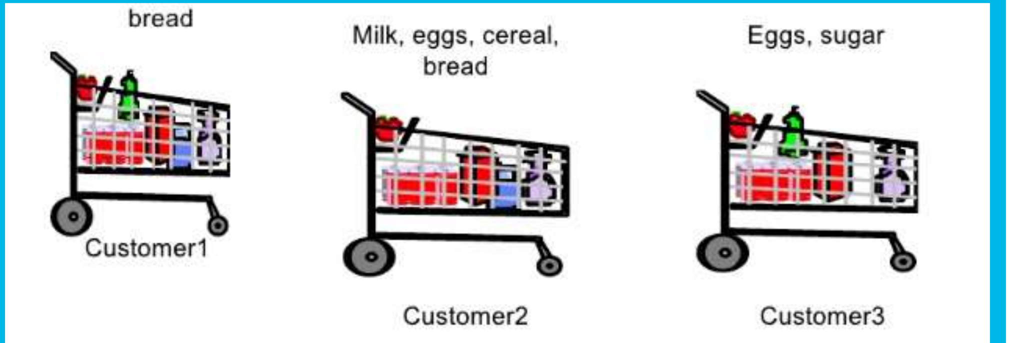
WHAT PRODUCTS ARE PURCHASED TOGETHER?

Examples

Statistics: Contingency Table, χ^2
Machine Learning: Association Rules

		Bought Wine		
		Yes	No	Σ
Bought	Yes	30	5	35
Cheese	No	5	60	65
	Σ	35	65	100
χ^2 =	?			

https://www.slideshare.net/aorriols/lecture13-association-rules



CROSS-DISCIPLINARY APPLICATIONS

EXAMPLE: REGRESSION ANALYSIS

	Forecasting *Linear regression*	Marketing *Logistic regression*	Market Research *Conjoint Analysis*
Type of dependent variable	Continuous	• Yes/No • Category	• Category • Rank
Dependent variable	• Sales • Orders • Shipments • Revenues	• Promotion response • Customer attrition	• Rating scale (linear) • Discrete choice (logistic)
Predictors	• Prior sales • Economic drivers • Promotions	• From transactions, "RFM" • Customer Attributes	Product attributes

THE MARKETING PERSPECTIVE
"RFM"

Recency

How recent was the last purchase?

Frequency

How often does the customer purchase?

Monetary

How much does the customer spend?

THE MARKET RESEARCH PERSPECTIVE
“PART WORTHS”

EFA’s for healthy joints and coat (premium)

Everyday adult

The packaging just might be unappealing?

CROSS-DISCIPLINARY APPLICATIONS

AGGREGATION TO FORECAST PREDICTORS

From….	Found by…	To…
Customer groups	Cluster Analysis	More certain forecast by customer class
Product-line groups	Cluster Analysis	More certain forecast by product class
Probability of customer attrition	Logistic Regression	Number of customers as forecast driver
New product market shares	Conjoint-based market simulator	New product forecast
Customer-level purchases	Association Rules	Product forecast

CROSS-DISCIPLINARY APPLICATIONS:

COMMON ELEMENTS

Task	Examples of Tools
Data Extraction and Assembly	SQL, “NoSQL”, Web Analytics, Excel
Analysis • Scripting • Spreadsheets • Other	 • R, Python, VBA for Microsoft Office • Excel • Commercial software, GUI apps
Visualization and Presentation	Generally accompanies analysis

DATABASES, DATA EXTRACTION, AND DATA ASSEMBLY

Databases, SQL, External Drivers, and Sample Queries

2

CUSTOMER-LEVEL DATA: DATA PREPARATION

- One record per customer, as may be available in a Customer Relationship Management System (CRMS) or relational database
- Customer-level measures may require aggregation from individual transactions, e.g.:

TRANSACTIONS TABLE (as of 05/31/22)

CustomerID	TransactionID	Date	ProductID	Quantity	PricePerUnit	PromoCode		
1000001	71102	06/03/2021	123A	1	29.99		*$144.95/year*	*Monetary*
1000001	34753	12/15/2021	123A	1	24.99	XMAS	*4x/year*	*Frequency*
1000001	118732	01/29/2022	123A	1	29.99		*91 days*	*Recency*
1000001	57778	02/28/2022	123A	2	29.99			
1000012	41429	08/15/2021	001B	5	1.99	BKTOSCHL	*$339.42/year*	*Monetary*
1000012	56629	09/15/2021	001B	3	2.29		*5x/year*	*Frequency*
1000012	56629	09/15/2021	213Z	1	53.99		*132 days*	*Recency*
1000012	71497	10/15/2021	001A	3	2.39			
1000012	30352	12/26/2021	973Z	1	249.99	POSTXMAS		
1000012	30147	01/19/2022	001B	5	2.29			

CUSTOMER OR ACCOUNTS TABLE

CustomerID	StreetAddress	City	State	ZipCode	MemberSince	LastPurchase	
1000001	123 Maple St	Owensboro	KY	42301	20190405	02/28/2022	
1000012	40 Pleasant St	Muncie	IN	47307	20161012	01/19/2022	
1000014	3 S Park Ave	Springfield	IL	62705	20180822	01/05/2020	*0/year*

PRODUCT-LEVEL DATA: DATA PREPARATION

- One record per product, as may be available in a Customer Relationship Management System (CRMS) or relational database management system (RDBMS)
- Product-level measures may require aggregation from individual transactions, e.g.:

TRANSACTIONS TABLE

CustomerID	TransactionID	Date	ProductID	Quantity	PricePerUnit	PromoCode		
1000001	71102	06/03/2021	123A	1	29.99		*# units*	*Volume*
1000001	34753	12/15/2021	123A	1	24.99	XMAS	*[$price x #units]*	*Revenue*
1000001	118732	01/29/2022	123A	1	29.99			
1000001	57778	02/28/2022	123A	2	29.99			
1000012	41429	08/15/2021	001B	5	1.99	BKTOSCHL		
1000012	56629	09/15/2021	001B	3	2.29			
1000012	56629	09/15/2021	123A	1	29.99			
1000012	71497	10/15/2021	001A	3	2.39			
1000012	30352	12/26/2021	9732	1	249.99	POSTXMAS		
1000012	30147	01/19/2022	001B	5	2.29			

PRODUCT-LEVEL DATA: DATA PREPARATION

1 OF 2

- One record per product, as may be available in a Customer Relationship Management System (CRMS) or relational database
- Like Customer-level data, Product-level measures may require aggregation from individual transactions, e.g.:

TRANSACTIONS TABLE

CustomerID	TransactionID	Date	ProductID	Quantity	PricePerUnit	PromoCode
1000001	71102	06/03/2021	123A	1	29.99	
1000001	34753	12/15/2021	123A	1	24.99	XMAS
1000001	118732	01/29/2022	123A	1	29.99	
1000001	57778	02/28/2022	123A	2	29.99	
1000012	41429	08/15/2021	001B	5	1.99	BKTOSCHL
1000012	56629	09/15/2021	001B	3	2.29	
1000012	56629	09/15/2021	123A	1	29.99	
1000012	71497	10/15/2021	001A	3	2.39	
1000012	30352	12/26/2021	9732	1	249.99	POSTXMAS
1000012	30147	01/19/2022	001B	5	2.29	

units — *Volume*

[$price x #units] — *Revenue*

PRODUCT-LEVEL DATA: DATA PREPARATION

2 OF 2

- Similarly, data for products and promotions may supplement aggregation of transactions

PRODUCT TABLE

ProductID	ProductName	ProductFamilyID	ProductFamilyName	DateIntroduced	Price	Cost	Price_fromDate	Price_toDate
001A	Keyboard Button 1	A	PC1	01/01/2017	2.39	1.91	01/01/2021	06/30/2022
001B	Keyboard Button 2	B	PC2	06/01/2020	2.29	1.95	01/01/2021	06/30/2022
123A	Backlit Keyboard PC1	A	PC1	01/01/2017	29.99	15.00	01/01/2021	06/30/2022
123B	Backlit Keyboard PC2	B	PC2	06/01/2020	35.99	19.79	01/01/2021	06/30/2022
. . .								
973Z	Desktop 8TB Drive	ES	External Storage	03/01/2021	249.99	125.0	01/01/2021	06/30/2024
123A	Backlit Keyboard PC1	A	PC1	01/01/2017	31.99		07/01/2022	06/30/2024
123B	Backlit Keyboard PC2	B	PC2	06/01/2020	37.99		07/01/2022	06/30/2024

PROMOTIONS TABLE

PromotionID	PromoCode	PromoName	ProductFamilyName	StartDate	EndDate	Discount
1001	SPRING	Spring Savings	PC1	03/23/2021	03/24/2021	0.15
1002	BKTOSCHL	Back to School	PC2	08/12/2021	09/05/2021	0.20
1003	XMAS	Christmas	PC1	11/26/2021	12/24/2021	0.33
1004	POSTXMAS	After-Christmas	PC2	12/26/2021	12/31/2021	0.50

BEFORE YOU COLLECT ANY DATA

WHAT YOU NEED TO KNOW

	A	B	C	D	E	F	G	H	I	J
1	YearQtr	Product1	Product2	Product3	UnempPct	CPI_U_Pct	RGDP	Time	Q4	
29	2018Q4	109893	3964	1253	3.83	1.4691	18733.7	28	1	
30	2019Q1	117173	2041	211	3.87	0.9574	18835.4	29	0	
31	2019Q2	2415	1670	118	3.60	3.1804	18962.2	30	0	
32	2019Q3	78754	3463	55	3.63	1.4842	19130.9	31	0	
33	2019Q4	147265	3691	608	3.60	2.4582	19215.7	32	1	
34	2020Q1	114262	4230	52	3.80	1.2983	18989.9	33	0	
35	2020Q2	50951	2762	90	12.97	-3.3590	17378.7	34	0	
36	2020Q3	29665	529	194	8.83	4.7939	18743.7	35	0	
37	2020Q4	133752	3458	515	6.77	2.2408	18924.3	36	1	
38	2021Q1	84110	3092	175	6.20	4.1187	19216.2	37	0	
39	2021Q2	69108	1344	61	5.90	8.1871	19544.2	38	0	
40	2021Q3	153447	2997	193	5.10	6.7159	19672.6	39	0	
41	2021Q4	138456	1197	1435	4.23	7.9122	20006.2	40	1	
42	2022Q1	144160	3868	82	3.80	9.2009	19924.1	41	0	*Holdout period* ✓
43	2022Q2	12391	2633	94	3.60	10.5309	19895.3	42	0	
44	2022Q3	92300	2953	111	3.57	5.6856	20039.4	43	0	
45	2022Q4				3.70	5.4000	20239.6	44	1	
46	2023Q1				3.80	4.5000	20277.5	45	0	*Near-term forecast* ✓
47	2023Q2				4.00	3.5000	20322.0	46	0	
48	2023Q3				4.30	3.1000	20499.5	47	0	
49	2023Q4				4.40	2.9000	20920.8	48	1	
50	2024Q1							49	0	
51	2024Q2							50	0	*Longer-term forecast* X
52	2024Q3							51	0	
53	2024Q4							52	1	
54										

- For **cause-and-effect forecasting**, drivers (independent variables, predictors) must be **available** for **both** the **historical** AND the **forecast** periods

SQL--BASICS

Extract data through a query

Basic query elements:

- SELECT: fields to choose (* for all)
- FROM: table to choose from
- WHERE: filter results
- ORDER BY: sort results

```
SELECT *
FROM customerTable
WHERE customerClass=‘HOME OFFICE’
ORDER BY customerID
```

SQL--AGGREGATION

Customer-level measures from transactions

Aggregation functions:

- COUNT
- SUM
- AVG
- MIN
- MAX

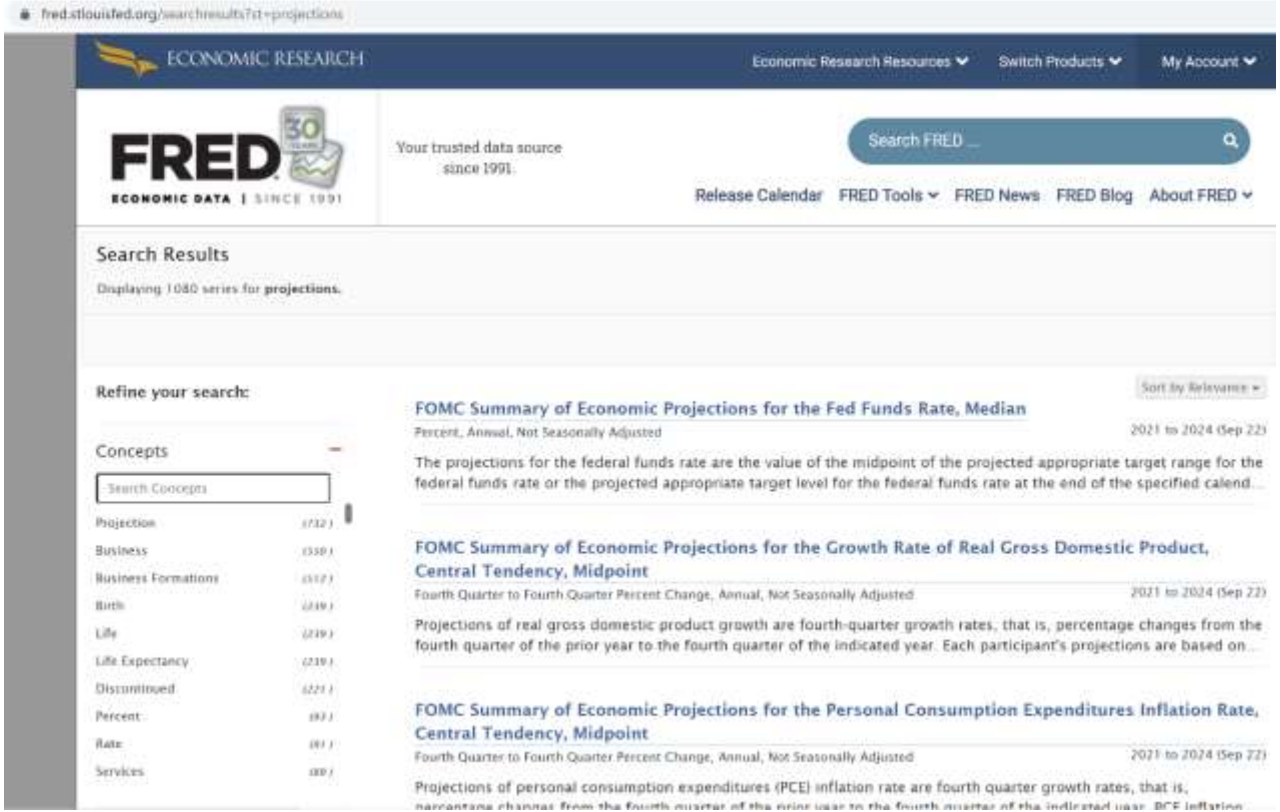
With

- GROUP BY
- HAVING

```
SELECT customerID,
    sum(purchaseAmount) as amountSpent
FROM transactionsTable
WHERE year(transactionDate)=2018
GROUP BY customerID
ORDER BY customerID
```

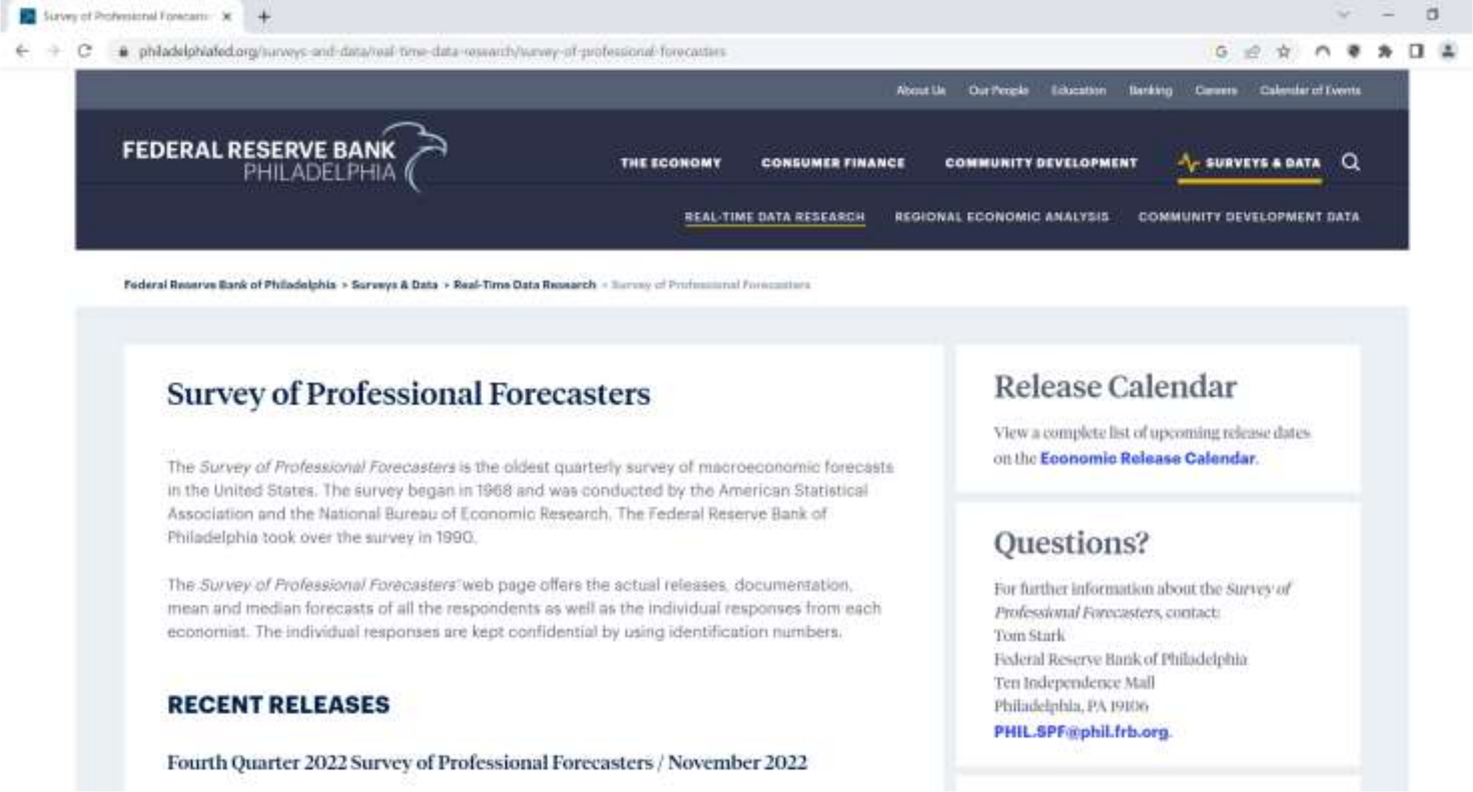
EXTERNAL DRIVERS—MACROECONOMIC

https://fred.stlouisfed.org/searchresults?st=projections#



EXTERNAL DRIVERS—MACROECONOMIC

https://www.philadelphiafed.org/surveys-and-data/real-time-data-research/survey-of-professional-forecasters



EXTERNAL DRIVERS—CUSTOMER-LEVEL

https://www.census.gov/data/data-tools.html

BROWSE BY TOPIC | EXPLORE DATA | LIBRARY | SURVEYS/ PROGRAMS | INFORMATION FOR... | FIND A CODE | ABOUT US

// Census.gov > Data > Data Tools

DATA

- **Data Tools and Apps**
- Datasets
- Errata Notes
- News
- Related Sites
- Software
- Tables
- Training & Workshops
- Visualizations
- < Back to Data

Data Tools and Apps

Find information using interactive applications to get statistics from multiple surveys.

Page 1 of 2 >

Data Tool

2020 Census Address Count Listing Files Viewer

This application was developed to supplement the Address Count Listing files. The files include housing unit and group quarters counts, by census block.

Data Tool

2020 Census Response Rate Map

Are you curious about how many people in your community are responding to the 2020 Census? Stay up to date with a map of response rates from across the U.S.

Data Tool | June 25, 2019

2020 Census: In-Field Address Canvassing (IFAC) Viewer

This map application shows the percentage of basic collection units (BCUs) and housing units (HUs) in the Self-Response Type of Enumeration Areas.

Data Tool

2020 Census: Mail Contact Strategies Viewer

This data tool was developed so that communities can plan for the mailings that their area will receive when self-response for the

NOTE:
Sample scripts at
http://ceresanalytics.com/R_for_IBF_BootCamp/

SCRIPTING

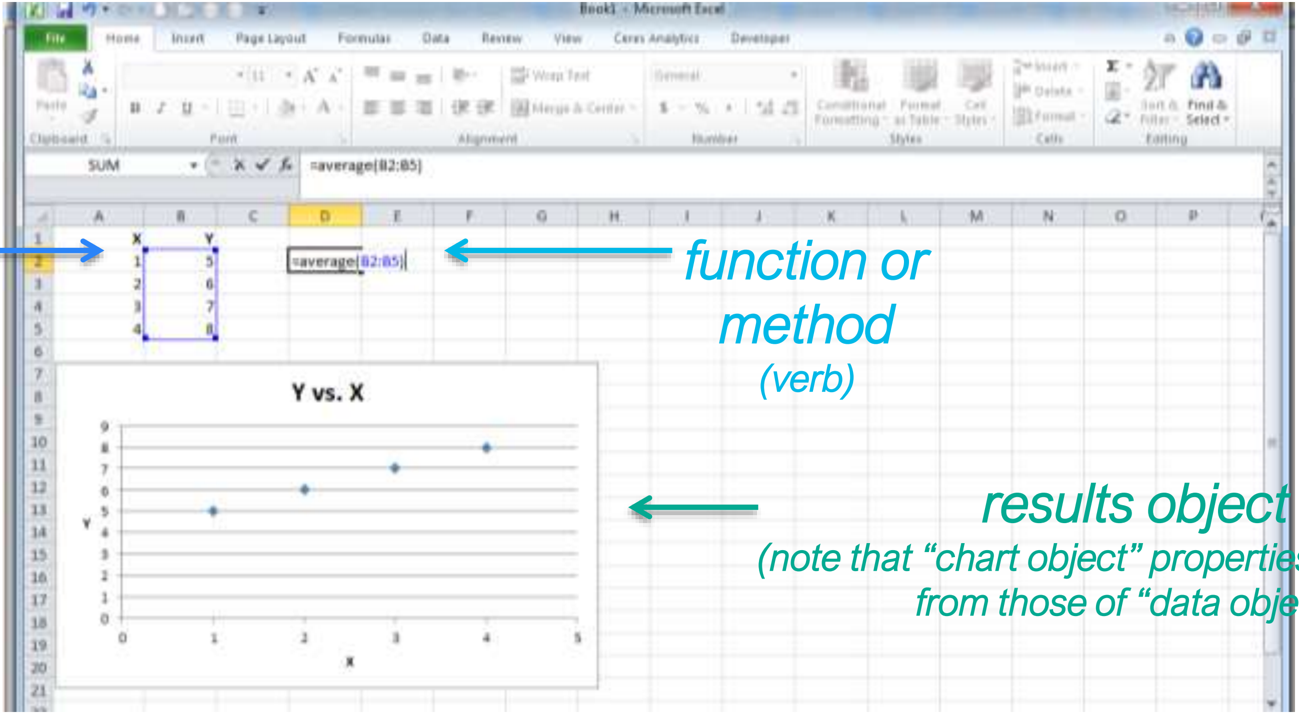
The basics, survey-style

3

WHAT'S THE LOGIC OF OBJECT-ORIENTED PROGRAMMING?

(YOU'VE SEEN IT BEFORE)

data object
(noun)



function or method
(verb)

results object
(note that "chart object" properties are different from those of "data object")

BACKGROUND

SCRIPTING IN ANY CONTEXT

What’s truly different

- Where to get your scripting tools
- Working environment
- How to install specialized libraries

Examples of similar functionality*:

- Variable types (though classes can differ by language)
- Documenting your code
- Variable Declaration and Assignment
- Arrays
- Importing/exporting data
- Repeating tasks (loops)
- If/then logic

* *with exceptions or limitations for SQL versions*

WHAT'S TRULY DIFFERENT

	R	Python	VBA (for Excel)	SQL
Where to get	https://cloud.r-project.org/ Can add RStudio IDE from https://posit.co/downloads/	https://www.python.org/downloads/ includes IDLE *(alternative distributions include Anaconda)*	Included, expose through: Options/Show tab/Developer	Largely dependent on underlying RDBMS
Working environment	• R GUI • Rstudio • R at prompt • Notebook apps (e.g., Jupyter with R)	• Anaconda • Other apps (e.g., Spyder) • Python at prompt • Notebook apps (e.g., Jupyter)	• Menu Developer/Visual Basic or • Alt+F11	
How to install specialized libraries	Menu "Packages" • Set CRAN mirror • Install package(s)	pip command to install libraries or modules	Tools/References (from VBA interface, e.g. Alt+F11)	

WHERE TO START? (SUGGESTIONS)

	R	Python	VBA (for Excel)	SQL
Official reference(s):	• https://cran.r-project.org/manuals.html	https://docs.python.org/3/ (note version selection at top of page)	• https://learn.microsoft.com/en-us/office/vba/library-reference/concepts/getting-started-with-vba-in-office • https://learn.microsoft.com/en-us/office/vba/api/overview/excel	RDBMS (or vendor)-specific
Other useful references:	• https://www.w3schools.com/r/ • https://cran.r-project.org/doc/contrib/Short-refcard.pdf	• https://www.w3schools.com/python/default.asp • https://www.cs.put.poznan.pl/csobaniec/software/python/py-qrc.html	• https://www.tutorialspoint.com/vba/index.htm	https://www.w3schools.com/sql/sql_quickref.asp
Key libraries for forecasting	• stats (in base R) • forecast (time series) • See list at *https://cran.r-project.org/web/packages/available_packages_by_name.html*	• pandas (data analysis) • NumPy (installs automatically w/Pandas) • Matplotlib (plotting) • scikit-learn (machine learning)	• Depends on external resources • Commercial software available to automate Excel without scripting	N/A

VARIABLE TYPES

	R	Python	VBA (for Excel)	SQL (as MS SQL Server)
Alphanumeric text	character	str (more in six library)	String	CHAR(32) VARCHAR(16) NVARCHAR(8)
Integer	integer	int	Integer Long	INT BIGINT
Real numbers, non-integer	numeric	• float • fractions.Fraction • decimal.Decimal(user-defined precision)	Single Double	FLOAT
Boolean	logical	bool	Boolean	Bit *(Oracle and MySql have Boolean)*
Category	factor	category *(pandas library)*	--	--

DOCUMENTING YOUR CODE

R	Python
Line is ‘#’	Line is ‘#’
`# see https://www.programiz.com/r/comments`	`# this is a comment line`
VBA (Excel)	**SQL *(as MS SQL Server)***
Line is “’” (single quote) or ‘Rem’ (typeface will turn green in VBA editor)	Line is ‘—” (two hypens), block is /*…*/
`'a VBA comment` `Rem another vba comment`	`--one-line comment` `/* multiple-line` `comment */`

Note: when you model your code after something you learn on the web, include the URL in a comment line!

VARIABLE DECLARATION, ASSIGNMENT AND TYPE

R

- Declaration not required
- Assignment by “<-” or “=“ (with “<-” preferred)

Note: Equality tested by “==“

```
a<-1 #for integer, a<-as.integer(1)
class(a)
print(a==1)
```

Python

- Declaration not required, but useful for arrays
- Assignment by “=“

Note: equality tested by “==“, with “is” testing pointer location

```
a=1
type(a)
print(a==1)
```

VBA (Excel)

- Declaration is optional unless top of module statement is ‘Option Explicit’
- Assignment by “=“, preceded by “Set” for objects (as opposed to numbers or strings)

Note: Equality tested by “==“

```
Dim a as integer
a=1
Debug.Print (a = 1) 'see Immediate window
```

SQL *(as MS SQL Server)*

- Declaration is required, with variable name beginning with ‘@’ and followed by type
- Assignment by “=“, preceded by “Set”

Note: Equality tested by “==“

```
DECLARE @a INT
SET @a=1
if (@a=1) print('True') else print('False')
--see Messages window
```

VECTORS/1-DIMENSIONAL ARRAYS

R

- Generally 1-based (first element is [1])
- Lists, types with subscript
- (naming elements mimics key/value pairs)

```
myVec<-1:10
myVec
```

Python

- Generally 0-based (first element is [0])
- NumPy library: array; array module: array;
- (also tuples, sets, list, dictionaries for key/value pairs)

```
import numpy
myVec=1+numpy.arange(9) #range defaults to 0:9
myVec
```

VBA (Excel)

- Generally 0-based (first element is [0] unless 1-based is specified)
- Declared by subscript with type

```
Dim i, myVec(1 To 10) As Integer
For i = 1 To 10
   myVec(i) = I
   'assuming you want to populate column 1
   ActiveSheet.Cells(i, 1) = myVec(i)
Next i
```

SQL *(as MS SQL Server)*

- No array functionality
- Work-arounds include temporary tables with automatic row # assignment, parsing of comma-delimited string within a loop

N/A

WORKING WITH/IMPORTING DATA

R (from CSV)	Python (from CSV)
• read.csv, read.csv2, read.delim, read.table • write.csv, write.csv2, write, write.table	• csv module has csv.reader, csv.writer • pandas library has pandas.read_csv, pandas.DataFrame.to_csv
VBA (Excel)	**SQL *(as MS SQL Server)***
• Workbook.open method • Define a “range” object and refer to cells as needed	• Can read by defining table, truncate it, and apply BULK INSERT • Can use namespaces System.IO, System.Data.SqlClient

END OF LINE/CONTINUATION OF LINE

R	Python
Logical completion of command (can span lines) OR ‘;’	Backslash (\) signals line continuation character, as do implicit signals of open parentheses, brackets, etc., with indentation
a=1 if (a > 0 & a <5) print('aha')	If (a>0 and a >5): print(‘aha’) *#unindented line follows*
VBA (Excel)	**SQL *(as MS SQL Server)***
Space and underscore (“ _”) signal line continuation; otherwise, code ends on line	Logical completion of command (can span lines) OR ‘;’
a = 1 If (a > 0 & a _ < 5) Then Debug.Print (_ "aha")	DECLARE @a int SET @a=1 if @a > 0 and @a <5 print('aha')

LOOP THROUGH A RANGE OF VALUES

(E.G., DO A TASK 10 TIMES)

R

- Note position of “k”, with “in” to denote its range of values and colon (“:”) to separate its limits
- Code to execute is within braces {…}

for (k in 1:10) {
  code lines or
  commands ending in “;”
}

Python

- Note position of k, with “range” to indicate that a range follows, with a comma (“,”) to separate its limits
- Code to execute is defined by indentation (by spaces, not tab)

for k in range(1,10):
    space-indented line 1
    space-indented line 2

VBA (Excel)

- Note the lack of parentheses with “k”, with “=“ indicating that a range follows, and “to” separating its limits
- Code to execute is within “for” and “next” lines

For k = 1 to 10
    code line 1
    code line 2
Next k

SQL *(as __)*

- Initialize k before execution of loop, then increment within the loop
- Code to execute is within “begin” and “end” statements

DECLARE @K as INT =1
While K <=10
BEGIN
    code lines
    K=K+1
END

IF-THEN LOGIC

(CONDITIONAL EXECUTION OF A BLOCK OF CODE)

R

- Parenthesized condition to test
- Conditionally executed code in braces

```
if (condition) {
   code line 1
   code line 2
}
```

Python

- Parenthesized condition to test, followed by colon (":")
- Conditionally executed code, space-indented uniformly

```
if (condition):
  space-indented code line 1
  space-indented code line 2
```

VBA

- Parenthesized condition to test, followed by "Then"
- Conditionally executed code concludes with "End If"

```
If (condition) Then
   code line 1
   code line 2
End If
```

SQL *(as MS SQL Server)*

- Parentheses are optional for condition to test
- Conditionally executed code between "BEGIN" and "END"

```
If condition
BEGIN
   code line 1
   code line 2
END
```

DESCRIPTIVE & PREDICTIVE TECHNIQUES

TODAY'S APPROACH TO TECHNIQUES

Big Data Perspective

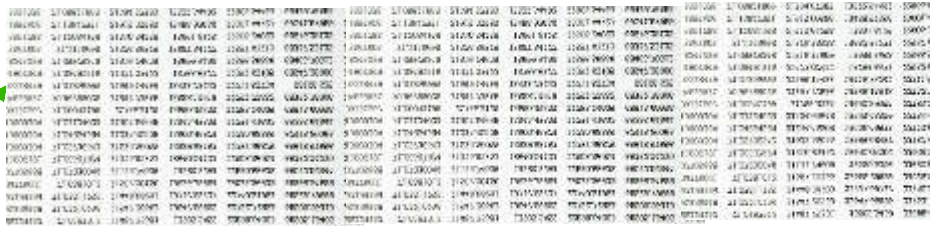
4 5

Row-wise and Column-wise Operations

DATA TABLES AS "LEAST COMMON DENOMINATOR"

4 5

Tables might be very wide…



… or very tall

ROW-WISE VS. COLUMN-WISE OPERATIONS

Date	SKU	Location	Demand
202111	A1234	101	36
202112	A1234	101	95
202201	A1234	101	39
202202	A1234	101	58
202203	A1234	101	80
202204	A1234	101	21
202205	A1234	101	48
202206	A1234	101	83
202207	A1234	101	73
202208	A1234	101	13
202209	A1234	101	55
202210	A1234	101	11
202111	A1234	102	47
202112	A1234	102	81
202201	A1234	102	68
202202	A1234	102	55
202203	A1234	102	26
202204	A1234	102	28
202205	A1234	102	64
202206	A1234	102	40
202207	A1234	102	81
202208	A1234	102	18
202209	A1234	102	10
202210	A1234	102	12
202111	A5678	101	24
202112	A5678	101	13
202201	A5678	101	64

4 5

Date	A1234	A5678	B9012
202111	83	130	64
202112	176	124	199
202201	107	147	162
202202	113	131	157
202203	106	195	201
202204	49	208	91
202205	112	196	73
202206	123	242	138
202207	154	139	205
202208	31	56	103
202209	65	63	287
202210	23	117	218

Time series data

- Usually in a column, with one row per time period
- Might be found as rows identified by combinations of sku (column), location (column) and time period (column), then demand (column)
- Might be found aggregated, with rows identified by time period, and values named by SKU, product class, business entity, etc.

Descriptive and Predictive Techniques

ROW-WISE VS. COLUMN-WISE OPERATIONS

4 5

1	Customer	Mon_AvgSales	Freq_NumPurchPastYear	Recency_TimeSinceLastPchs
2	10000001	41	52	41
3	10000002	106	6	202
4	10000003	105	6	189
5	10000004	43	53	92
6	10000005	71	26	207
7	10000006	69	26	137
8	10000007	72	27	249
9	10000008	70	26	174
10	10000009	43	53	120
11	10000010	49	56	337
12	10000011	46	54	208
13	10000012	44	53	134
14	10000013	41	52	20
15	10000014	45	54	186
16	10000015	67	25	58
17	10000016	68	25	124
18	10000017	69	25	133
19	10000018	71	26	224
20	10000019	100	4	2
21	10000020	108	7	303
22	10000021	67	25	83
23	10000022	74	28	332
24	10000023	44	54	149
25	10000024	71	26	217
26	10000025	72	27	241
27	10000026	68	25	109
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Customer-level data

- Rows are observations (customers)
- Columns are measures that pertain to the customer, e.g.:
  - Monetary: how much they spend per month
  - Frequency: number of purchases in past year
  - Recency: how many days since last purchasea

ROW-WISE AND COLUMN-WISE OPERATIONS

<table>
<tr><th>Where</th><th>Why</th><th>Transactions Data</th><th>Customer Attribute Records</th></tr>
<tr><td>Row-wise</td><td>Obtain Customer-level Measures
(Big Data too tall)

Obtain product class aggregates for forecasting</td><td>Aggregate:
• Sum (Σ)
• Average (μ)
• Coefficient of Variation (σ/μ)</td><td></td></tr>
<tr><td>Column-wise</td><td>Obtain predictors
(Big Data too wide)</td><td></td><td>Select predictors
Combine predictors</td></tr>
<tr><td>Row-wise operations</td><td>Group customers, products
(Big Data too tall)</td><td colspan="2">• Find product groups that forecast well because they're alike in patterns
• Find groups of customers for more precise group-level forecasts</td></tr>
<tr><td>Row-wise</td><td>Construct test/training/validation datasets
(Big Data FORTUNATELY tall)</td><td colspan="2">• Sample records to balance attributes among datasets</td></tr>
</table>

SELECT PREDICTORS

Goal	Statistics	Machine Learning
Evaluate each independent variable (predictor, feature), either: • by itself • vs. other predictors • vs. the dependent variable	EDA • Univariate distributions • Correlations (continuous) • Chi-square test ("correlation among categories")	• "Filter Methods"
Search for best set of predictors	Automated Variable Selection • Stepwise Regression • Backwards Stepwise Regression • All-possible subsets regression	"Wrapper Methods" • Best-first search • Recursive Feature Elimination • Heuristics
Learn which features improve model accuracy the most		"Embedded Methods" • Regularization (penalization) which constrains the optimization of an algorithm toward less complexity (LASSO, Ridge Regression, Elastic Net)

Machine Learning per https://machinelearningmastery.com/an-introduction-to-feature-selection/

EACH TECHNIQUE

What it is/how it works

Statistical Methods | Machine Learning

How to evaluate results

How to apply results

Example

with link to R program

DESCRIPTIVE TECHNIQUES

Cluster Analysis

4

CLUSTER ANALYSIS

4

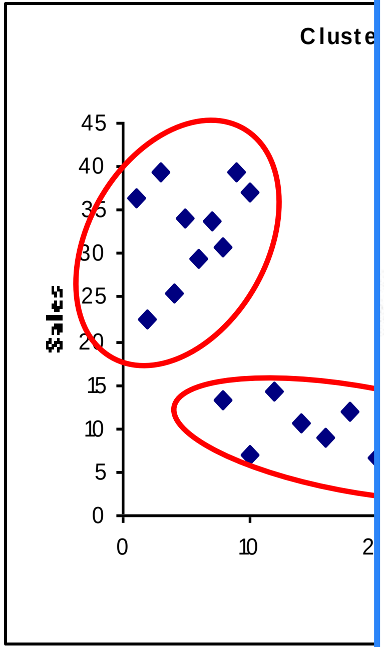
CLUSTER ANALYSIS

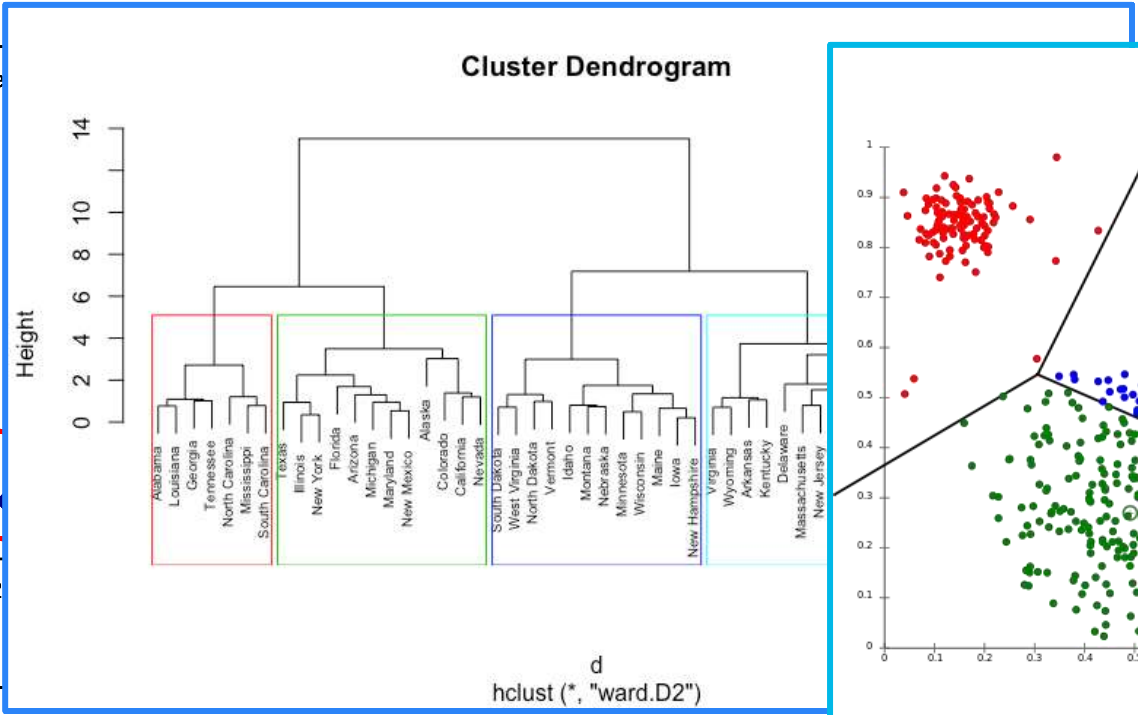
- What it is/how it works
  - Descriptive technique that—through either statistics or machine learning—finds groups in data for which observations are as similar as possible, while groups are as different as possible
  - Statistics vs. Machine Learning—different approaches to same task
- How to evaluate your results
  - Key measures, visualization (color the clusters)
- How to apply your results
  - Name them and learn from their profiles
  - Aggregate by segment to create class-level history and forecasts
- Example
  - Script in R (for today)

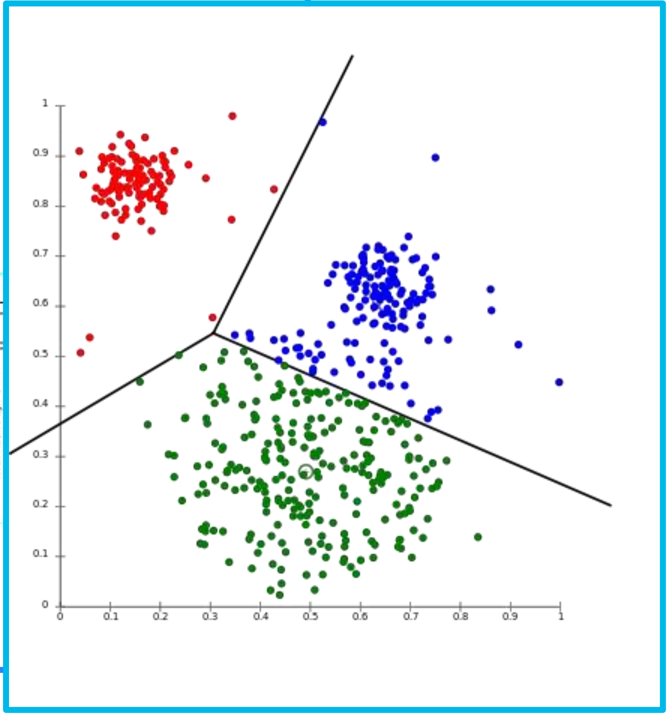
CLUSTER ANALYSIS

AGGLOMERATIVE (HIERARCHICAL) VS. PARTITIONING METHODS

https://stats.stackexchange.com/questions/146339/what-does-cluster-size-mean-in-context-of-k-means



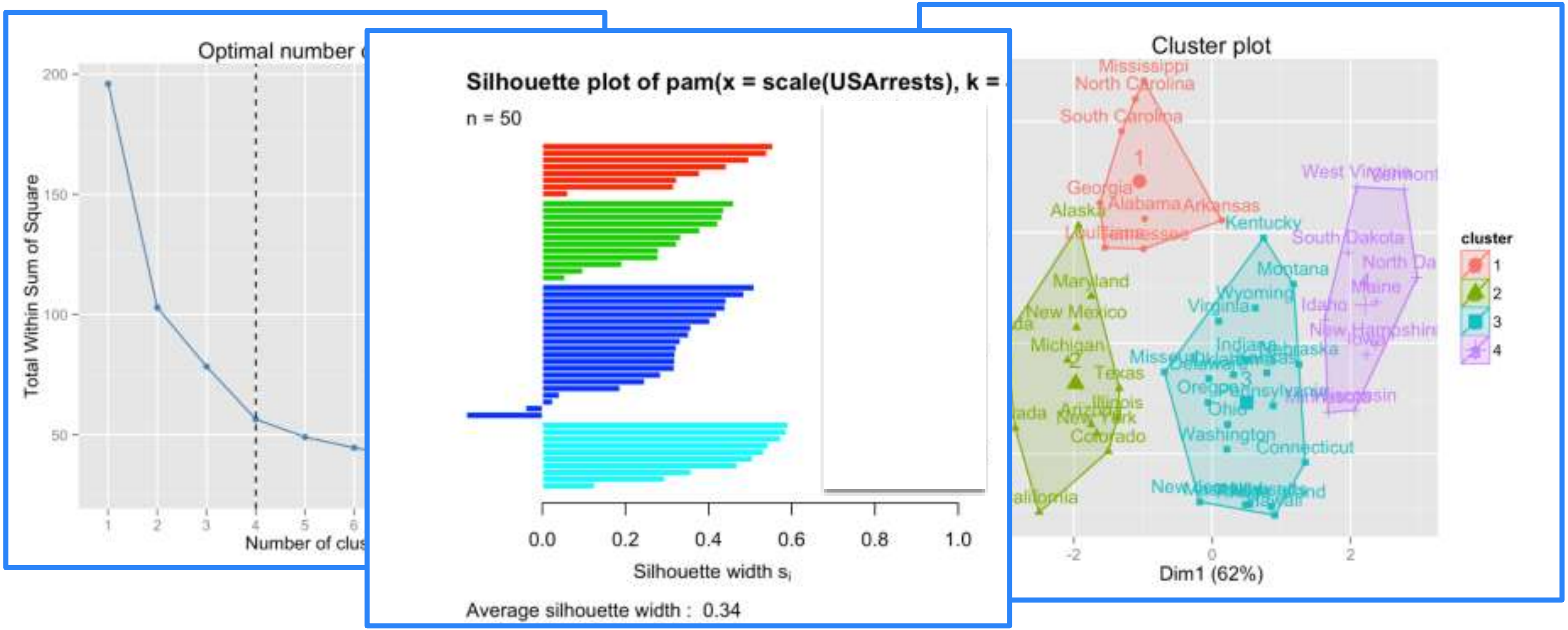




https://uc-r.github.io/hc_clustering

CLUSTER ANALYSIS

EVALUATION OF RESULTS



Source (all, with silhouettes modified): http://www.sthda.com/english/wiki/print.php?id=236

CLUSTER ANALYSIS

PROCESS

Concept	Considerations
Identify the purpose of the cluster analysis	• To construct new aggregates for forecasting? • To learn about/isolate group(s) from customer base? • To learn about/isolate group(s) from products?
Identify what will be clustered	• Identify **customer** segments? • Segregate **products** by key sales characteristics? • Characterize unprofitable **time** periods?
Assemble data	• One **row** per observation (e.g., customer, product) • **Columns** are measures that will define groups (e.g., RFM, inventory and profit metrics)

CLUSTER ANALYSIS

E.G., AGGREGATE TO CLASS-LEVEL BY ASSIGNED CLUSTER

Customer	Mon_AvgSales	Freq_NumPurchPastYear	Recency_TimeSinceLastPchs	OldGroupNum	OldGroup	NewCluster	NewClusterName
10000001	41	52	41	1	SouthEast	1	ThirstyThrifties
10000002	106	6	202	1	SouthEast	2	SippingSnobs
10000003	105	6	189	1	SouthEast	2	SippingSnobs
10000004	43	53	92	1	SouthEast	1	ThirstyThrifties
10000005	71	26	207	1	SouthEast	3	ImpressiveImbibers
10000006	69	26	137	1	SouthEast	3	ImpressiveImbibers
10000007	72	27	249	1	SouthEast	3	ImpressiveImbibers
10000008	70	26	174	2	NorthEast	3	ImpressiveImbibers
10000009	43	53	120	2	NorthEast	1	ThirstyThrifties
10000010	49	56	337	2	NorthEast	1	ThirstyThrifties
10000011	46	54	208	2	NorthEast	1	ThirstyThrifties
10000012	44	53	134	2	NorthEast	1	ThirstyThrifties
10000013	41	52	20	2	NorthEast	1	ThirstyThrifties
10000014	45	54	186	2	NorthEast	1	ThirstyThrifties
10000015	67	25	58	2	NorthEast	3	ImpressiveImbibers
10000016	68	25	124	2	NorthEast	3	ImpressiveImbibers
10000017	69	25	133	2	NorthEast	3	ImpressiveImbibers
10000018	71	26	224	2	NorthEast	3	ImpressiveImbibers
10000019	100	4	2	2	NorthEast	2	SippingSnobs
10000020	108	7	303	2	NorthEast	2	SippingSnobs
10000021	67	25	83	2	NorthEast	3	ImpressiveImbibers

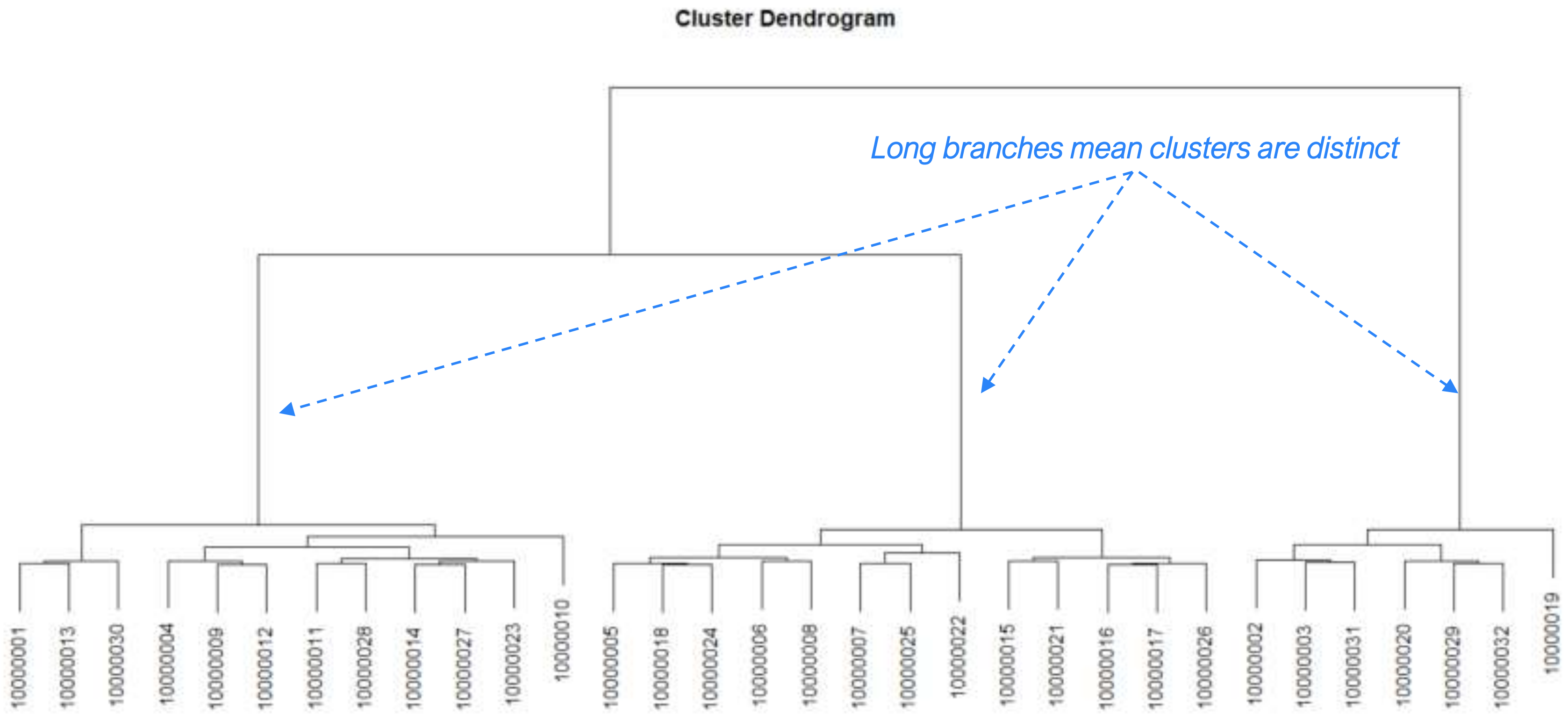
DEMO: CLUSTER ANALYSIS

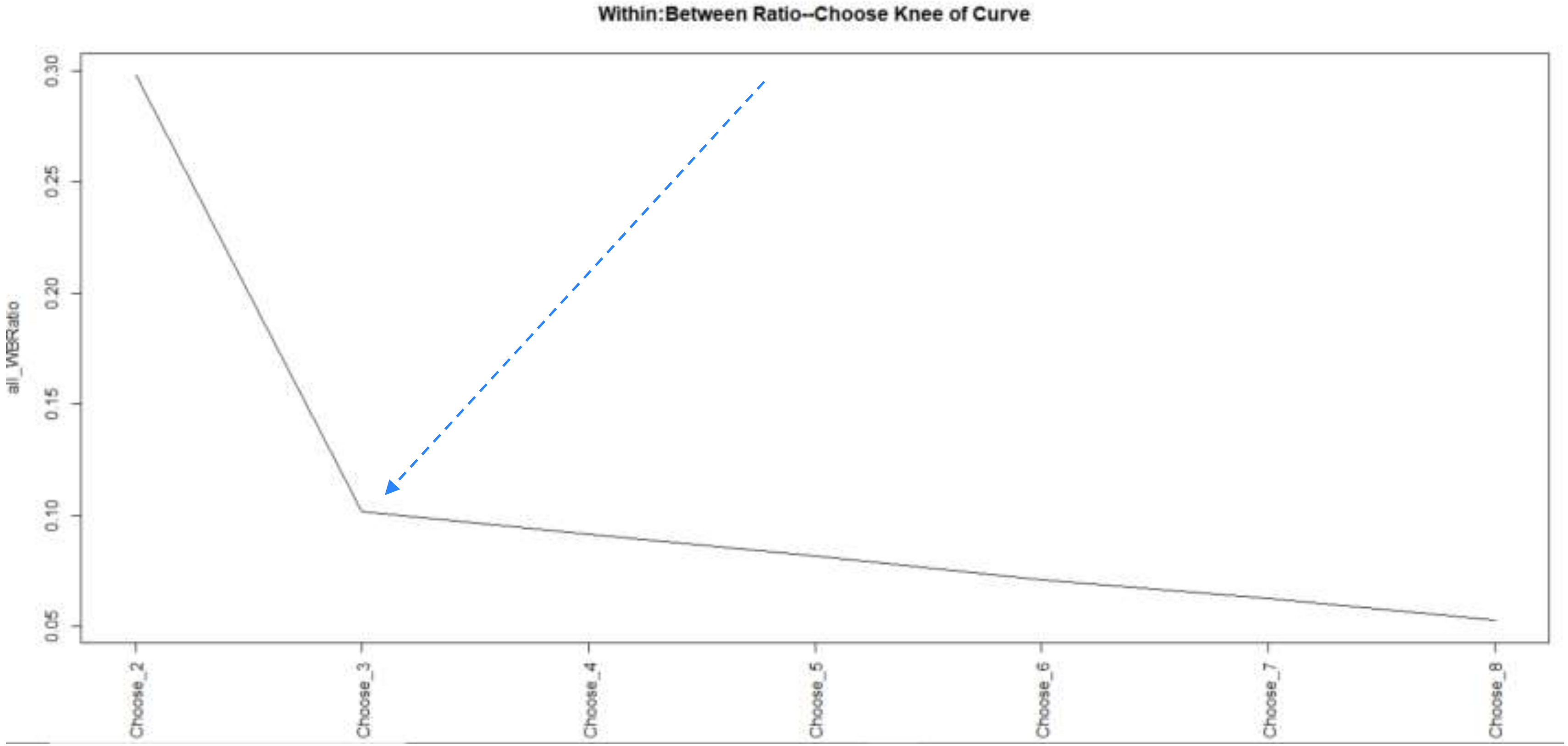
HTTP://CERESANALYTICS.COM/R_FOR_IBF_BOOTCAMP/IBF_CLUSTERS.R

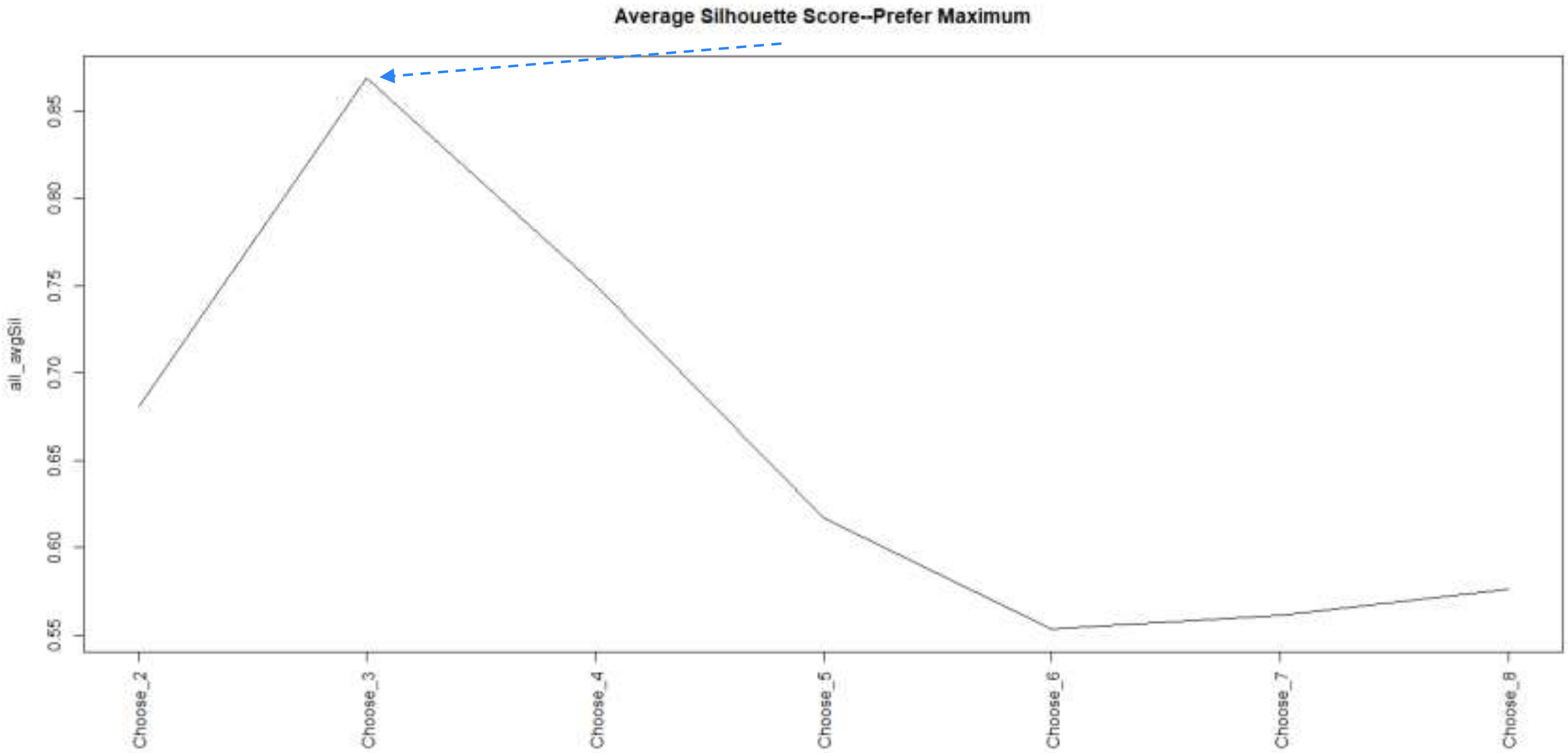
WEB APP: HTTPS://SBTHESIS.COM/SHINY_IBF/

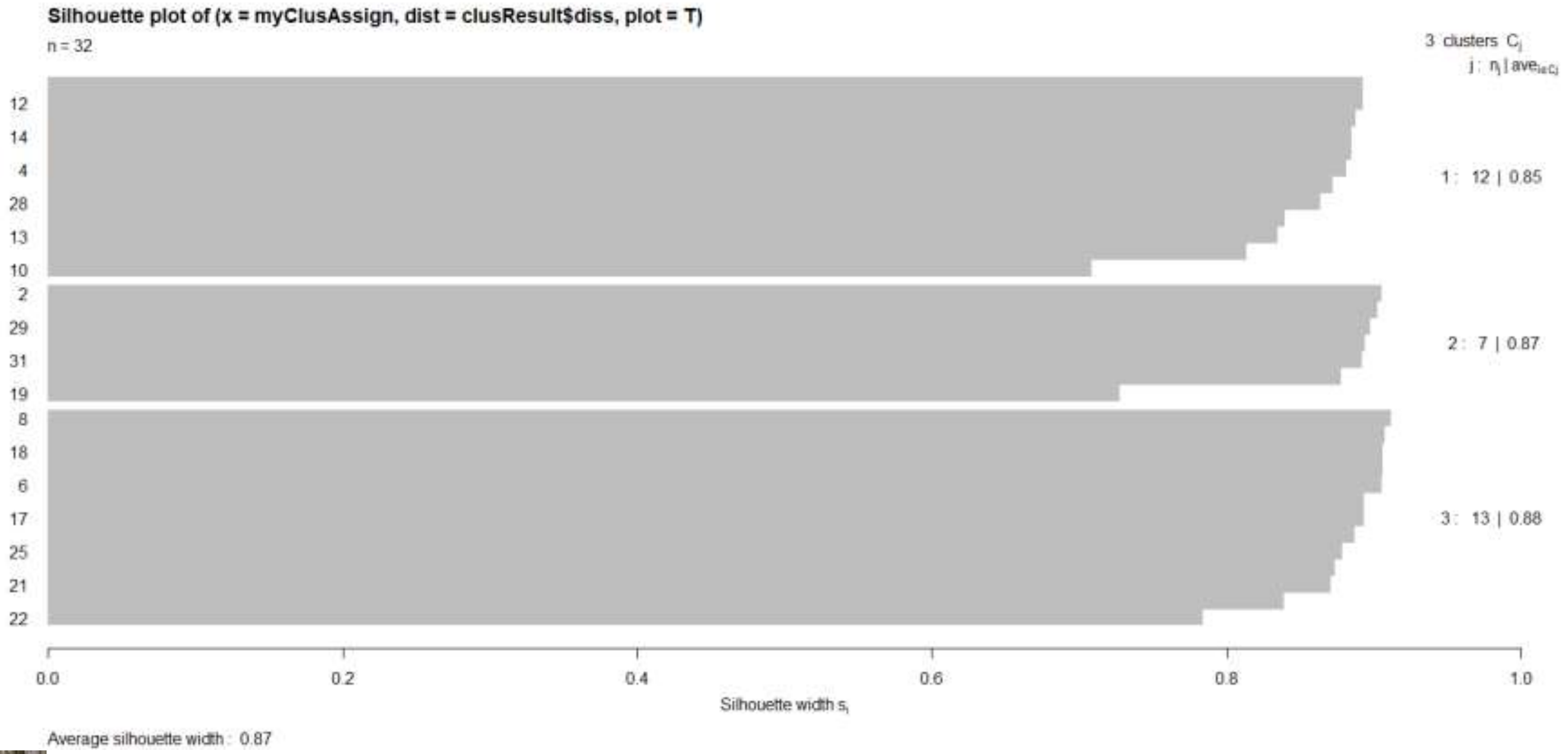
```
> clusterData
```

	CustomerID	Mon_AvgSales	Freq_NumPurchPastYear	Recency_TimeSinceLastPchs	OldGroupNum	OldGroup
1	10000001	22	53	7	1	SouthEast
2	10000002	111	11	32	1	SouthEast
3	10000003	110	10	32	1	SouthEast
4	10000004	25	55	8	1	SouthEast
5	10000005	71	31	16	1	SouthEast
6	10000006	68	29	16	1	SouthEast
7	10000007	74	32	17	1	SouthEast
8	10000008	70	30	16	2	NorthEast
9	10000009	27	56	8	2	NorthEast
10	10000010	38	63	11	2	NorthEast
11	10000011	31	59	9	2	NorthEast
12	10000012	27	56	8	2	NorthEast
13	10000013	21	53	7	2	NorthEast
14	10000014	30	58	9	2	NorthEast
15	10000015	63	26	15	2	NorthEast
16	10000016	67	28	15	2	NorthEast
17	10000017	67	28	15	2	NorthEast
18	10000018	72	31	16	2	NorthEast
19	10000019	100	4	30	2	NorthEast
20	10000020	117	14	33	2	NorthEast
21	10000021	65	27	15	2	NorthEast
22	10000022	78	35	18	3	Central
23	10000023	28	57	9	3	Central
24	10000024	72	31	16	3	Central
25	10000025	73	32	17	3	Central
26	10000026	66	28	15	3	Central
27	10000027	30	58	9	3	Central
28	10000028	32	59	9	3	Central
29	10000029	115	13	33	3	Central
30	10000030	20	52	7	3	Central
31	10000031	109	9	32	3	Central
32	10000032	116	13	33	3	Central


Cluster Dendrogram
Long branches mean clusters are distinct
10000001
10000013
10000030
10000004
10000009
10000012
10000011
10000028
10000014
10000027
10000023
10000010
10000005
10000018
10000024
10000006
10000008
10000007
10000025
10000022
10000015
10000021
10000016
10000017
10000026
10000002
10000003
10000031
10000020
10000029
10000032
10000019


Within:Between Ratio--Choose Knee of Curve
all_WBRatio
0.30
0.25
0.20
0.15
0.10
0.05
Choose_2
Choose_3
Choose_4
Choose_5
Choose_6
Choose_7
Choose_8


Average Silhouette Score--Prefer Maximum
all_avgSil
0.55
0.60
0.65
0.70
0.75
0.80
0.85
Choose_2
Choose_3
Choose_4
Choose_5
Choose_6
Choose_7
Choose_8


Silhouette plot of (x = myClusAssign, dist = clusResult$diss, plot = T)
n = 32
3 clusters C
j : n | ave
12
14
4
28
13
10
2
29
31
19
8
18
6
17
25
21
22
1 : 12 | 0.85
2 : 7 | 0.87
3 : 13 | 0.88
0.0
0.2
0.4
0.6
0.8
1.0
Silhouette width s
Average silhouette width : 0.87

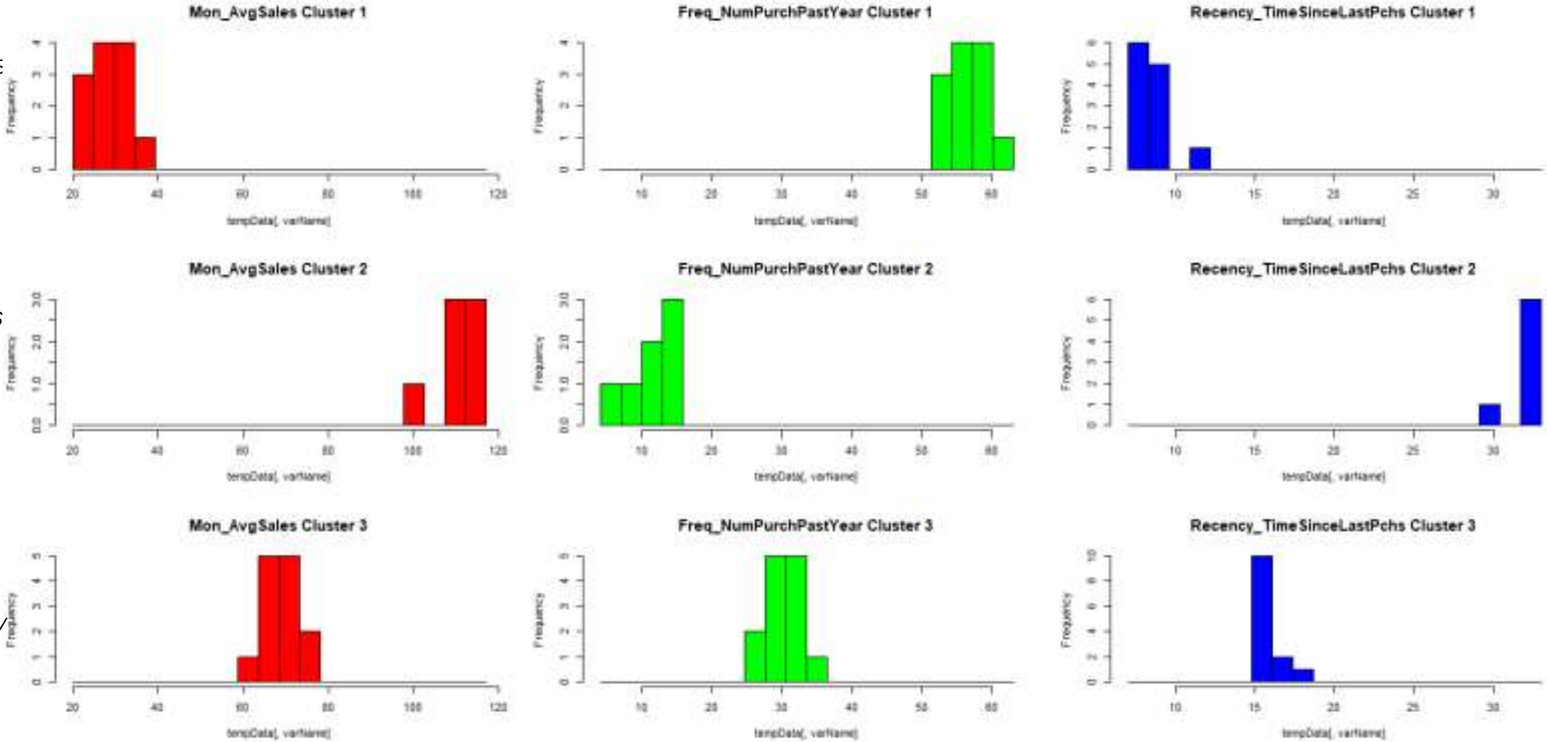
The cluster assignment can be used like any other data element…

```
> head(afterClusterData)
  CustomerID Mon_AvgSales Freq_NumPurchPastYear Recency_TimeSinceLastPchs OldGroupNum  OldGroup myClusAssign
1   10000001           22                    53                         7           1 SouthEast            1
2   10000002          111                    11                        32           1 SouthEast            2
3   10000003          110                    10                        32           1 SouthEast            2
4   10000004           25                    55                         8           1 SouthEast            1
5   10000005           71                    31                        16           1 SouthEast            3
6   10000006           68                    29                        16           1 SouthEast            3
> |
```

Cluster 1
- *Low average sales*
- *High frequency*
- *Low recency*

Cluster 2
- *High average sales*
- *Low frequency*
- *High recency*

Cluster 3
- *Medium average sales*
- *Medium frequency*
- *Medium recency*



PRINCIPAL COMPONENTS

4

PRINCIPAL COMPONENTS

- What it is/how it works
  - An algorithm by which, after a pool of eligible X (explanatory) variables is identified, they are weighted and combined into new explanatory variables ("Z's")
  - Statistics vs. Machine Learning—although a number of sources will classify PCA as machine learning, the PC's are computed directly through an algorithm known as "finding the root of a matrix", leading to an "eigen decomposition"
- How to evaluate your results
  - "Knee of curve": cumulative eigenvalues
- How to apply your results
  - Use top (or top few) component(s) as composite variables
  - Note which variables contribute most heavily to each component
- Example
  - Excel VBA script

PRINCIPAL COMPONENTS

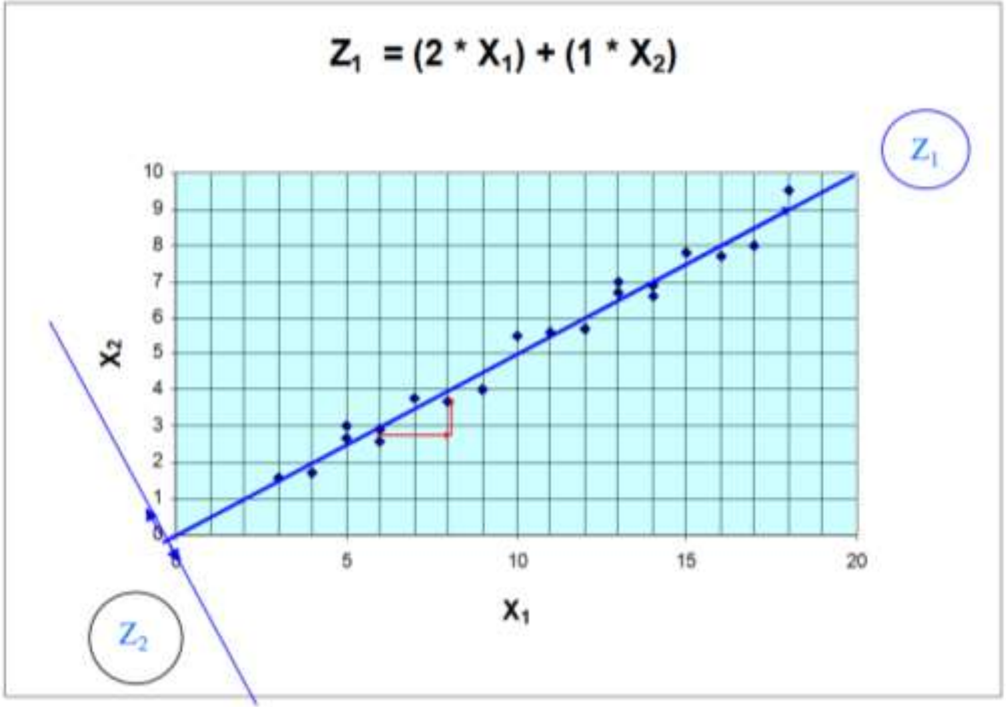
THE PROCESS

- The computer constructs the first Z--the weighted combination of X's which captures the maximum amount of variation within the sample
- Logic (why you use them to forecast): variation in the X's must help to explain variation in Y
- *after all, If Y didn't vary, forecasting would be easy: your forecast would be a constant value, such as Y's average*
- *but there's no guarantee…*
- Subsequent combinations, each capturing maximum movement, while remaining uncorrelated with other new Z's, are created
- Usually the first three to five new Z's capture most of the sample variation

https://uc-r.github.io/hc_clustering

PRINCIPAL COMPONENTS

- In this simple example based on two predictors (X_1 and X_2), Z_1 captures most of the variation in the two X's, along with their correlation
- In more advanced applications, factor rotation re-expresses the loading of X_1 and X_2 on the new axes



https://uc-r.github.io/hc_clustering

PRINCIPAL COMPONENTS

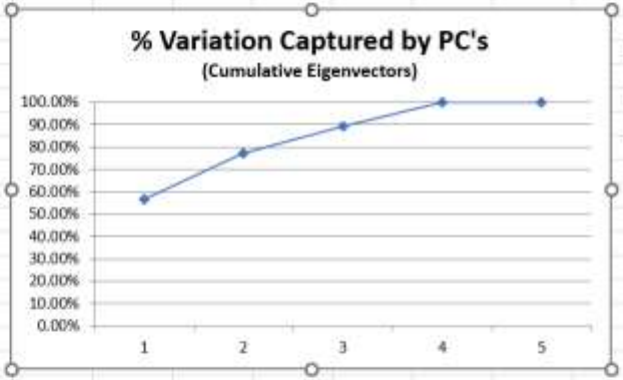
EVALUATION OF RESULTS

	A	B	C	D	E	F
63	2.8429	1.0100	0.6130	0.5261	0.0080	
66	*Eigenvalues incremental*					
68	56.86%	20.20%	12.26%	10.52%	0.16%	
71	*Eigenvalues cumulative*					
73	56.86%	77.06%	89.32%	99.84%	100.00%	
76	*Eigenvectors (Matrix of)*					
78	PcorrfromFull1	PcorrfromFull2	PcorrfromFull3	PcorrfromFull4	PcorrfromFull5	
79	0.4253	-0.1061	0.8349	-0.2984	-0.1474	JnempPct
80	-0.4349	-0.1895	0.4706	0.7427	0.0434	CPI_U_Pct
81	-0.5749	-0.0080	0.0440	-0.3227	-0.7506	RGDP
82	-0.5430	-0.0451	0.2063	-0.4978	0.6424	Time
83	-0.0681	0.9751	0.1922	0.0862	0.0160	Q4
86	*Derived from Input Data:*					

How much variation is "packed" into each new Z

How few PC's can you use to capture variation?

How are original X's weighted in new Z's?



PRINCIPAL COMPONENTS

USE ONE OR MORE PC'S ("Z'S") IN PLACE OF EXPLANATORY VARIABLES (X'S)

The construction of composite variables, through PCA, is most useful when:

- there are too many explanatory variables and too few observations
- a number of explanatory variables are similar in concept
- multicollinearity confounds use of separate predictors

PRINCIPAL COMPONENTS

USE ONE OR MORE PC'S ("Z'S") IN PLACE OF EXPLANATORY VARIABLES (X'S)

	A	B	C	D	E	F	G	H
1	Year	Quarter	YearQuart	TIME	WASA	EMPL	QTR_4	BLDG
2	1	1	Y01Q1	1	1193.3	87973	0	9392
3	1	2	Y01Q2	2	1217.3	90021	0	13637
4	1	3	Y01Q3	3	1254.4	90120	0	14392
5	1	4	Y01Q4	4	1284.7	91180	1	13301
6	2	1	Y02Q1	5	1319.8	89832	0	10411
7	2	2	Y02Q2	6	1335.1	90668	0	13057
8	2	3	Y02Q3	7	1360.1	89850	0	13621
9	2	4	Y02Q4	8	1411.6	91276	1	13354
10	3	1	Y03Q1	9	1451.7	89964	0	11056
11	3	2	Y03Q2	10	1478.1	91566	0	14559
12	3	3	Y03Q3	11	1512.6	91405	0	14067
13	3	4	Y03Q4	12	1530.6	91691	1	12286
14	4	1	Y04Q1	13	1542.7	89341	0	9658
15	4	2	Y04Q2	14	1563.9	90310	0	13659
16	4	3	Y04Q3	15	1579.8	89324	0	12351
17	4	4	Y04Q4	16	1586	89436	1	12307
18	5	1	Y05Q1	17	1610.7	88815	0	10508
19	5	2	Y05Q2	18	1643.5	89920	0	16287
20	5	3	Y05Q3	19	1671.3	90480	0	16697
21	5	4	Y05Q4	20	1715.4	92367	1	15523
22	6	1	Y06Q1	21	1755.8	91637	0	13540
23	6	2	Y06Q2	22	1793.1	94256	0	19190
24	6	3	Y06Q3	23	1819.5	95021	0	18827
25	6	4	Y06Q4	24	1847.7	96547	1	17215
26	7	1	Y07Q1	25	1882.7	95450	0	13848
27	7	2	Y07Q2	26	1939.8	97657	0	20319
28	7	3	Y07Q3	27	1976	97985	0	20403
29	7	4	Y07Q4	28	2012.8	99383	1	19179

Prin Comp demo CORR | Prin Comp demo VCOV | EDA_timeSeries

→

	A	B	C	D
1	Pcorrdemo1	Pcorrdemo2	Pcorrdemo3	Pcorrdemo4
2	-2.6069	-0.3871	0.3173	0.0238
3	-2.3283	-0.3921	0.5308	-0.0421
4	-2.2157	-0.4021	0.4760	-0.0220
5	-1.8817	1.8829	0.4325	-0.0187
6	-2.0497	-0.4216	0.3068	0.0157
7	-1.8993	-0.4273	0.3636	-0.0301
8	-1.8919	-0.4370	0.1935	-0.0057
9	-1.4916	1.8455	0.1862	0.0284
10	-1.6487	-0.4597	0.0623	0.0715
11	-1.4091	-0.4657	0.2148	0.0241
12	-1.3250	-0.4758	0.1268	0.0471
13	-1.0829	1.8097	-0.0128	0.0499
14	-1.3803	-0.4930	-0.2586	0.0697
15	-1.2085	-0.4993	-0.1876	0.0315
16	-1.2306	-0.5080	-0.3747	0.0430
17	-1.0226	1.7789	-0.5305	0.0279
18	-1.1368	-0.5229	-0.5528	0.0193
19	-0.9349	-0.5307	-0.4705	-0.0001
20	-0.7922	-0.5386	-0.4581	-0.0127
21	-0.3591	1.7458	-0.3994	-0.0075
22	-0.4604	-0.5585	-0.4457	0.0183
23	-0.1079	-0.5645	-0.1638	-0.0389
24	0.0522	-0.5719	-0.1232	-0.0605
25	0.4274	1.7142	-0.1032	-0.0757
26	0.2831	-0.5899	-0.1953	-0.0492
27	0.6260	-0.5994	0.0197	-0.0545
28	0.7590	-0.6089	-0.0040	-0.0433
29	1.1348	1.6757	-0.0062	-0.0375

Prin Comp demo CORR | Prin Comp demo VC

DEMO: PRINCIPAL COMPONENTS

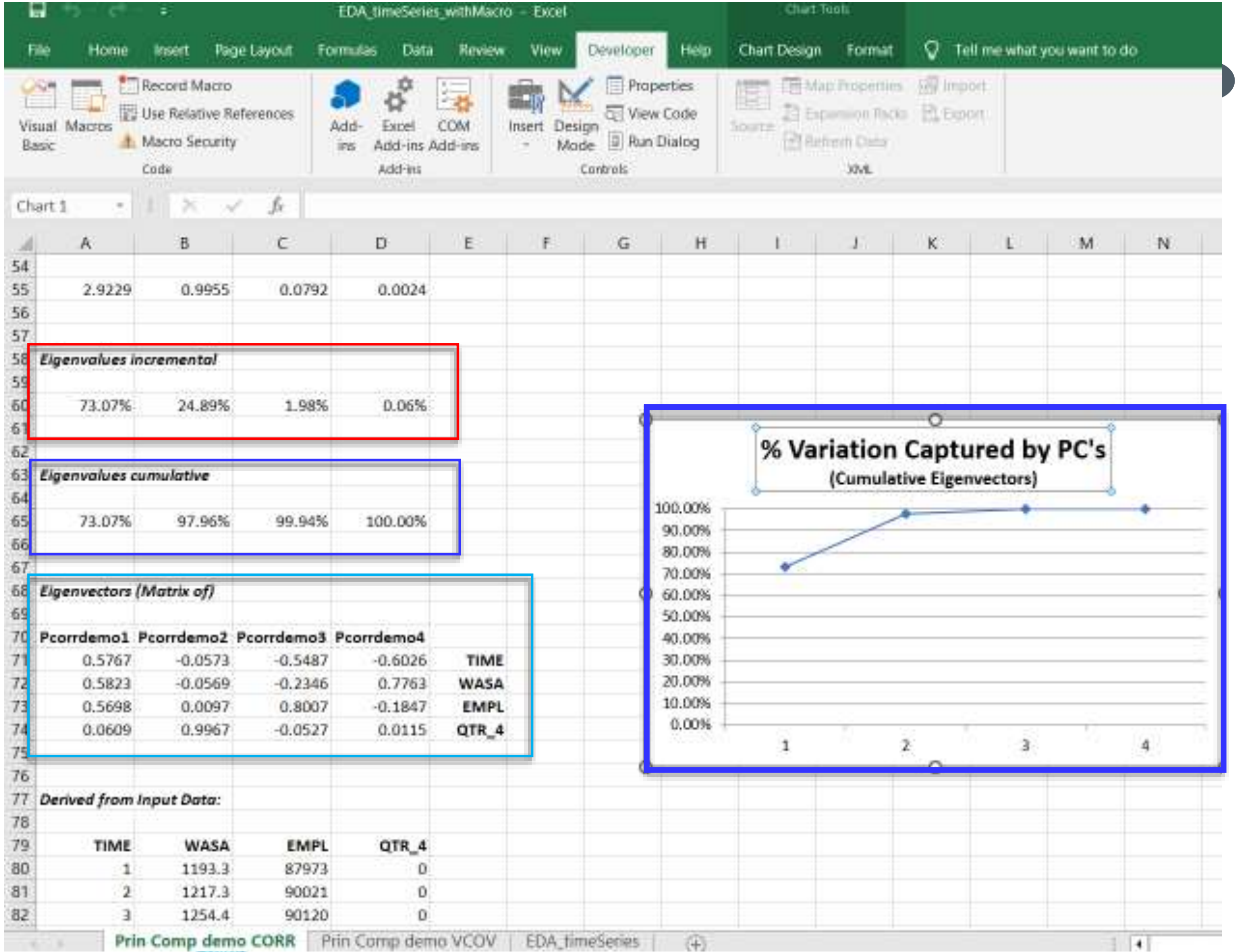
HTTP://CERESANALYTICS.COM/R_FOR_IBF_BOOTCAMP/PRINCIPALCOMPONENTS.BAS

How much variation is packed into each component?

How much variation is packed into first “n” components?

How is each “X” variable weighted in each new “Z” component?

- *E.g., first PC weights Time, WASA, and Empl roughly equally*



	A	B	C	D	E
55	2.9229	0.9955	0.0792	0.0024	
58	***Eigenvalues incremental***				
60	73.07%	24.89%	1.98%	0.06%	
63	***Eigenvalues cumulative***				
65	73.07%	97.96%	99.94%	100.00%	
68	***Eigenvectors (Matrix of)***				
70	**Pcorrdemo1**	**Pcorrdemo2**	**Pcorrdemo3**	**Pcorrdemo4**	
71	0.5767	-0.0573	-0.5487	-0.6026	**TIME**
72	0.5823	-0.0569	-0.2346	0.7763	**WASA**
73	0.5698	0.0097	0.8007	-0.1847	**EMPL**
74	0.0609	0.9967	-0.0527	0.0115	**QTR_4**
77	***Derived from Input Data:***				
79	**TIME**	**WASA**	**EMPL**	**QTR_4**	
80	1	1193.3	87973	0	
81	2	1217.3	90021	0	
82	3	1254.4	90120	0	

Institute of Business Forecasting & Planning

PREDICTIVE TECHNIQUES

Linear regression analysis, Logistic regression analysis, Neural Network, Association Rules

5

LINEAR REGRESSION ANALYSIS

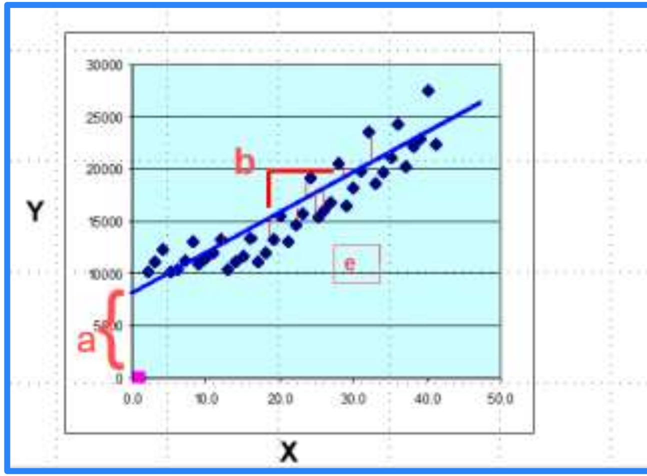
5

LINEAR REGRESSION ANALYSIS

- What it is/how it works
  - Predictive technique that articulates a linear relationship between predictors (independent variables) and the level of an outcome (dependent variable)
  - Statistics vs. Machine Learning—variable selection algorithms, optimization algorithms
- How to evaluate your results
  - R^2, Adjusted R^2, t-statistics, outliers, RMSE, ex-post forecasts
- How to apply your results
  - Extrapolate with equation
- Example
  - R (today), Excel

LINEAR REGRESSION

ESTIMATE AN EQUATION TO PREDICT Y FROM ONE OR MORE X'S

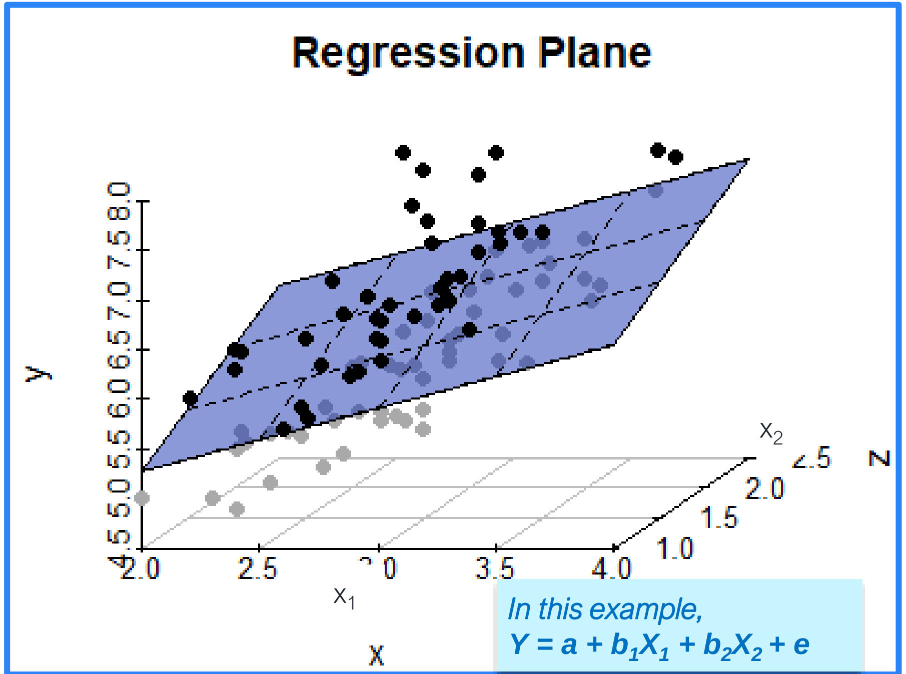


You'll forecast by the equation,

$Y = a + bx$

In this example,

$Y = 8000 + 40X$



In this example,

$Y = a + b_1X_1 + b_2X_2 + e$

modified from https://stackoverflow.com/questions/47344850/scatterplot3d-regression-plane-with-residuals

Calculation of Regression Coefficients

straightforward application of matrix algebra

Least squares estimates in matrix notation

Here's the punchline: the $(k+1) \times 1$ vector containing the estimates of the $(k+1)$ parameters of the regression function can be shown to equal:

$$b = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{bmatrix} = (X'X)^{-1}X'Y$$

where:

- $(X'X)^{-1}$ is the **inverse** of the $X'X$ matrix, and
- X' is the **transpose** of the X matrix.

Source: https://online.stat.psu.edu/stat462/node/132/

LINEAR REGRESSION

A FEW KEY MEASURES

Concept	Measures
Overall Fit	• R^2 and Adjusted R^2 • F –Statistic • RMSE
Each predictor's fit	• Coefficients (+ or -) and their t-statistics • Error patterns
Other (depends on software)	• Standard error of the equation • Durbin-Watson (error tracks itself through time) • More

PREDICTIVE MODEL PERFORMANCE

MEAN AVERAGE PERCENT ERROR (MAPE)

Institute of Business Forecasting & Planning | Events | Training | Assessments / Advisory | Certification | Knowledge | EmployAbility | Job Board

Home > Knowledge > Glossary > MAPE (Mean Absolute Percentage Error)

MAPE (Mean Absolute Percentage Error)

What is MAPE? It is a simple average of absolute percentage errors. The MAPE calculation is as follows:

$$\text{MAPE} = \frac{\sum \frac{|A-F|}{A} \times 100}{N}$$

Here A= Actual, F= Forecast, N= Number of observations, and the vertical bars stand for absolute values.

For intermittent demand (many Actuals=0), consider Mean Absolute Deviation (MAD)= $\frac{\sum |A-F|}{N}$*; see https://robjhyndman.com/papers/foresight.pdf*

LINEAR REGRESSION

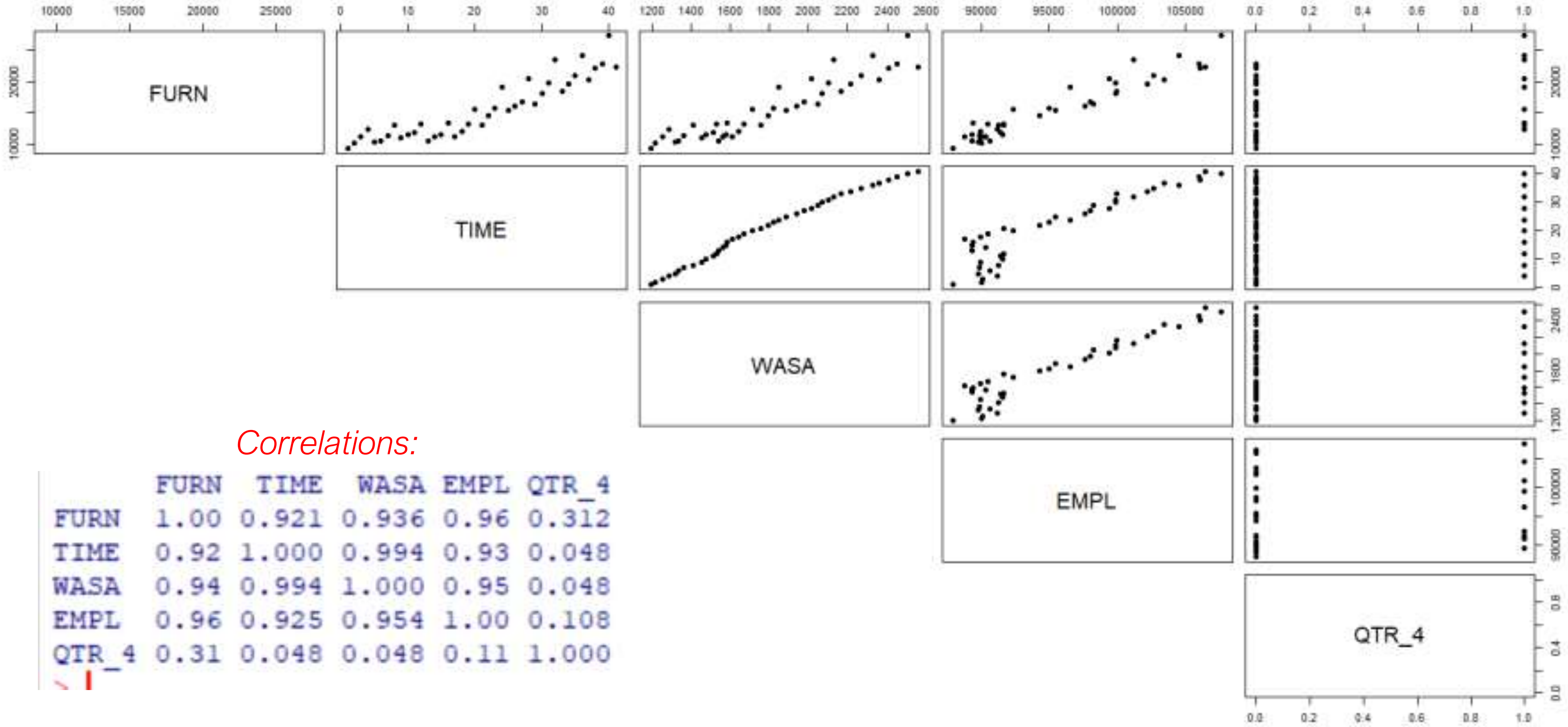
FORECAST BY APPLYING THE EQUATION

#	Step	Details
1	Assemble data…	…for what you wish to forecast (dependent variable) and the predictors that you think are associated
2	Separate “ex post” holdouts	• “training” and “test” data • Be wary how a model fits vs. how it forecasts (MAPE, MAD) • Historic random variation can be mistaken for “the real thing” • Hence, a predictive model should be tested on data that did not create it
3	Fit the model with “training” and “test”	And see what it did with those *ex post* holdouts!
4	Re-fit …	… with training and test data combined
5	Forecast (extrapolate)	Given future estimates of predictors

DEMO: LINEAR REGRESSION

HTTP://CERESANALYTICS.COM/R_FOR_IBF_BOOTCAMP/IBF_LINEAR.R

WEB APP: HTTPS://SBTHESIS.COM/SHINY_IBF



Correlations:

	FURN	TIME	WASA	EMPL	QTR_4
FURN	1.00	0.921	0.936	0.96	0.312
TIME	0.92	1.000	0.994	0.93	0.048
WASA	0.94	0.994	1.000	0.95	0.048
EMPL	0.96	0.925	0.954	1.00	0.108
QTR_4	0.31	0.048	0.048	0.11	1.000

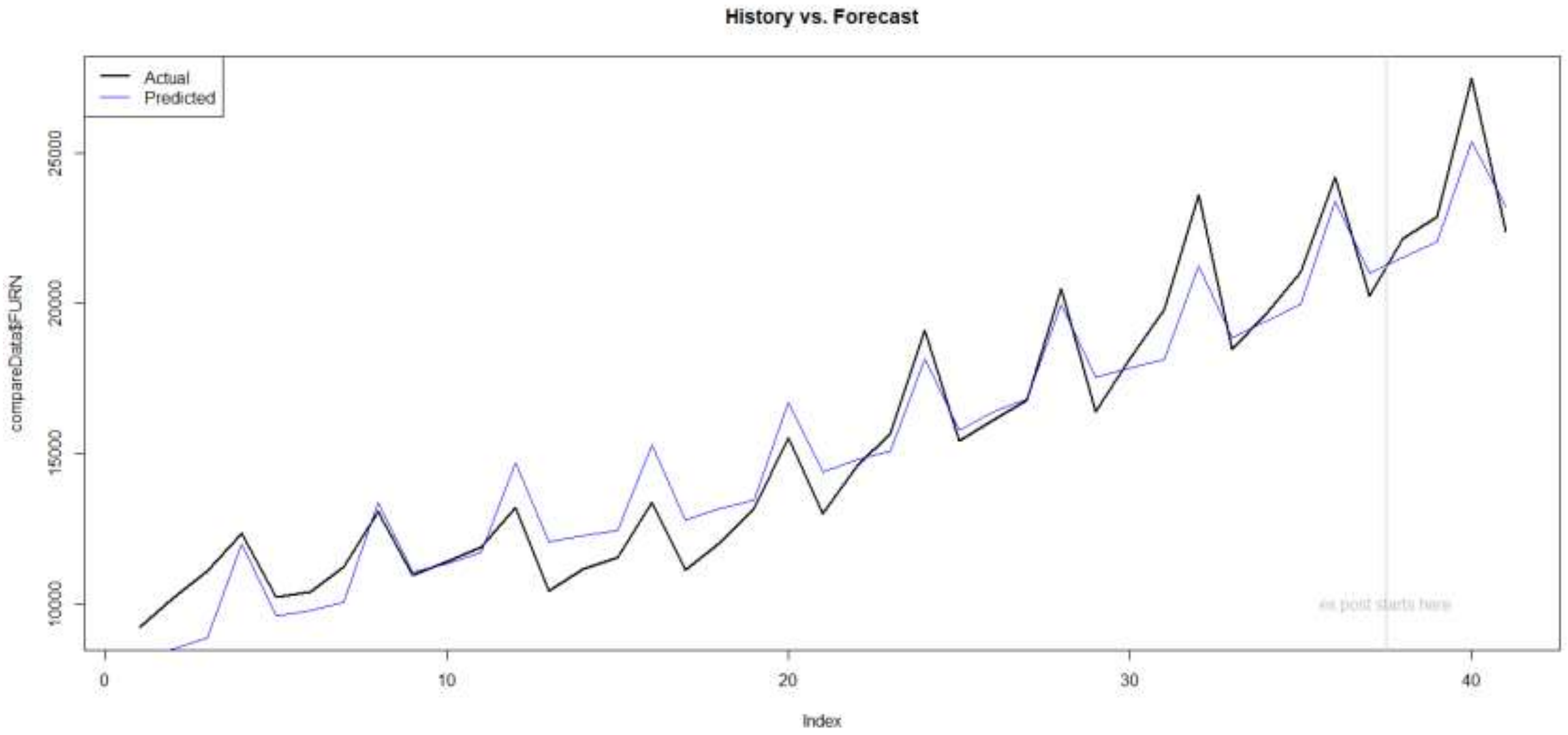
Example Data


```
> myData
```

	TIME	Year	Quarter	YearQuarterLabel	WASA	EMPL	QTR_4	BLDG	AUTO	FURN	GMER	isTrain
1	1	2010	1	2010Q1	1193.3	87973	0	9392	42960	9203	30363	1
2	2	2010	2	2010Q2	1217.3	90021	0	13637	47516	10174	25812	1
3	3	2010	3	2010Q3	1254.4	90120	0	14392	43888	11078	26026	1
4	4	2010	4	2010Q4	1284.7	91180	1	13301	40298	12334	38081	1
5	5	2011	1	2011Q1	1319.8	89832	0	10411	40016	10228	21909	1
6	6	2011	2	2011Q2	1335.1	90668	0	13057	39073	10394	26615	1
7	7	2011	3	2011Q3	1360.1	89850	0	13621	40484	11246	27039	1
8	8	2011	4	2011Q4	1411.6	91276	1	13354	38703	13071	40170	1
9	9	2012	1	2012Q1	1451.7	89964	0	11056	41919	10954	23641	1
10	10	2012	2	2012Q2	1478.1	91566	0	14559	45398	11410	30477	1
11	11	2012	3	2012Q3	1512.6	91405	0	14067	46635	11892	30065	1
12	12	2012	4	2012Q4	1530.6	91691	1	12286	39970	13204	43805	1
13	13	2013	1	2013Q1	1542.7	89341	0	9658	41554	10439	24256	1
14	14	2013	2	2013Q2	1563.9	90310	0	13659	48384	11148	30976	1
15	15	2013	3	2013Q3	1579.8	89324	0	12351	46041	11556	30579	1
16	16	2013	4	2013Q4	1586.0	89436	1	12307	46411	13370	45471	1
17	17	2014	1	2014Q1	1610.7	88815	0	10508	47558	11114	25891	1
18	18	2014	2	2014Q2	1643.5	89920	0	16287	59342	12043	33050	1
19	19	2014	3	2014Q3	1671.3	90480	0	16697	57487	13165	33385	1
20	20	2014	4	2014Q4	1715.4	92367	1	15523	57165	15534	50311	1
21	21	2015	1	2015Q1	1755.8	91637	0	13540	62200	13002	29177	1
22	22	2015	2	2015Q2	1793.1	94256	0	19190	74678	14602	36855	1
23	23	2015	3	2015Q3	1819.5	95021	0	18827	69074	15684	35418	1
24	24	2015	4	2015Q4	1847.7	96547	1	17215	67030	19088	53126	1
25	25	2016	1	2016Q1	1882.7	95450	0	13848	70453	15436	30325	1
26	26	2016	2	2016Q2	1939.8	97657	0	20319	85102	16117	37977	1
27	27	2016	3	2016Q3	1976.0	97985	0	20403	85142	16782	37028	1
28	28	2016	4	2016Q4	2012.8	99383	1	19179	71203	20491	54369	1
29	29	2017	1	2017Q1	2044.1	98213	0	16790	73380	16383	31594	1
30	30	2017	2	2017Q2	2069.8	99850	0	24400	88139	18180	37617	1
31	31	2017	3	2017Q3	2097.9	99877	0	24215	92899	19782	36297	1
32	32	2017	4	2017Q4	2128.4	101169	1	22042	81838	23605	52457	1
33	33	2018	1	2018Q1	2163.2	99922	0	18182	75122	18483	30718	1
34	34	2018	2	2018Q2	2211.9	102186	0	22670	90515	19659	41090	1
35	35	2018	3	2018Q3	2265.1	102657	0	21204	89907	21042	40706	1
36	36	2018	4	2018Q4	2325.0	104522	1	19746	77163	24203	60491	1
37	37	2019	1	2019Q1	2358.7	103445	0	16656	84508	20242	34723	1
38	38	2019	2	2019Q2	2405.6	106050	0	24552	99279	22143	42871	0
39	39	2019	3	2019Q3	2452.2	105963	0	23775	94556	22870	42072	0
40	40	2019	4	2019Q4	2505.1	107644	1	22449	87039	27486	64101	0
41	41	2020	1	2020Q1	2560.7	106425	0	18145	87696	22399	36516	0

```
> {
+ print(summary(myLinModel))
+ print(rev(mapeVec)) #reverse order to get history first
+ }

Call:
lm(formula = FURN ~ WASA + QTR_4, data = trainData)

Residuals:
    Min      1Q  Median      3Q     Max
-1928.0  -903.5   -27.5   623.1  2351.8

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -4885.6419   981.9892  -4.975 1.85e-05 ***
WASA           10.9793     0.5579  19.680  < 2e-16 ***
QTR_4        2770.4650   432.5868   6.404 2.58e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1127 on 34 degrees of freedom
Multiple R-squared:  0.9288,    Adjusted R-squared:  0.9246
F-statistic: 221.6 on 2 and 34 DF,  p-value: < 2.2e-16

        MAPE History (%) MAPE ex post Forecast (%)
                     6.5                       4.4
```

Ex post Forecast



Institute of Business Forecasting & Planning

```
> summary(myLinModel)

Call:
lm(formula = FURN ~ WASA + QTR_4, data = trainData)

Residuals:
    Min      1Q  Median      3Q     Max
-1928.0  -903.5   -27.5   623.1  2351.8

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -4885.6419   981.9892  -4.975 1.85e-05 ***
WASA           10.9793     0.5579  19.680  < 2e-16 ***
QTR_4        2770.4650   432.5868   6.404 2.58e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '

Residual standard error: 1127 on 34 degrees of freedom
Multiple R-squared:  0.9288,	Adjusted R-squared:  0.9246
F-statistic: 221.6 on 2 and 34 DF,  p-value: < 2.2e-16
```

```
> summary(myLinModel_fin)

Call:
lm(formula = FURN ~ WASA + QTR_4, data = myData)

Residuals:
    Min      1Q  Median      3Q     Max
-2069.0  -946.0   123.3   703.4  2319.6

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -5360.6147   841.1960  -6.373 1.76e-07 ***
WASA           11.2556     0.4557  24.697  < 2e-16 ***
QTR_4        2948.1918   411.0998   7.171 1.44e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1129 on 38 degrees of freedom
Multiple R-squared:  0.9471,	Adjusted R-squared:  0.9443
F-statistic:   340 on 2 and 38 DF,  p-value: < 2.2e-16
```

MACHINE LEARNING ALTERNATIVE
NEURAL NETWORKS

5

NEURAL NETWORK

- What it is/how it works
  - An algorithm meant to mimic the attributes of the human brain, which uses interconnected processing nodes to recognize patterns among inputs and one or more outputs…
  - Statistics vs. Machine Learning—This is a Machine Learning technique, which relies on interactions and nonlinearities
- How to evaluate your results
  - Much like *performance of* a regression model (ex post forecast)
- How to apply your results
  - Apply fitted model after train/test/extrapolate
- Example
  - Script in R or Excel *(depending on consensus)*

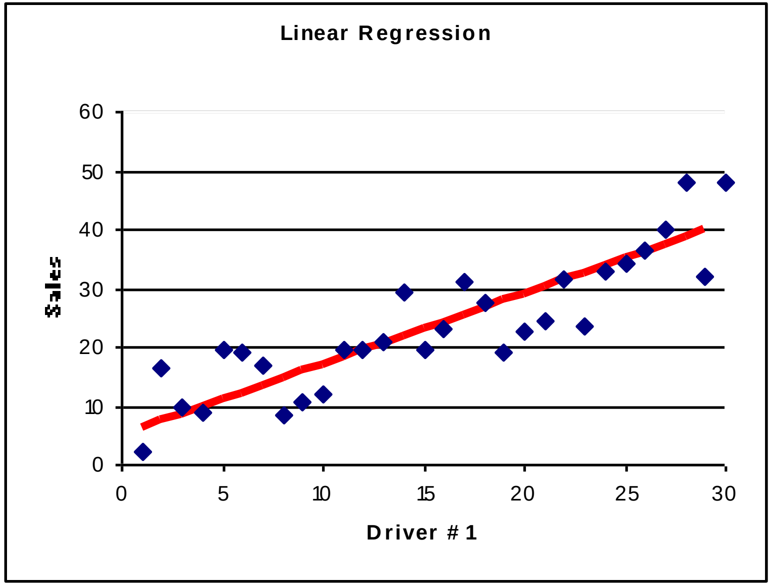
NEURAL NETWORK

COMPARISON WITH BRAIN

Brain	Neural Network
100,000,000,000 neurons	Neurons number from few to hundreds or thousands
Pattern recognition	Pattern recognition
Vast amount of perceptions	Vast amount of data (or signals)
Learns relatively quickly	Learns relatively slowly
Parallel processing of multiple inputs is aggregated through synapses between cells	Parallel processing of multiple inputs is aggregated through arithmetic combinations between nodes

PREFACE: NEURAL NETWORK IN REGRESSION CONTEXT

TWO APPLICATIONS OF REGRESSION
PREDICTION OF LEVELS OR CLASSES (EVENTS)



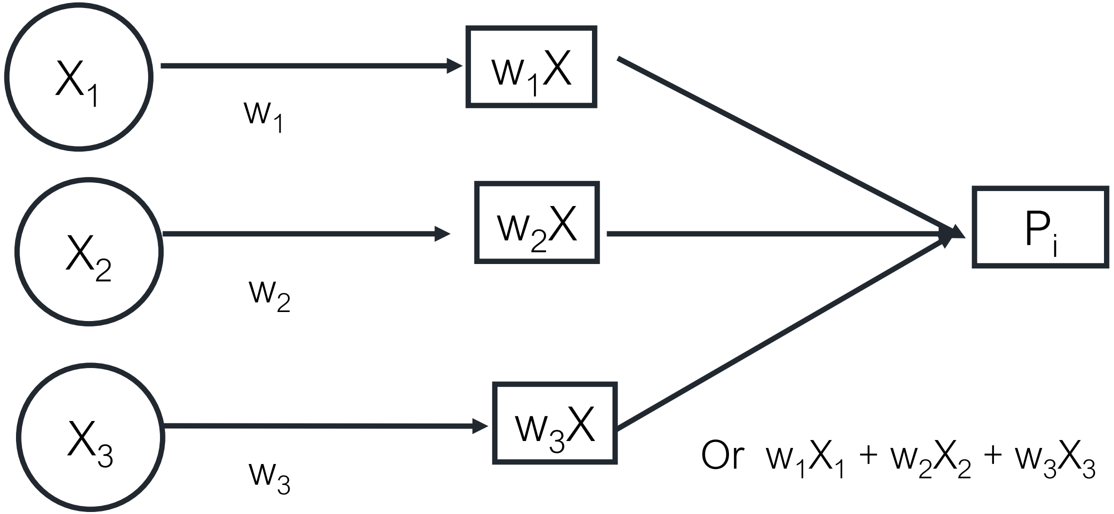
Levels

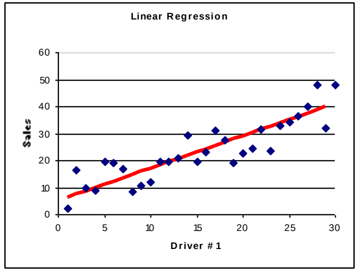
Classes (Events)

THE PERCEPTRON NEURON

CAN COMPUTE A LINEAR COMBINATION LIKE LINEAR REGRESSION…

Start with a linear weighted sum, where the weights are just like regression's β…



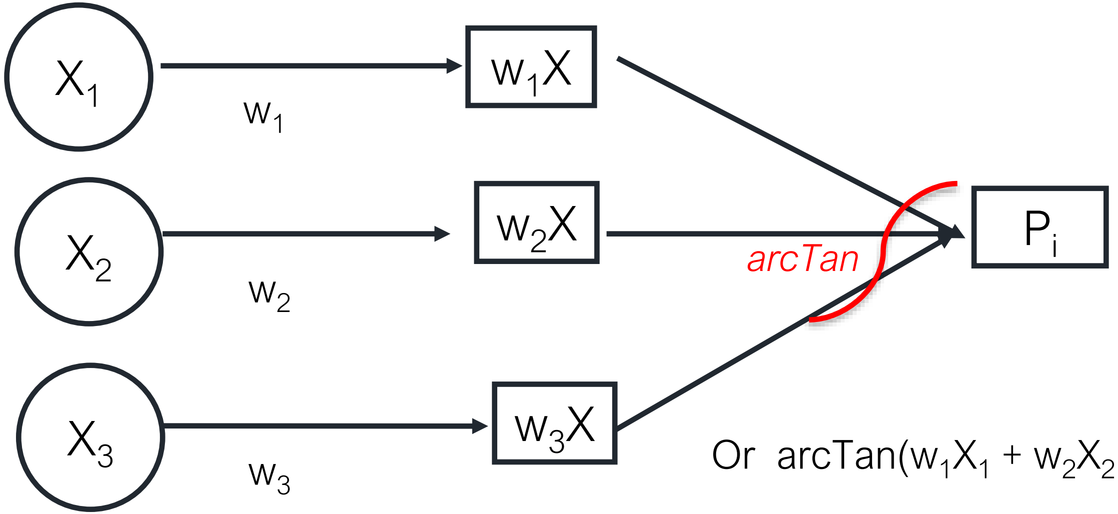


Or $w_1X_1 + w_2X_2 + w_3X_3$

THE PERCEPTRON NEURON

...OR APPLY A NONLINEAR TWIST TO ACT AS A CLASSIFIER

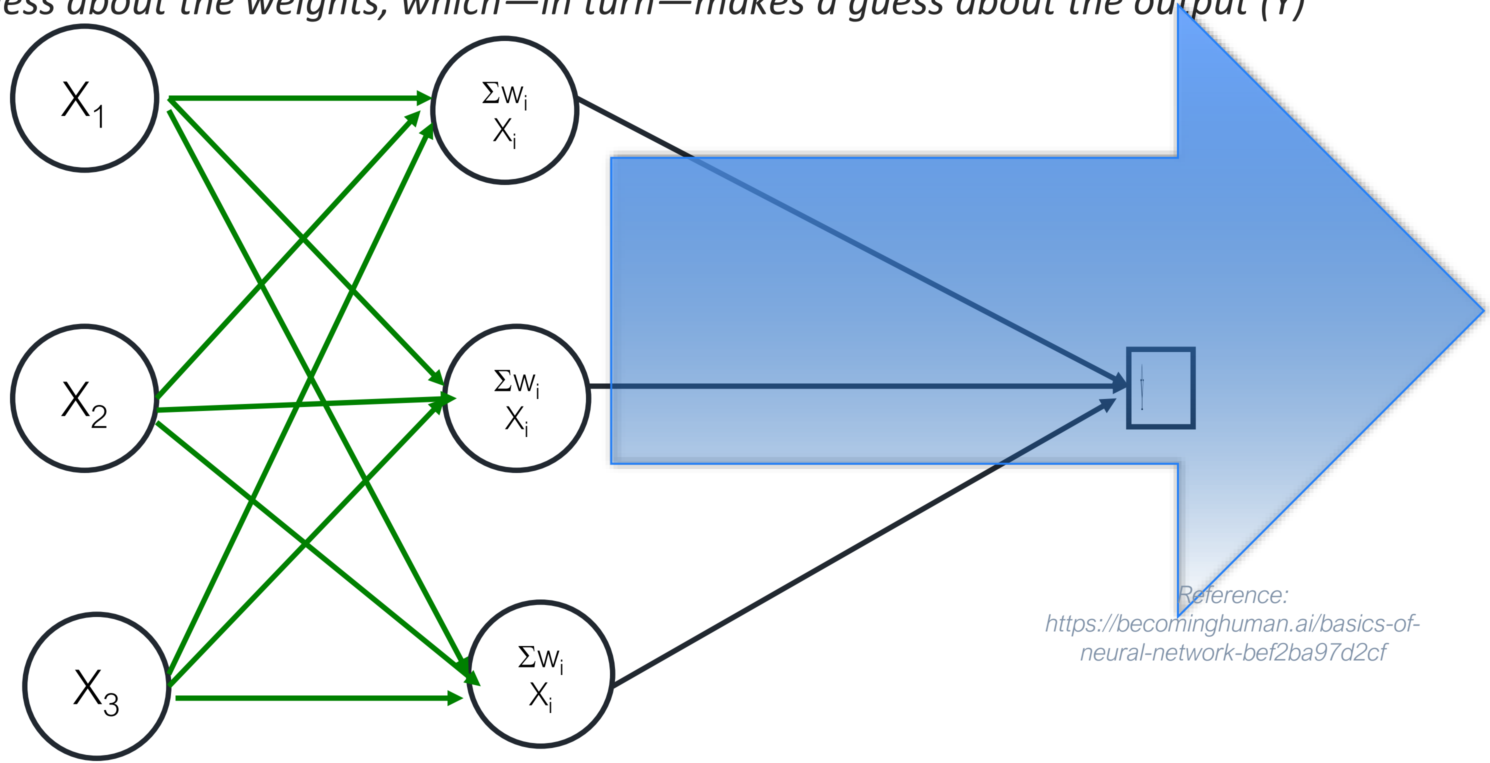
Start with a linear weighted sum, where the weights are just like regression's β...



Or $\arctan(w_1X_1 + w_2X_2 + w_3X_3)$

Neural Net: Forward Propagation

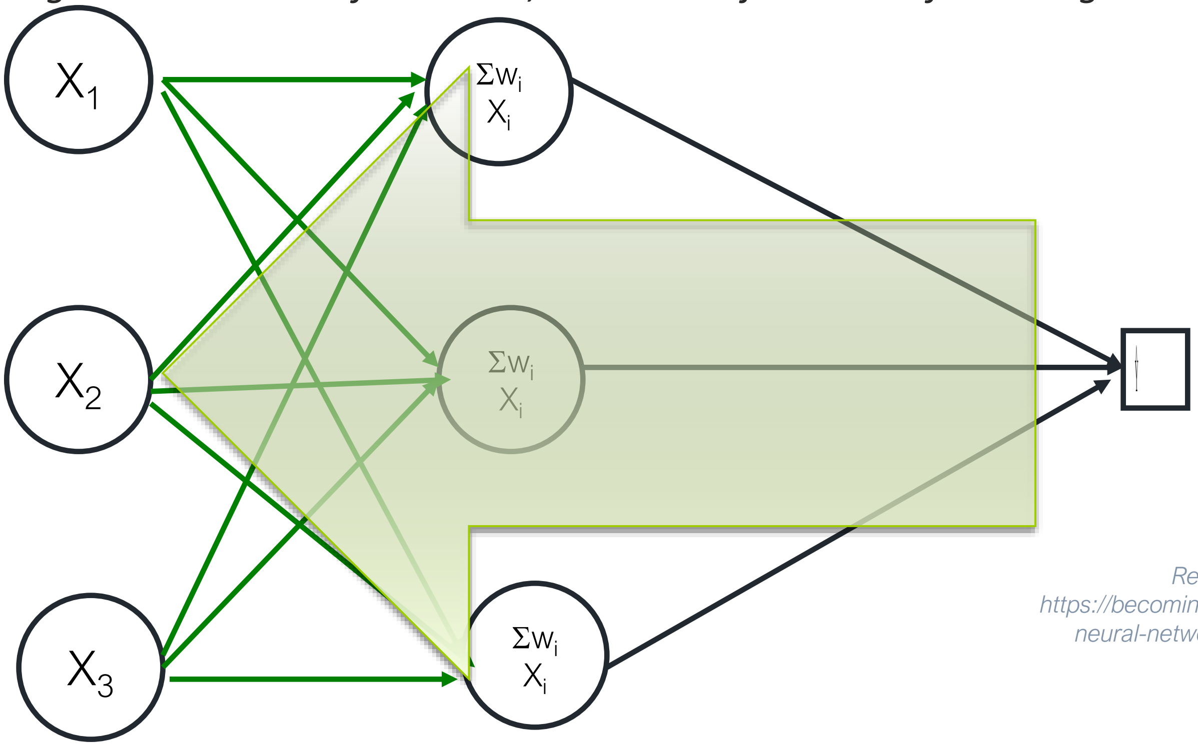
make a guess about the weights, which—in turn—makes a guess about the output (Y)



Reference: https://becominghuman.ai/basics-of-neural-network-bef2ba97d2cf

Neural Net: Backward Propagation

Recognize the extent of the error, to allow adjustment of the weights



Reference: https://becominghuman.ai/basics-of-neural-network-bef2ba97d2cf

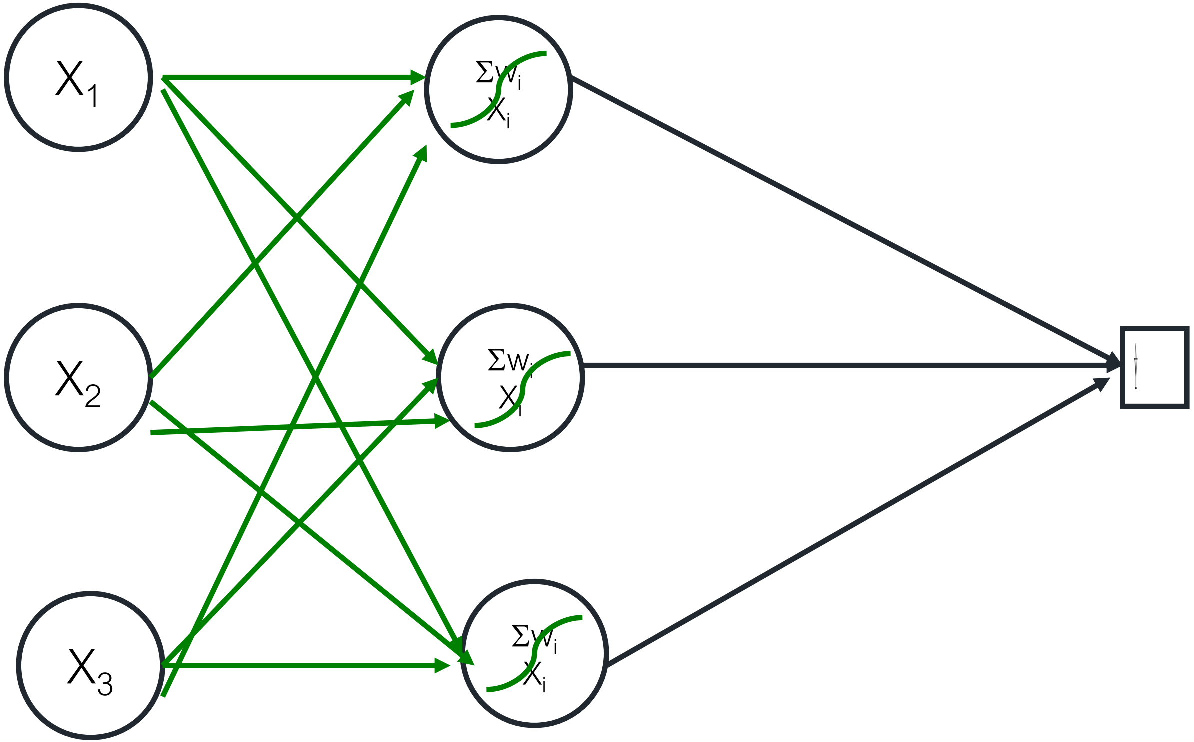
NEURAL NET: NUTS AND BOLTS

- Partial derivatives essentially tell you how changing the error can be attributed to the weights
- Cost function—like linear regression’s minimizing the sum of squared errors—is how those errors should be interpreted for optimization
- Learning rate is how big an adjustment you allow when a weight changes
- Activation function is how you process output that would otherwise be linear (e.g., to separate classification tasks)

Reference: https://becominghuman.ai/basics-of-neural-network-bef2ba97d2cf

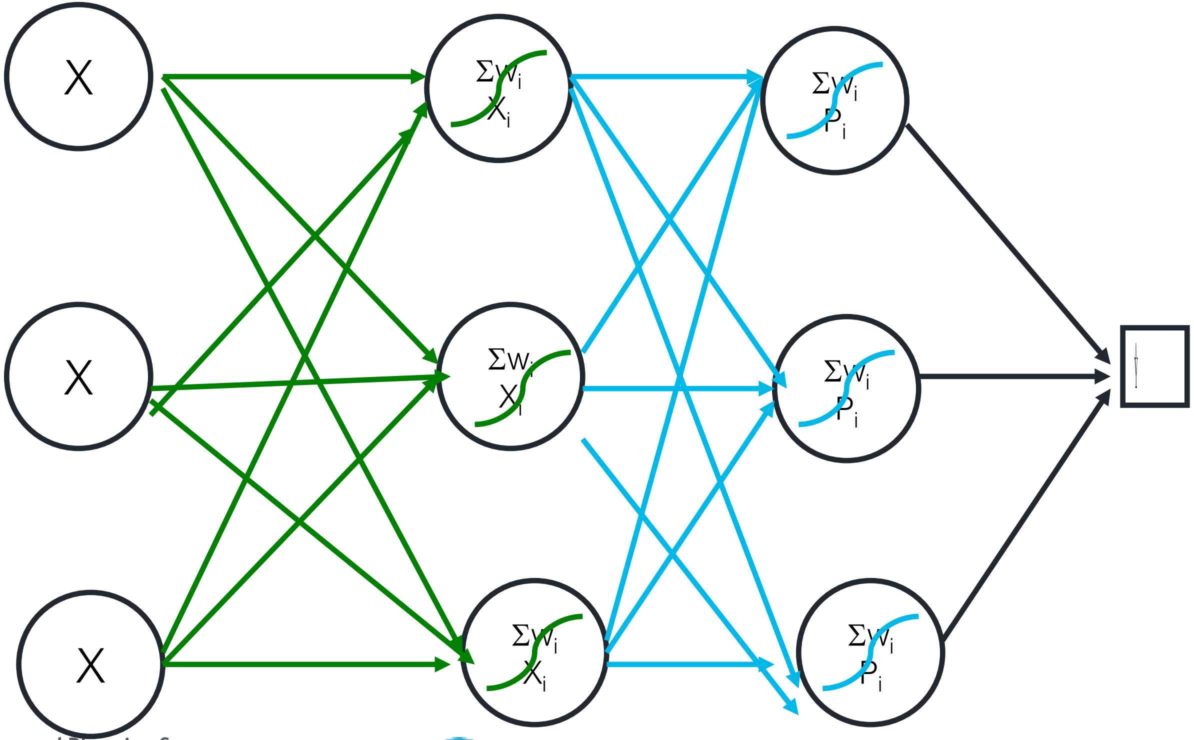
CONVENTIONAL DEPICTION

3-NODE NETWORK WITH 1 HIDDEN LAYER



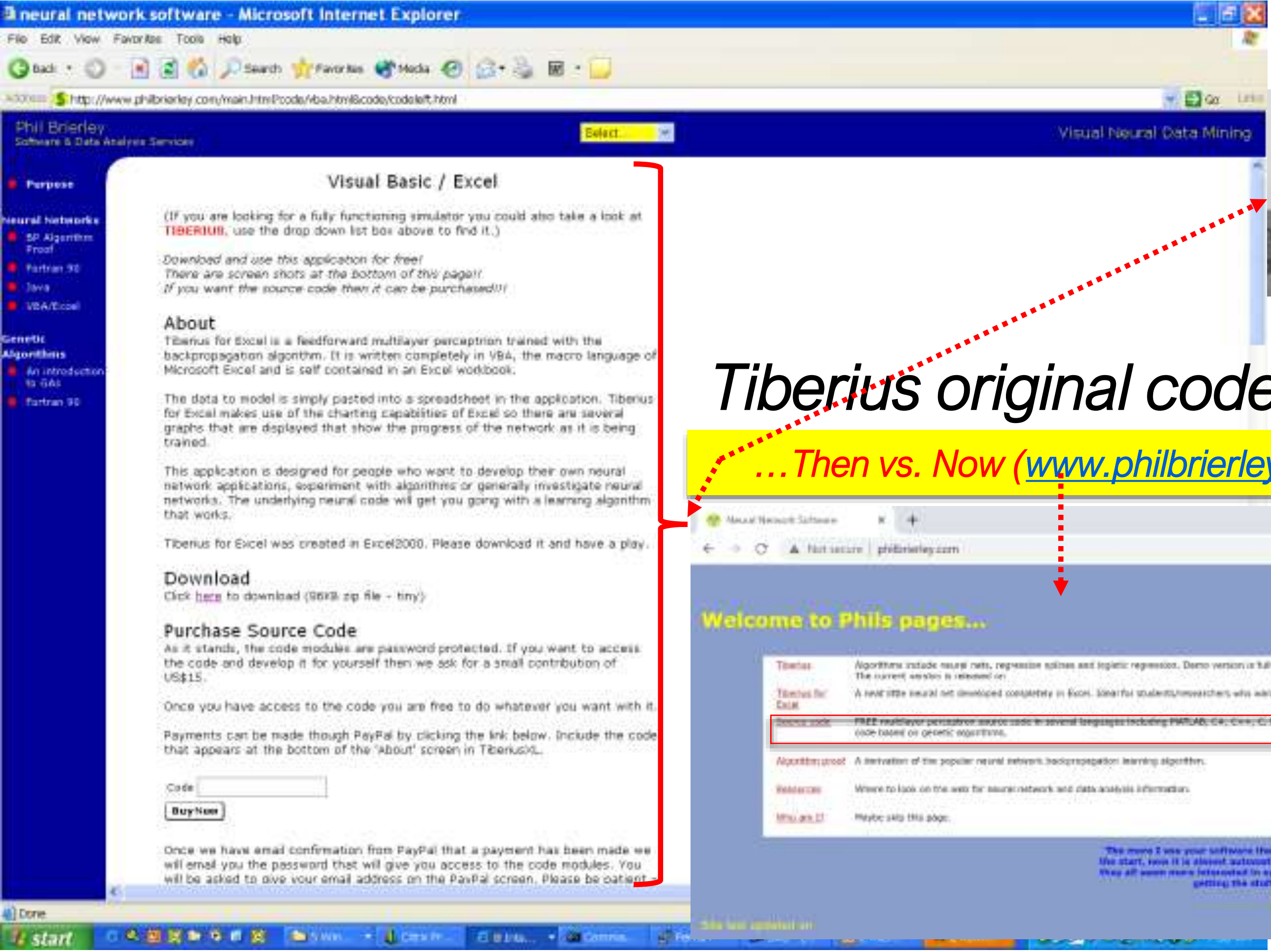
CONVENTIONAL DEPICTION

3-NODE NETWORK WITH 2 HIDDEN LAYERS



DEMO: NEURAL NETWORK

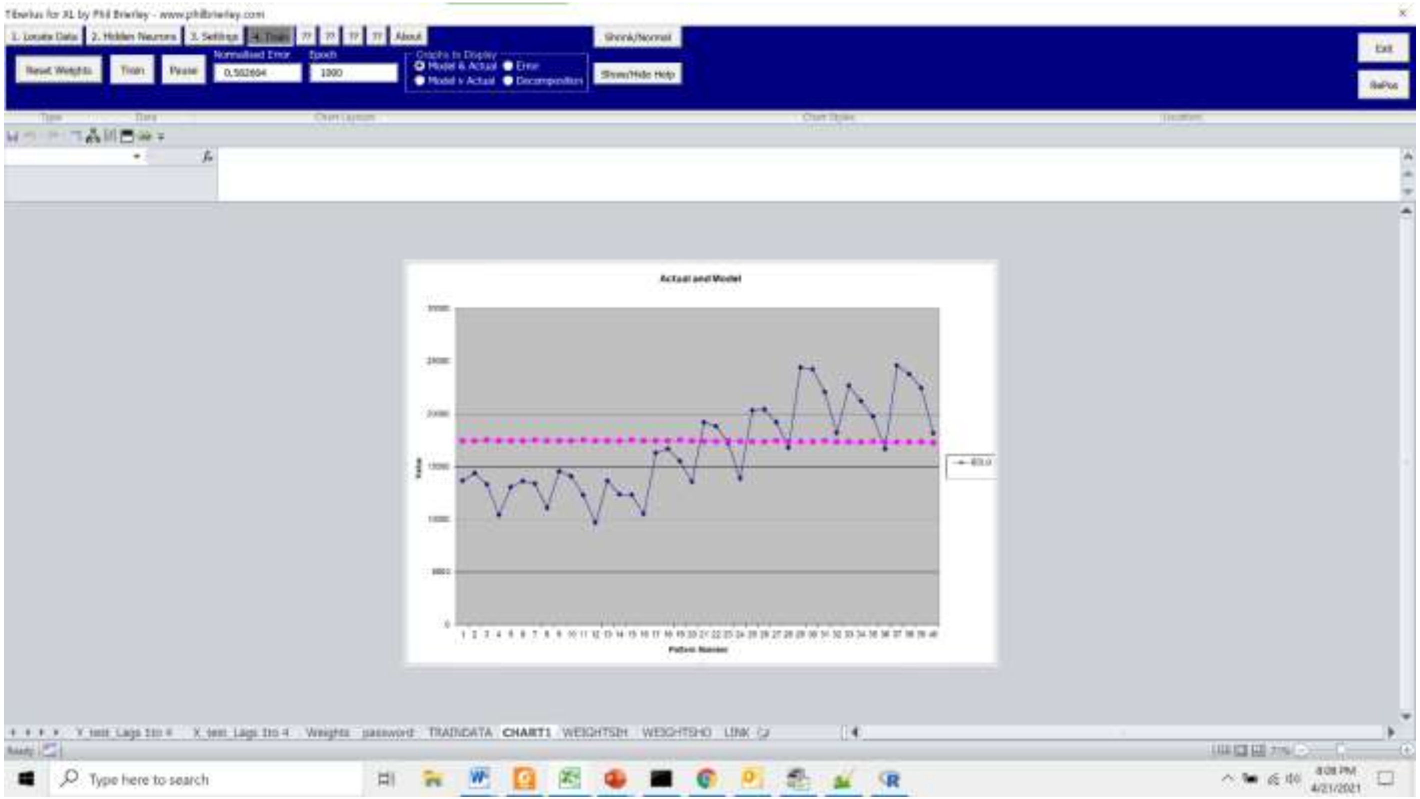
http://ceresanalytics.com/R_for_IBF_BootCamp/ibf_neuralNetwork.r



Tiberius original code by Phil Brierley, PhD

…Then vs. Now (www.philbrierley.com) …

http://www.ceresAnalytics.com/for_IBF_Predictive_Analytics_2021/neuralNet_cfTiberius.xlsm

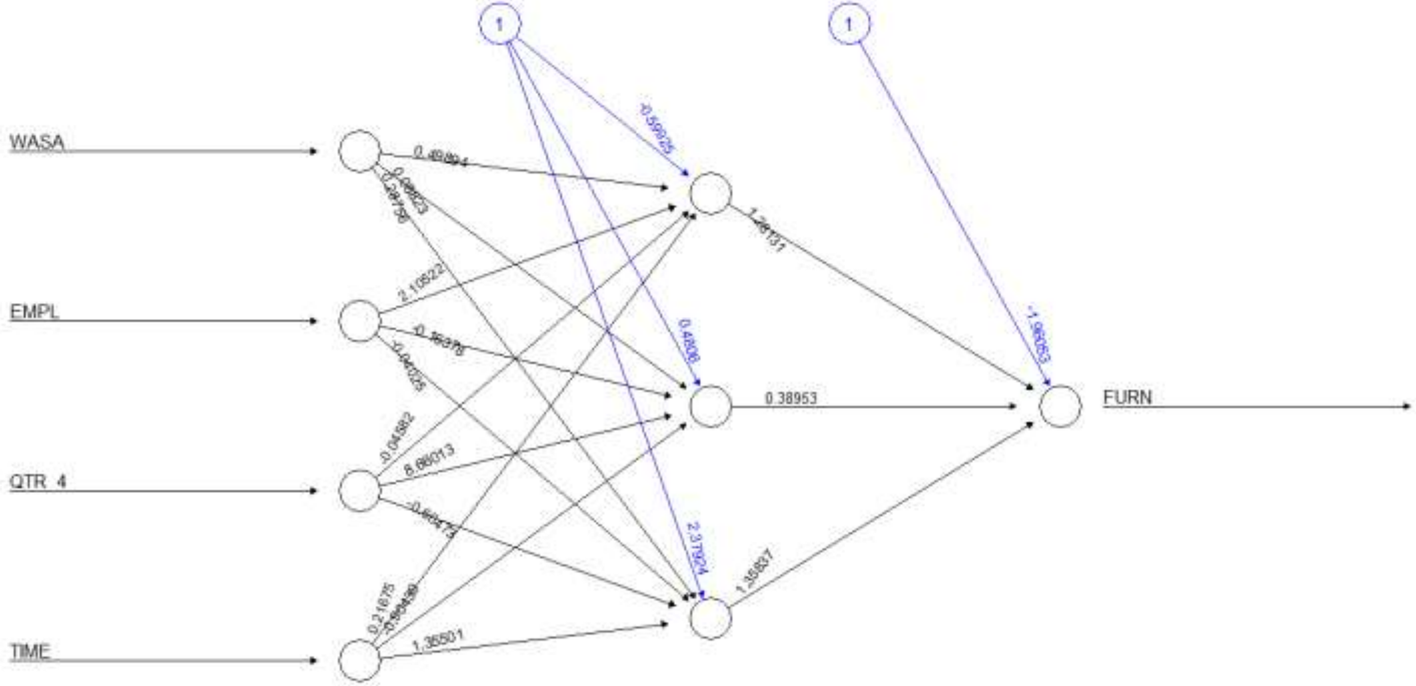


R-based example
Same Data as Regression


```
> print(myData)
   TIME Year Quarter YearQuarterLabel   WASA   EMPL QTR_4  BLDG  AUTO  FURN  GMER isTrain
1     1 2010       1           2010Q1 1193.3  87973     0  9392 42960  9203 30363       1
2     2 2010       2           2010Q2 1217.3  90021     0 13637 47516 10174 25812       1
3     3 2010       3           2010Q3 1254.4  90120     0 14392 43888 11078 26026       1
4     4 2010       4           2010Q4 1284.7  91180     1 13301 40298 12334 38081       1
5     5 2011       1           2011Q1 1319.8  89832     0 10411 40016 10228 21909       1
6     6 2011       2           2011Q2 1335.1  90668     0 13057 39073 10394 26615       1
7     7 2011       3           2011Q3 1360.1  89850     0 13621 40484 11246 27039       1
8     8 2011       4           2011Q4 1411.6  91276     1 13354 38703 13071 40170       1
9     9 2012       1           2012Q1 1451.7  89964     0 11056 41919 10954 23641       1
10   10 2012       2           2012Q2 1478.1  91566     0 14559 45398 11410 30477       1
11   11 2012       3           2012Q3 1512.6  91405     0 14067 46635 11892 30065       1
12   12 2012       4           2012Q4 1530.6  91691     1 12286 39970 13204 43805       1
13   13 2013       1           2013Q1 1542.7  89341     0  9658 41554 10439 24256       1
14   14 2013       2           2013Q2 1563.9  90310     0 13659 48384 11148 30976       1
15   15 2013       3           2013Q3 1579.8  89324     0 12351 46041 11556 30579       1
16   16 2013       4           2013Q4 1586.0  89436     1 12307 46411 13370 45471       1
17   17 2014       1           2014Q1 1610.7  88815     0 10508 47558 11114 25891       1
18   18 2014       2           2014Q2 1643.5  89920     0 16287 59342 12043 33050       1
19   19 2014       3           2014Q3 1671.3  90480     0 16697 57487 13165 33385       1
20   20 2014       4           2014Q4 1715.4  92367     1 15523 57165 15534 50311       1
21   21 2015       1           2015Q1 1755.8  91637     0 13540 62200 13002 29177       1
22   22 2015       2           2015Q2 1793.1  94256     0 19190 74678 14602 36855       1
23   23 2015       3           2015Q3 1819.5  95021     0 18827 69074 15684 35418       1
24   24 2015       4           2015Q4 1847.7  96547     1 17215 67030 19088 53126       1
25   25 2016       1           2016Q1 1882.7  95450     0 13848 70453 15436 30325       1
26   26 2016       2           2016Q2 1939.8  97657     0 20319 85102 16117 37977       1
27   27 2016       3           2016Q3 1976.0  97985     0 20403 85142 16782 37028       1
28   28 2016       4           2016Q4 2012.8  99383     1 19179 71203 20491 54369       1
29   29 2017       1           2017Q1 2044.1  98213     0 16790 73380 16383 31594       1
30   30 2017       2           2017Q2 2069.8  99850     0 24400 88139 18180 37617       1
31   31 2017       3           2017Q3 2097.9  99877     0 24215 92899 19782 36297       1
32   32 2017       4           2017Q4 2128.4 101169     1 22042 81838 23605 52457       1
33   33 2018       1           2018Q1 2163.2  99922     0 18182 75122 18483 30718       1
34   34 2018       2           2018Q2 2211.9 102186     0 22670 90515 19659 41090       1
35   35 2018       3           2018Q3 2265.1 102657     0 21204 89907 21042 40706       1
36   36 2018       4           2018Q4 2325.0 104522     1 19746 77163 24203 60491       1
37   37 2019       1           2019Q1 2358.7 103445     0 16656 84508 20242 34723       1
38   38 2019       2           2019Q2 2405.6 106050     0 24552 99279 22143 42871       0
39   39 2019       3           2019Q3 2452.2 105963     0 23775 94556 22870 42072       0
40   40 2019       4           2019Q4 2505.1 107644     1 22449 87039 27486 64101       0
41   41 2020       1           2020Q1 2560.7 106425     0 18145 87696 22399 36516       0
```

Scale Variables

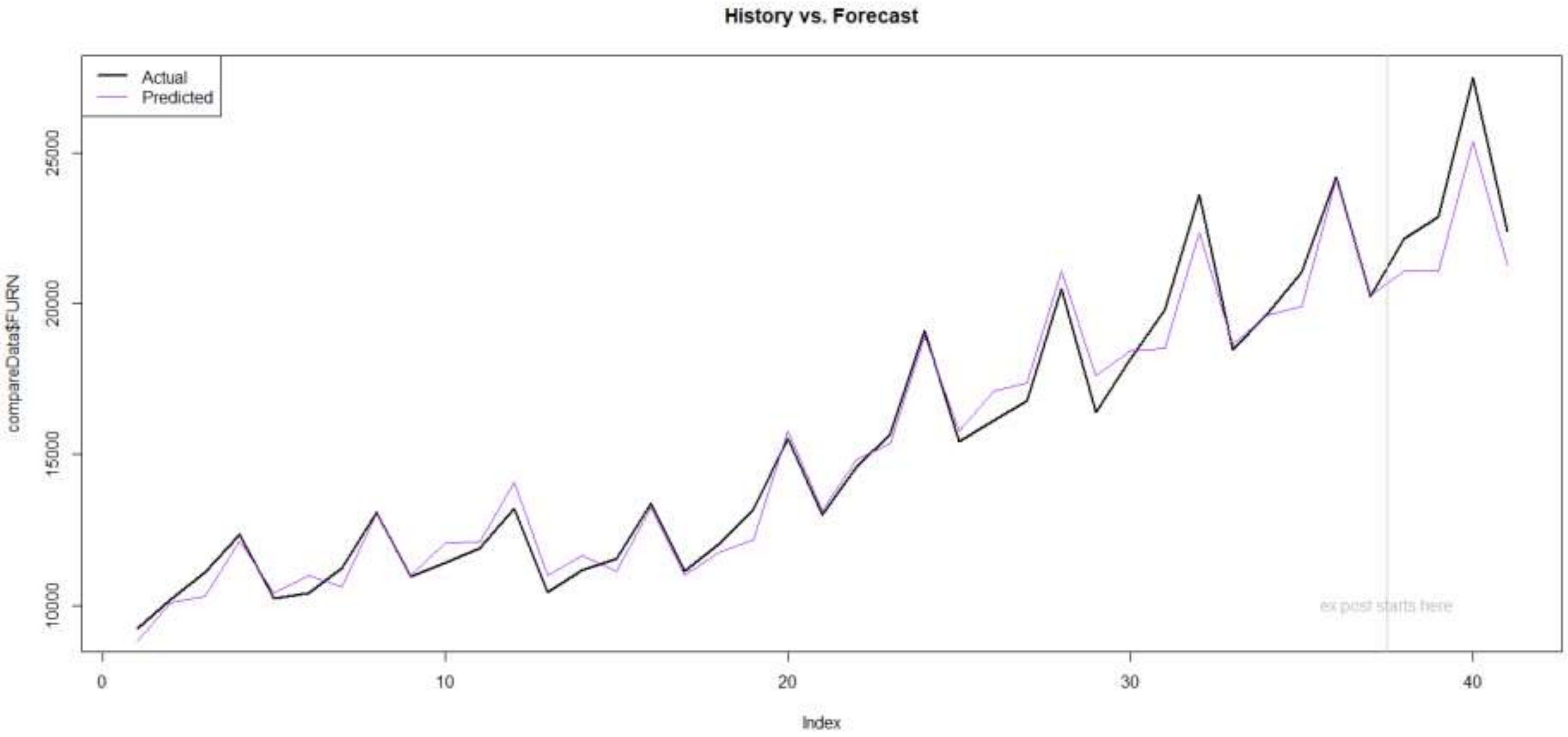

```
> print(rbind(scaled_dataFor_NN_train,scaled_dataFor_NN_test))
        FURN  WASA  EMPL QTR_4  TIME
2010Q1 0.000 0.000 0.000     0 0.000
2010Q2 0.053 0.018 0.104     0 0.025
2010Q3 0.103 0.045 0.109     0 0.050
2010Q4 0.171 0.067 0.163     1 0.075
2011Q1 0.056 0.093 0.095     0 0.100
2011Q2 0.065 0.104 0.137     0 0.125
2011Q3 0.112 0.122 0.095     0 0.150
2011Q4 0.212 0.160 0.168     1 0.175
2012Q1 0.096 0.189 0.101     0 0.200
2012Q2 0.121 0.208 0.183     0 0.225
2012Q3 0.147 0.234 0.174     0 0.250
2012Q4 0.219 0.247 0.189     1 0.275
2013Q1 0.068 0.256 0.070     0 0.300
2013Q2 0.106 0.271 0.119     0 0.325
2013Q3 0.129 0.283 0.069     0 0.350
2013Q4 0.228 0.287 0.074     1 0.375
2014Q1 0.105 0.305 0.043     0 0.400
2014Q2 0.155 0.329 0.099     0 0.425
2014Q3 0.217 0.350 0.127     0 0.450
2014Q4 0.346 0.382 0.223     1 0.475
2015Q1 0.208 0.411 0.186     0 0.500
2015Q2 0.295 0.439 0.319     0 0.525
2015Q3 0.354 0.458 0.358     0 0.550
2015Q4 0.541 0.479 0.436     1 0.575
2016Q1 0.341 0.504 0.380     0 0.600
2016Q2 0.378 0.546 0.492     0 0.625
2016Q3 0.415 0.572 0.509     0 0.650
2016Q4 0.617 0.599 0.580     1 0.675
2017Q1 0.393 0.622 0.521     0 0.700
2017Q2 0.491 0.641 0.604     0 0.725
2017Q3 0.579 0.662 0.605     0 0.750
2017Q4 0.788 0.684 0.671     1 0.775
2018Q1 0.508 0.709 0.607     0 0.800
2018Q2 0.572 0.745 0.723     0 0.825
2018Q3 0.648 0.784 0.746     0 0.850
2018Q4 0.820 0.828 0.841     1 0.875
2019Q1 0.604 0.852 0.787     0 0.900
2019Q2 0.708 0.887 0.919     0 0.925
2019Q3 0.748 0.921 0.915     0 0.950
2019Q4 1.000 0.959 1.000     1 0.975
2020Q1 0.722 1.000 0.938     0 1.000
```



Error: 0.019146 Steps: 84

Random seed = 2

Compare fit vs. actual



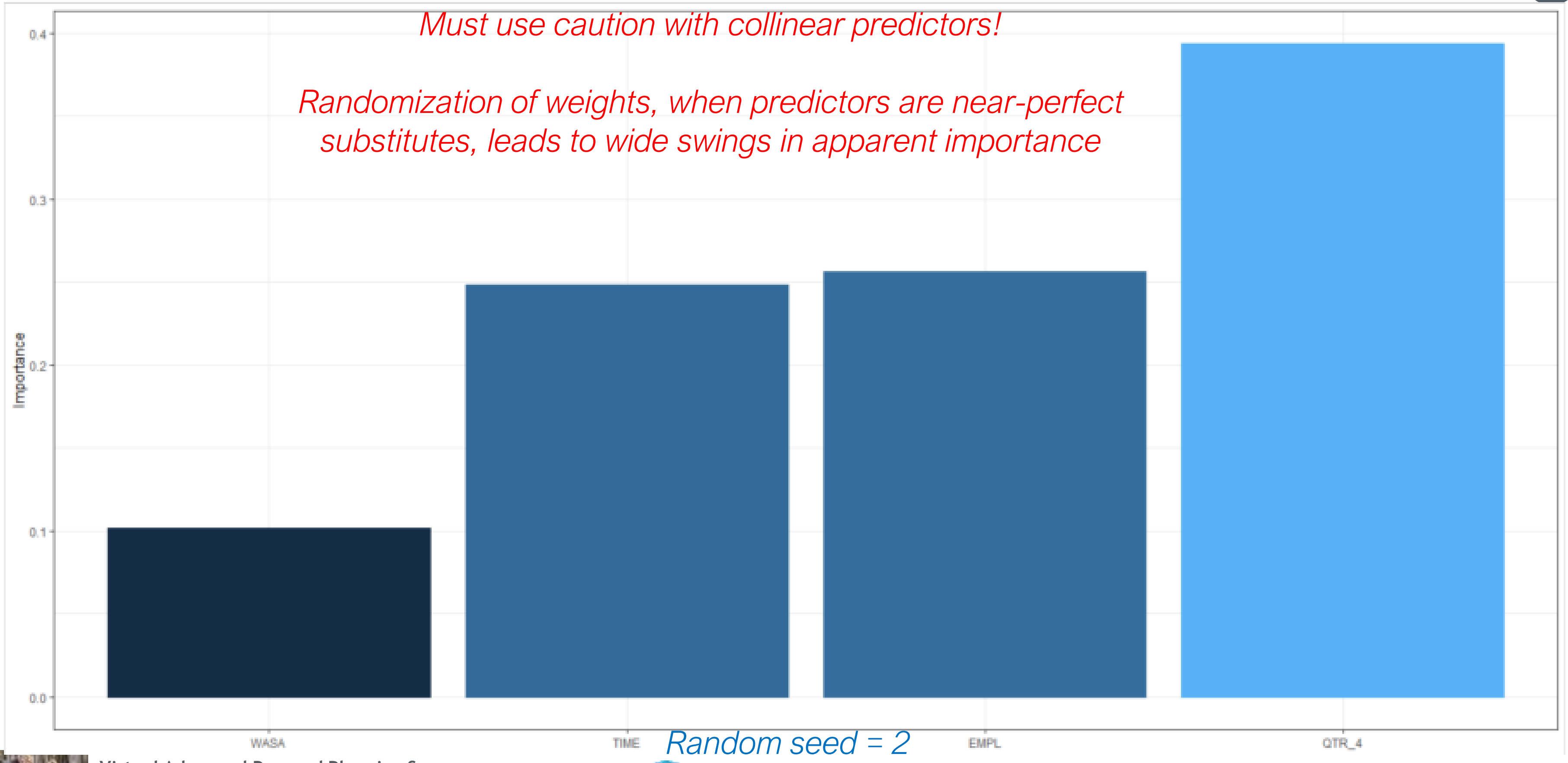
Compare MAPE with Regression

Neural Network

MAPE History (%)	MAPE ex post Forecast (%)
3.2	6.3

Linear Regression

MAPE History (%)	MAPE ex post Forecast (%)
6.5	4.4


Must use caution with collinear predictors!
Randomization of weights, when predictors are near-perfect substitutes, leads to wide swings in apparent importance
Importance
0.4
0.3
0.2
0.1
0.0
WASA
TIME
EMPL
QTR_4
Random seed = 2

PREDICTING EVENTS
AN INTRODUCTION

5

PREDICTING EVENTS

"LIMITED DEPENDENT VARIABLES"

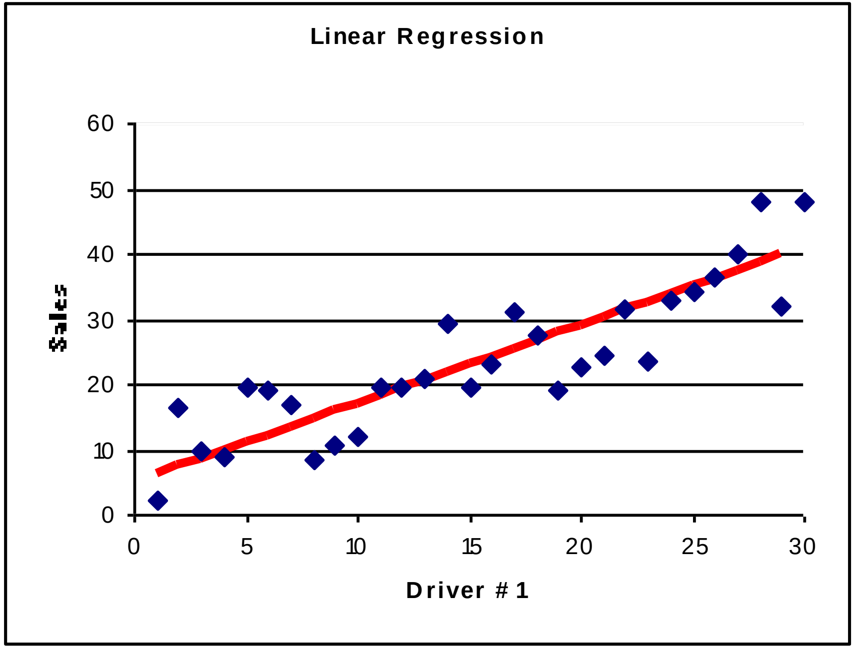
DateVal	SKU	Demand
202111	1234	94
202112	1234	64
202201	1234	26
202202	1234	0
202203	1234	27
202204	1234	59
202205	1234	47
202206	1234	27
202207	1234	97
202208	1234	4
202209	1234	23
202210	1234	62

CustomerID	Promotion Response	Numeric Response
1000001	No	0
1000002	No	0
1000003	No	0
1000004	No	0
1000005	No	0
1000006	No	0
1000007	Yes	1
1000008	No	0
1000009	Yes	1
1000010	No	0
1000011	No	0
1000012	No	0

CustomerID	Lost to Competitor	Numeric Response
300409	Yes	1
300410	No	0
300411	No	0
300412	No	0
300413	Yes	1
300414	Yes	1
300415	Yes	1
300416	No	0
300417	No	0
300418	No	0
300419	No	0
300420	No	0

Limited

PREDICTING LEVELS VS. EVENTS



Level: Scalar or Continuous

Limited: Event Yes/No

THE "CONFUSION MATRIX"

		PREDICTED	
		Yes	No
ACTUAL	Yes	True Positive	False Negative
	No	False Positive	True Negative

RISK VS. REWARD
CHEAP E-MAIL PROMOTION

Result	Model says...	In reality...	Impact
True Positive	YES customer will respond	YES customer would respond	✓
False Positive	YES customer will respond	NO s/he's not interested	Who cares-- it was cheap! ✓–
True Negative	NO customer will not respond	NO s/he's not interested	✓
False Negative	NO a customer will not respond	YES s/he would have!	Darn--it was cheap: you could have had more business! X

RISK VS. REWARD
EXPENSIVE LUXURY GIFT PROMOTION

Result	Model says...	In reality...	Impact
True Positive	**YES** customer will respond	**YES** customer would respond	✓
False Positive	**YES** customer will respond	**NO** s/he's not interested	Darn-- you spent: a lot for that useless gift! X
True Negative	**NO** customer will not respond	**NO** s/he's not interested	✓
False Negative	**NO** a customer will not respond	**YES** s/he would have!	Though you didn't make money, you avoided expensive risk! ✓ –

THE "CONFUSION MATRIX"

SUPPORTS BUSINESS STRATEGY

		PREDICTED	
		Yes	No
ACTUAL	Yes	True Positive	False Negative
	No	False Positive	True Negative

Risk this if:
- *Cost of failing is high ("it costs a lot to try for those customers")*
- *Cost of missed opportunity is low*

Risk this if:
- *Cost of failing is low ("e-mail is cheap")*
- *Cost of missed opportunity is high*

WHICH STRATEGY IS EMBRACED?

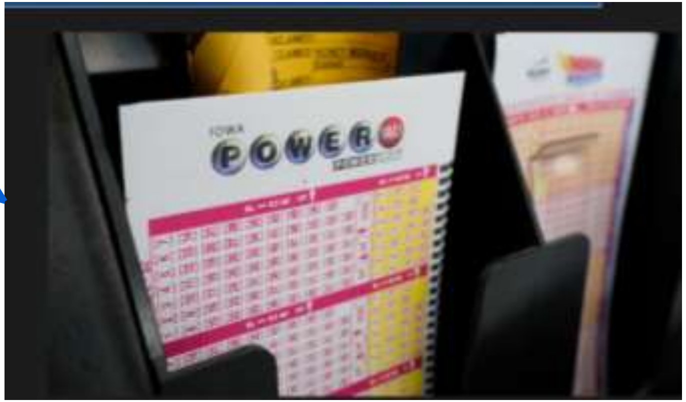
BY THE LOTTERY TICKET BUYER

		PREDICTED	
		Yes	No
ACTUAL	Yes	True Positive	False Negative
	No	False Positive	True Negative

- *Is the cost of failing high?*
- *Is the cost of a missed opportunity low?*

?

- *Is the cost of failing low?*
- *Is the cost of a missed opportunity high?*



Source: AP Photo/Charlie Neibergall

LOGISTIC REGRESSION ANALYSIS

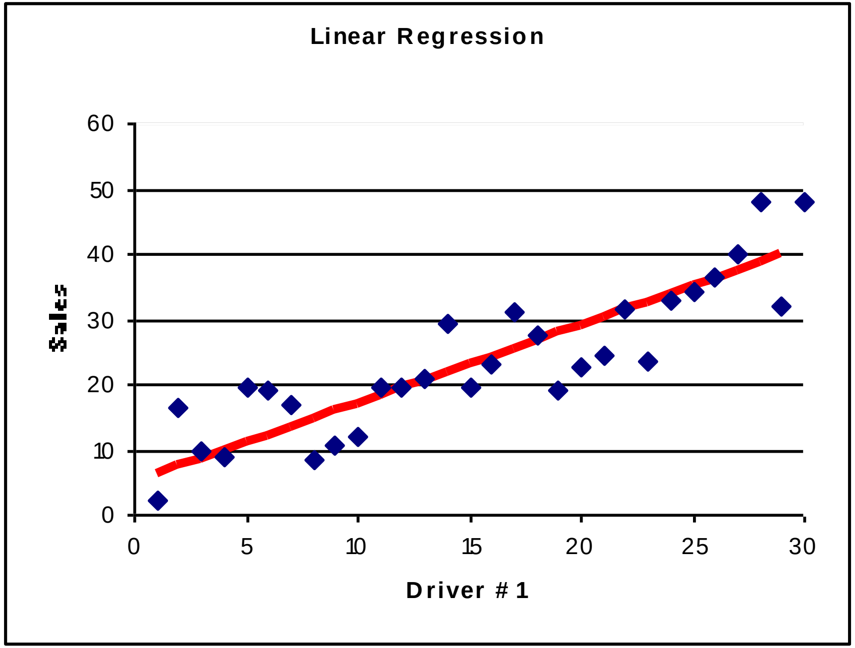
5

LOGISTIC REGRESSION ANALYSIS

- What it is/how it works
  - Predictive technique that articulates an S curve based on predictors (independent variables) and a yes/no outcome through its log-of-odds (which can be transformed to probability)
  - Statistics vs. Machine Learning—variable selection algorithms, optimization algorithms
- How to evaluate your results
  - roughly similar to linear regression
- How to apply your results
  - Extrapolate with equation and/or apply fraction of yes/no's to add-factor forecast
- Example
  - R (today), Excel

LOGISTIC REGRESSION

COMPARE WITH LINEAR REGRESSION…



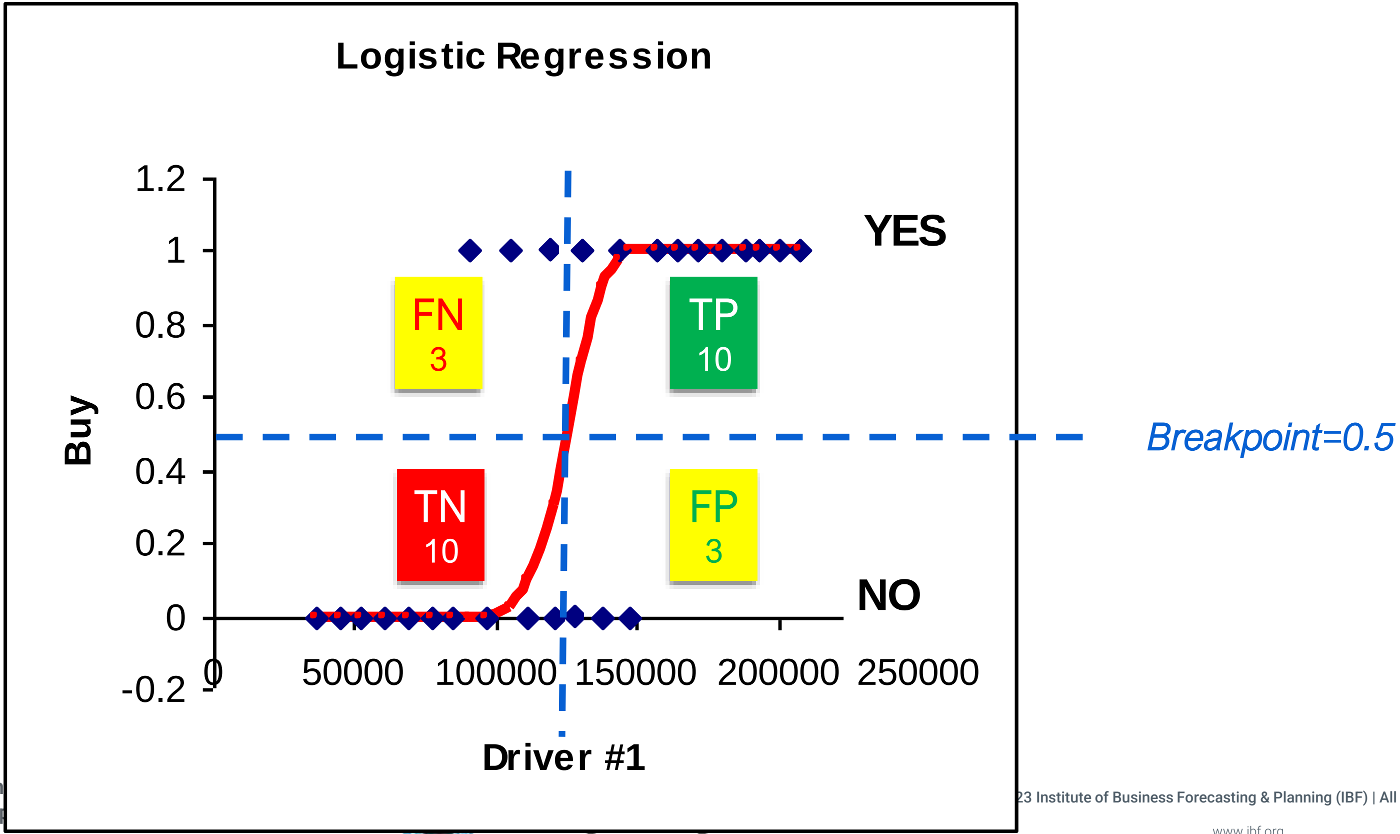
LOGISTIC REGRESSION

ODDS AND PROBABILITIES

- **Odds**: as in horse racing ("odds of winning")
- **Odds and odds ratios**
- **Relationship of Odds to Probability**
  - Odds = P/Q = P/(1-P)
  - P = Odds/(1 + Odds)

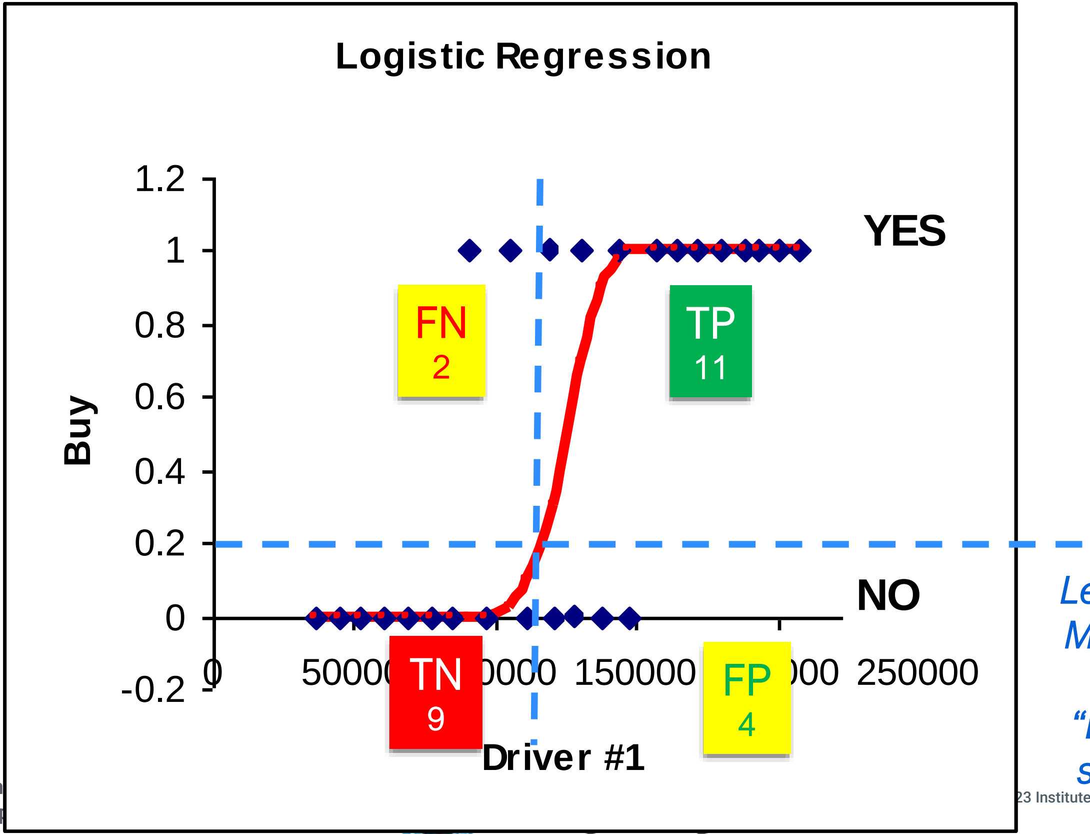
LOGISTIC REGRESSION

CHOOSING BREAKPOINT BETWEEN YES AND NO



LOGISTIC REGRESSION

CHOOSING BREAKPOINT BETWEEN YES AND NO

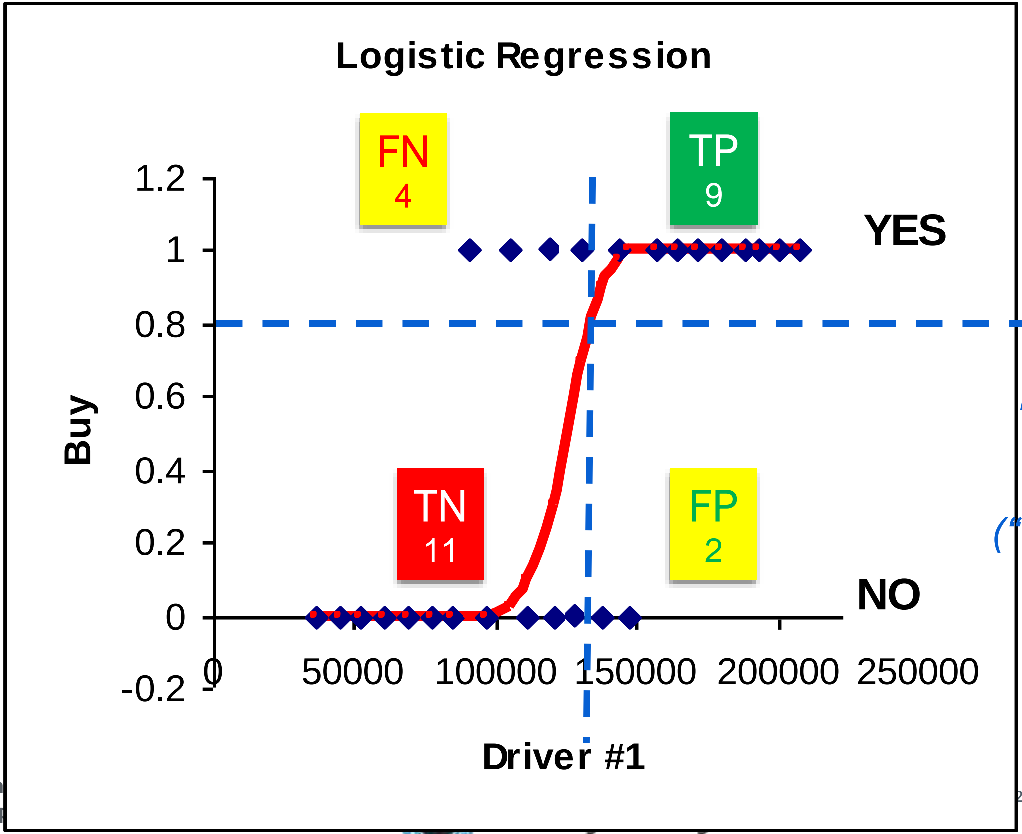


Breakpoint=0.2
Less risk of False Negatives
More risk of False Positives

"Don't miss an opportunity since e-mail is so cheap!"

LOGISTIC REGRESSION

CHOOSING BREAKPOINT BETWEEN YES AND NO

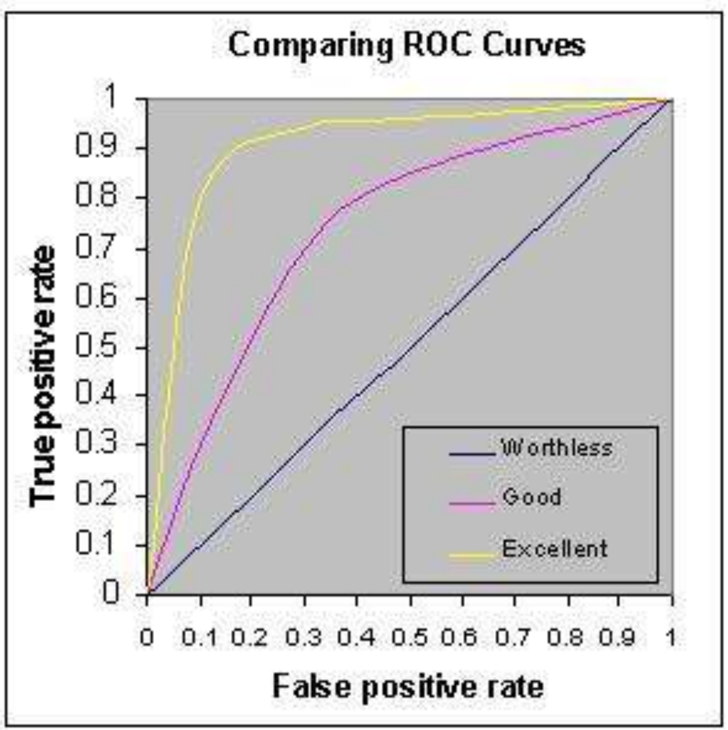


Breakpoint=0.8
Less risk of False Positives
More risk of False Negatives

Useful if cost of FP is high
("the gifts for this promotion are expensive!")

LOGISTIC REGRESSION

ROC CURVE



http://gim.unmc.edu/dxtests/RoC3.htm

- AUC (Area Under Curve): higher curve is better
- Choice between False Positives and False Negatives can be trade-off between relative risks
- TPR=TP/(TP+FN)
- FPR=FP/(FP+TN)

LOGISTIC REGRESSION

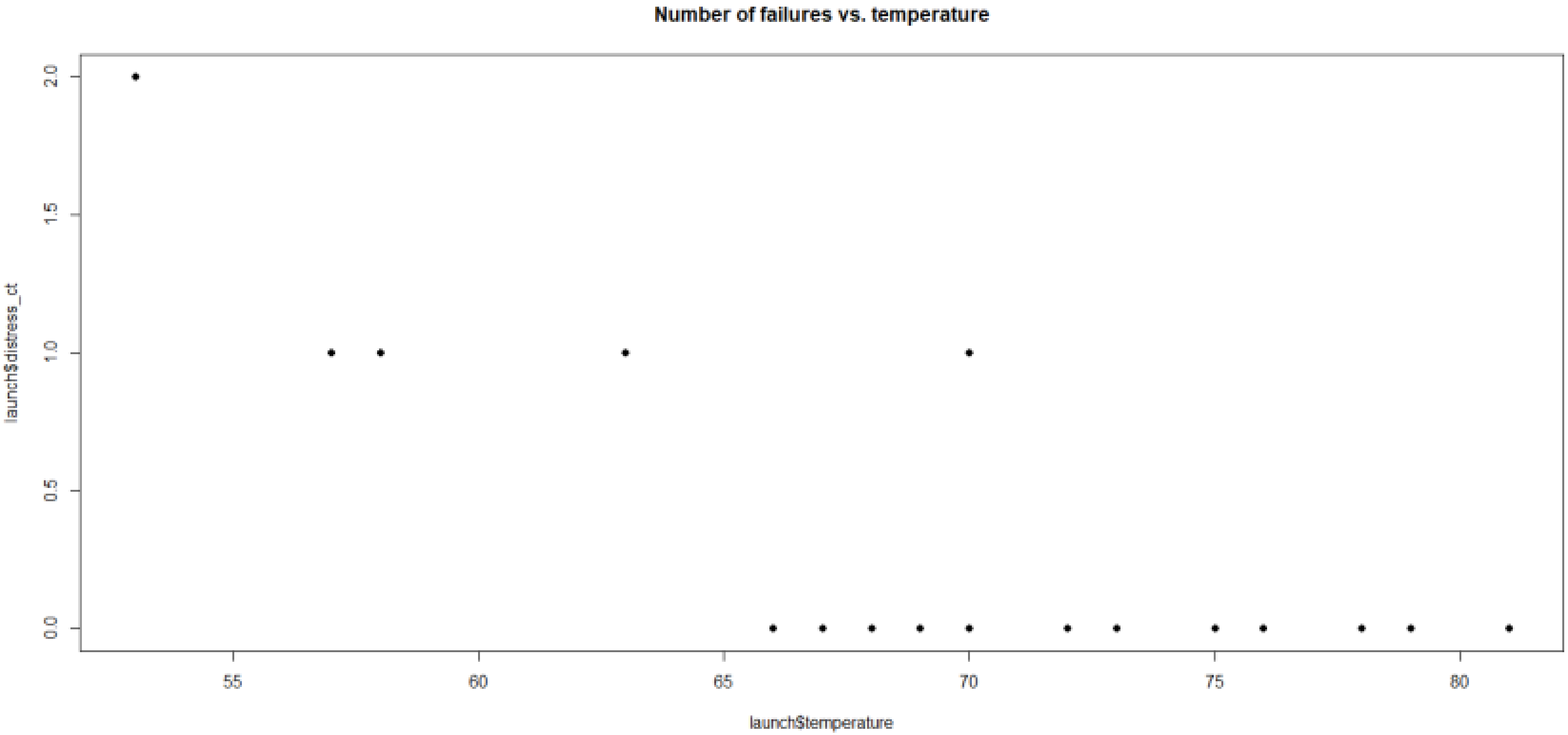
APPLICATIONS IN FORECASTING

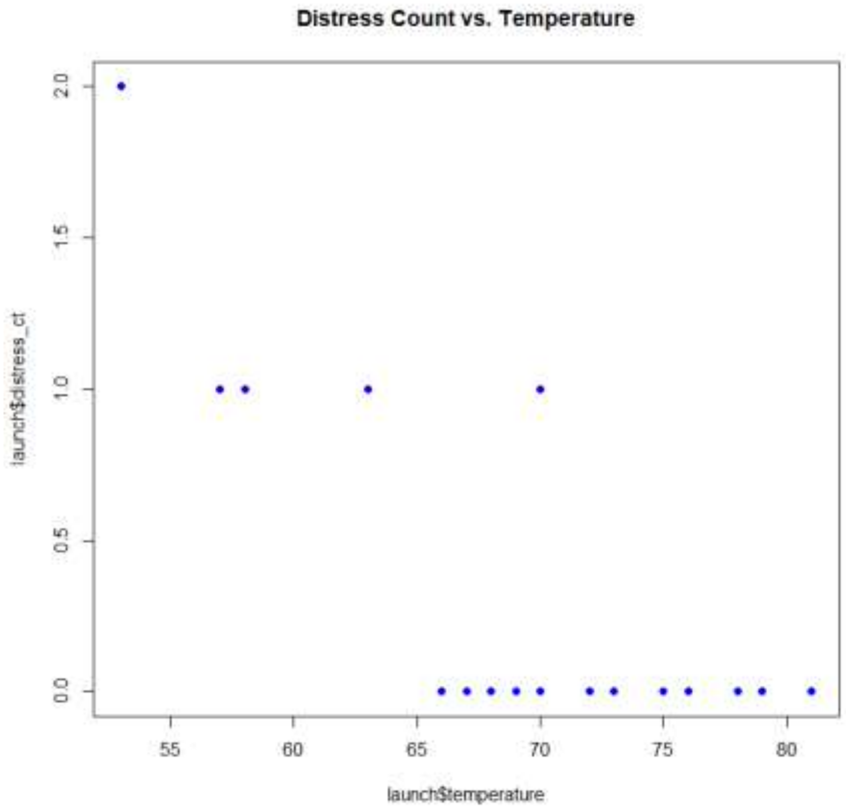
Goal	Process
To forecast event	• Estimate logit (log-odds) model • Use the logit (log-odds) equation to “score” out-of-sample records • Convert log-odds predicted value to probability of event
To apply probability of event as forecast driver	• Aggregate customers within class to get average probability of event • Apply aggregate: --If assembled or scored over history, as a class-level explanatory variable in cause/effect forecasts --If static analysis suggests departure from a trend, as a class-level adjustment to forecast values

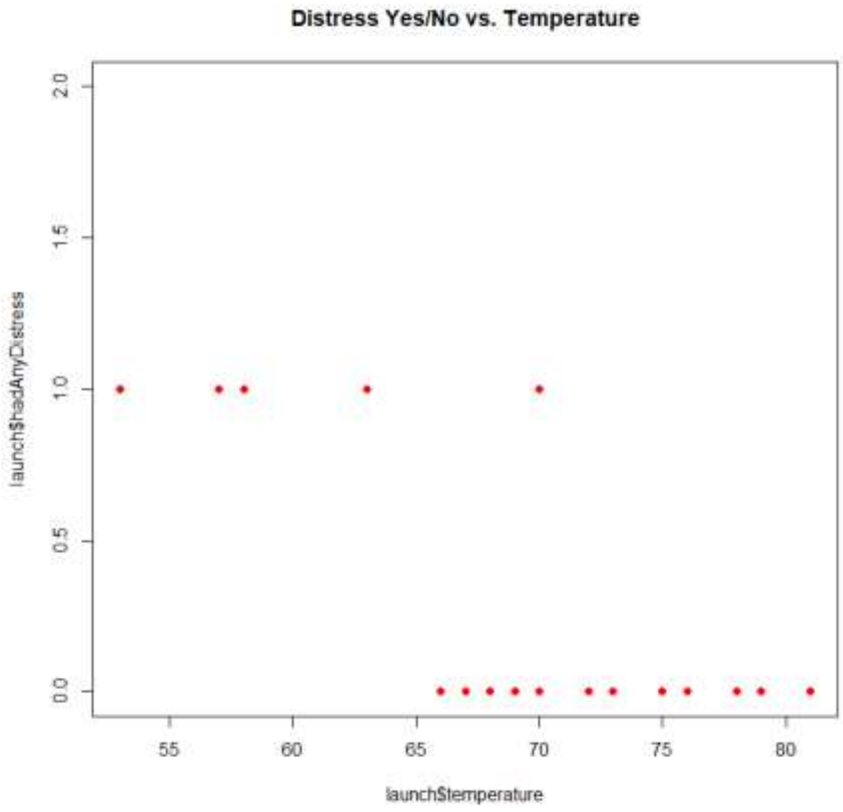
DEMO: LOGISTIC REGRESSION

HTTP://CERESANALYTICS.COM/R_FOR_IBF_BOOTCAMP/IBF_LOGISTIC.R

WEB APP: HTTPS://SBTHESIS.COM/SHINY_IBF

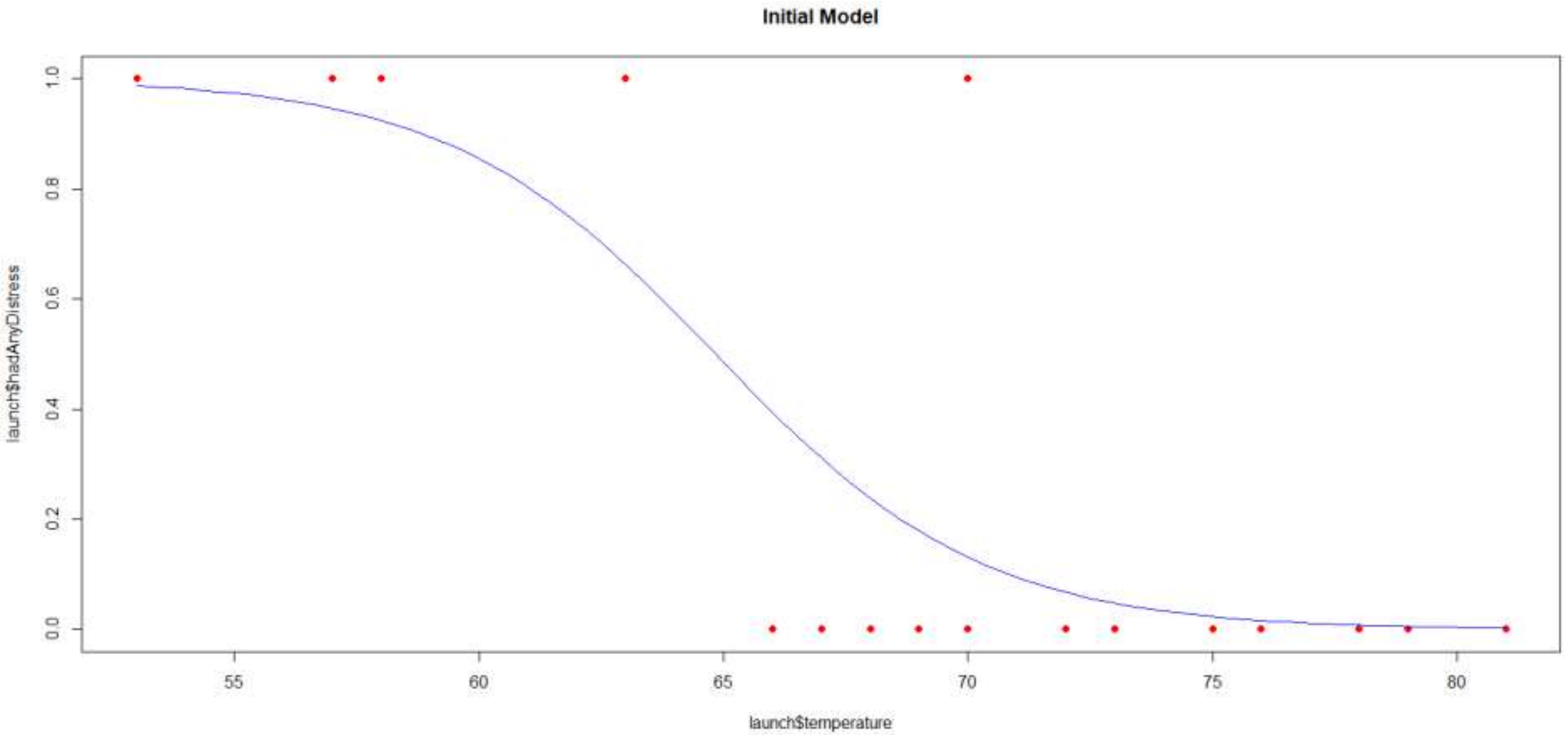

Number of failures vs. temperature
launch$distress_ct
0.0
0.5
1.0
1.5
2.0
55
60
65
70
75
80
launch$temperature


Distress Count vs. Temperature
launch$distress_ct
launch$temperature
0.0
0.5
1.0
1.5
2.0
55
60
65
70
75
80

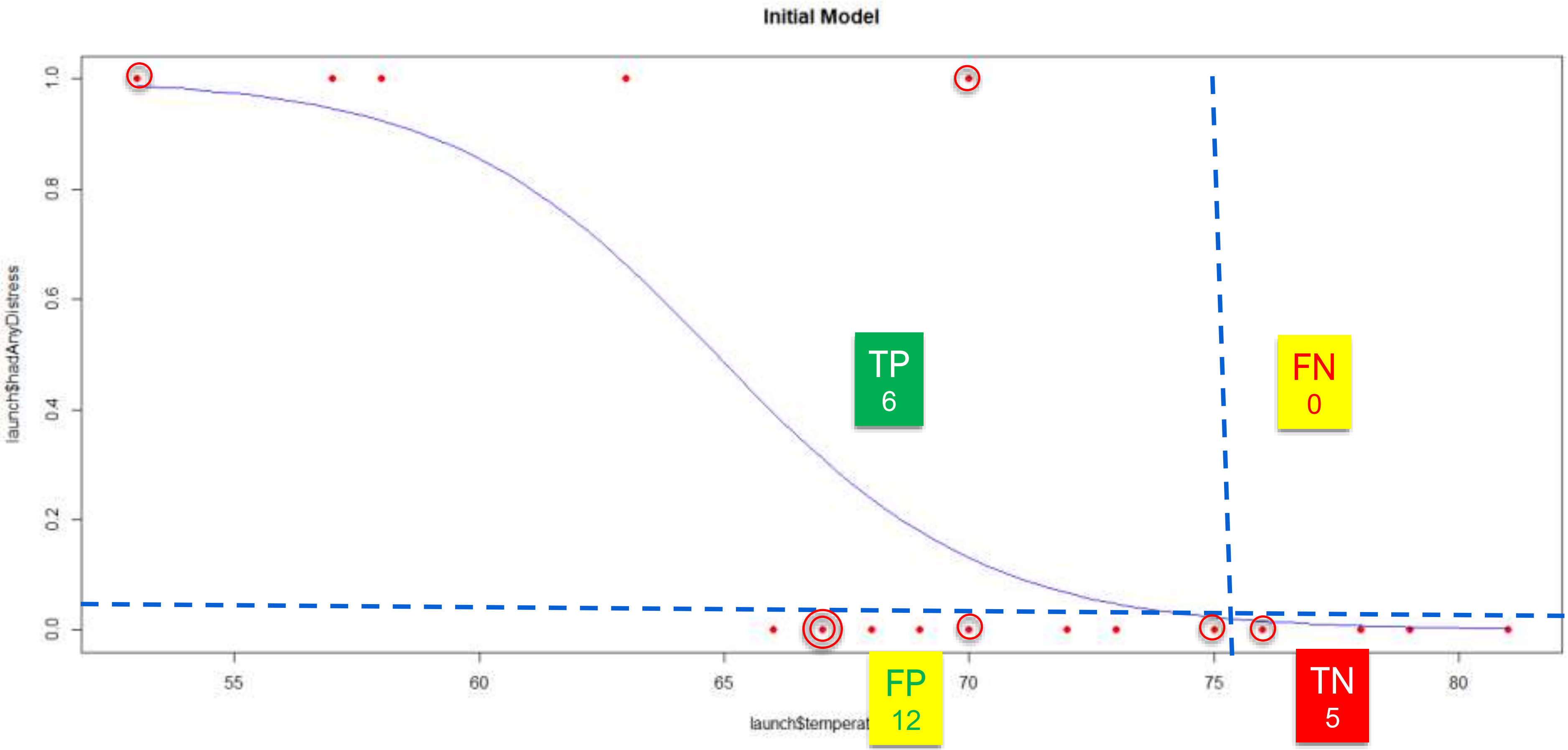

Distress Yes/No vs. Temperature
launch$hadAnyDistress
launch$temperature
0.0
0.5
1.0
1.5
2.0
55
60
65
70
75
80

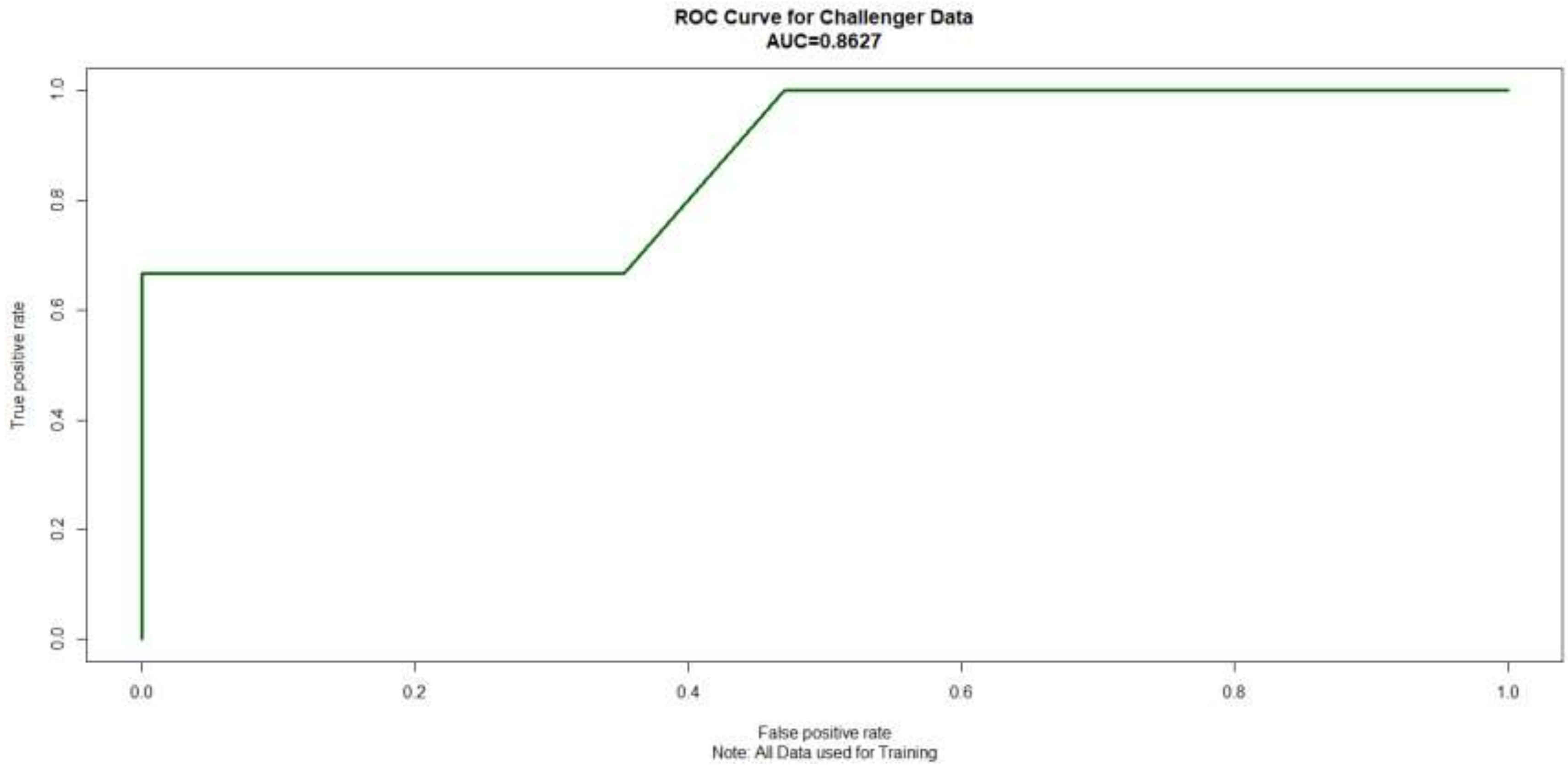
```
Call:
glm(formula = hadAnyDistress ~ temperature, family = binomial(link = "logit"),
    data = launch)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.00214  -0.57815  -0.21802   0.04401   2.01706

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  23.7750    11.8204   2.011   0.0443 *
temperature  -0.3667     0.1752  -2.093   0.0363 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 26.402  on 22  degrees of freedom
Residual deviance: 14.426  on 21  degrees of freedom
AIC: 18.426

Number of Fisher Scoring iterations: 6

[1] "McFadden is pseudo-R-square:"
fitting null model for pseudo-r2
        llh     llhNull          G2    McFadden        r2ML        r2CU
 -7.2128958 -13.2011833  11.9765750   0.4536175   0.4059077   0.5945565
```

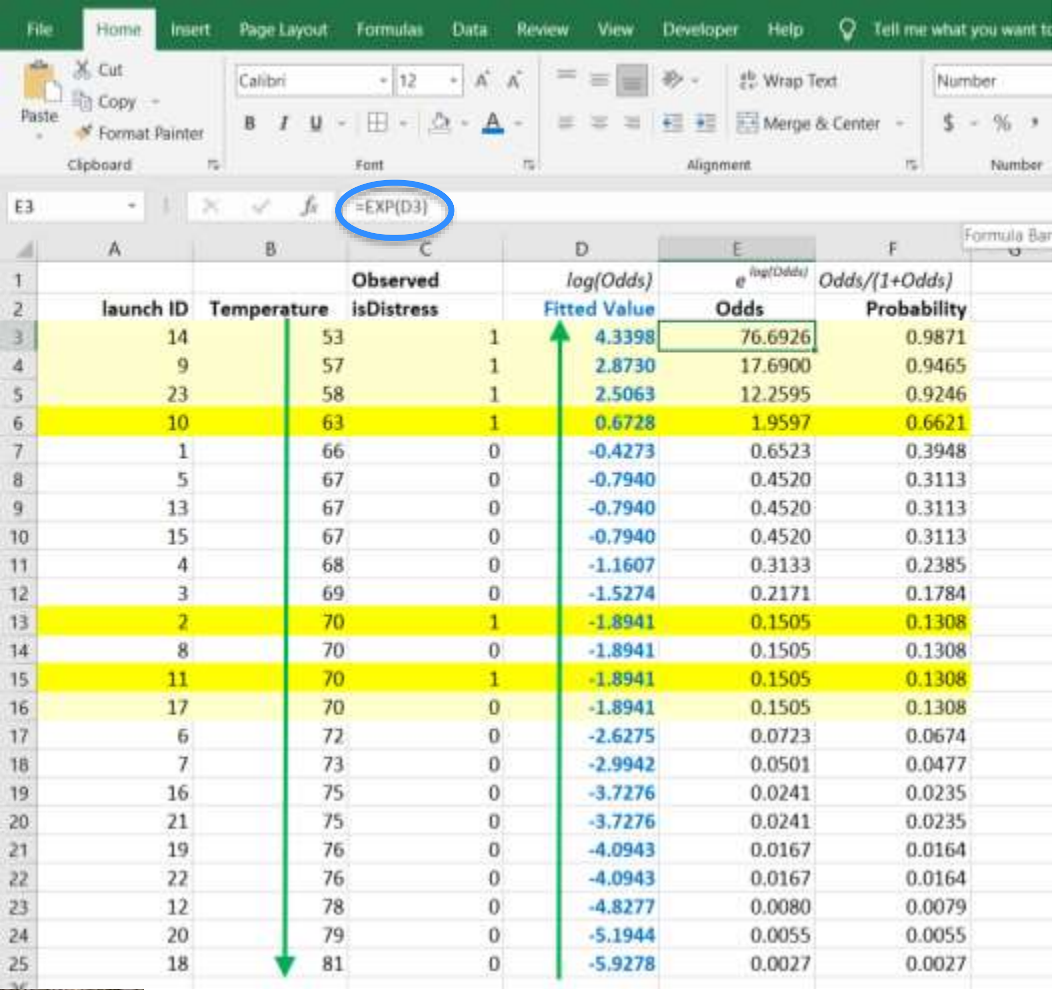
What does fit look like?



What does fit look like?




ROC Curve for Challenger Data
AUC=0.8627
True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
False positive rate
Note: All Data used for Training



A	B	C	D	E	F
		Observed	*log(Odds)*	$e^{log(Odds)}$	*Odds/(1+Odds)*
launch ID	Temperature	isDistress	Fitted Value	Odds	Probability
14	53	1	4.3398	76.6926	0.9871
9	57	1	2.8730	17.6900	0.9465
23	58	1	2.5063	12.2595	0.9246
10	63	1	0.6728	1.9597	0.6621
1	66	0	-0.4273	0.6523	0.3948
5	67	0	-0.7940	0.4520	0.3113
13	67	0	-0.7940	0.4520	0.3113
15	67	0	-0.7940	0.4520	0.3113
4	68	0	-1.1607	0.3133	0.2385
3	69	0	-1.5274	0.2171	0.1784
2	70	1	-1.8941	0.1505	0.1308
8	70	0	-1.8941	0.1505	0.1308
11	70	1	-1.8941	0.1505	0.1308
17	70	0	-1.8941	0.1505	0.1308
6	72	0	-2.6275	0.0723	0.0674
7	73	0	-2.9942	0.0501	0.0477
16	75	0	-3.7276	0.0241	0.0235
21	75	0	-3.7276	0.0241	0.0235
19	76	0	-4.0943	0.0167	0.0164
22	76	0	-4.0943	0.0167	0.0164
12	78	0	-4.8277	0.0080	0.0079
20	79	0	-5.1944	0.0055	0.0055
18	81	0	-5.9278	0.0027	0.0027

JUST IN CASE YOU NEED THE MATH...

MACHINE LEARNING ALTERNATIVE
DECISION TREES (CHAID)

5

DECISION TREE

- What it is/how it works
  - Predictive technique that—through machine learning—identifies combinations of criteria that distinguish class categories (CHAID: yes/no, e.g. customer attrition, credit card default) or a range of scalar levels (C&RT: historical demand for forecasting)
  - Statistics vs. Machine Learning—generally accepted as machine learning
- How to evaluate your results
  - “Node purity” (classification), forecast error rate (scalar), other
- How to apply your results
  - Often a tree can simply be informative in understanding your data, e.g.
    - For classifications:
      - the probability of an outcome (e.g., 70% yes, 30%no) or
      - assign a value based on a breakpoint (e.g., 70% yes means the predicted value is yes)
    - For predictive modeling of scalar values (demand) the average for the categories can be a reasonable forecast
- Example
  - CHAID for telecom churn

DECISION TREE METHOD: CHAID *

CHAID (CHi-square Automated Interaction Detector) is a decision tree method of statistics

- The dependent variable (or outcome to be explained) is usually categorical, like a yes/no event or group classification, though other decision tree algorithms can explain scalar values like the level of demand
- Predictors can take several forms:

Form of Dependent Variable	Examples	
Categorical	Yes/No	Red/Blue/Green/Yellow
Rank-Ordered	Yes/Maybe/No	First Choice/Second Choice/Not Chosen
Scalar	93.2%, 5%, 47.8%	$10,325; $32,500; $50,000; 65,789

** Other Decision Tree methods, exist, such as Classification & Regression Trees (C&RT), known in its commercial implementation as CART*

How

Institute of Business Forecasting & Planning

CHAID—HOW IT WORKS

1. At the top of the tree (the first node), CHAID searches for the predictor that corresponds most strongly to the outcome.

- If the predictor is scalar, CHAID searches for breakpoint(s) that make the strongest categories.
- For example, if telecom customers change providers ("churn") when their bills are high, then great (strong) CHAID nodes would show "Yes" for churn to coincide with the customer's bill level.
- Conversely, "No" would coincide with low bill levels:

	Telecom Churn (Quarter)		
	Yes	No	Total
Bill > $60	N= 80	N= 2	82
Bill < $60	N= 3	N= 915	918
Total	83	917	1000

CHAID—HOW IT WORKS

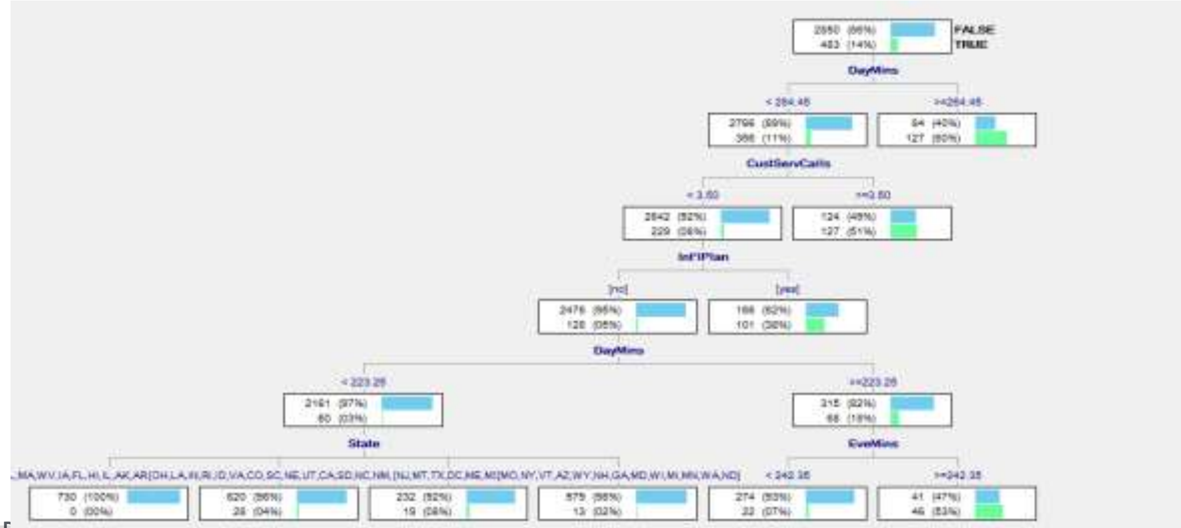
In contrast, a mediocre (weak) CHAID nodes shows a more even distribution.

- If the correspondence of the predictor and the outcome isn't statistically significant, then the nodes would not be defined at all.
- A weak but significant example:

	Telecom Churn (Quarter)		
	Yes	No	Total
Bill > $40	N= 43	N= 445	498
Bill < $40	N= 40	N= 472	512
Total	83	917	1000

CHAID—HOW IT WORKS

2. After the first split, new nodes emerge below that contain the results of the first classification rule. That is:
   - Each category of the outcome branches into a new node
     (below, the branches go to the right to favor "Yes", and left to favor "No")
   - The resulting nodes are each a potential "parent" that can, in turn, branch to more "children" as the search for classifiers continues



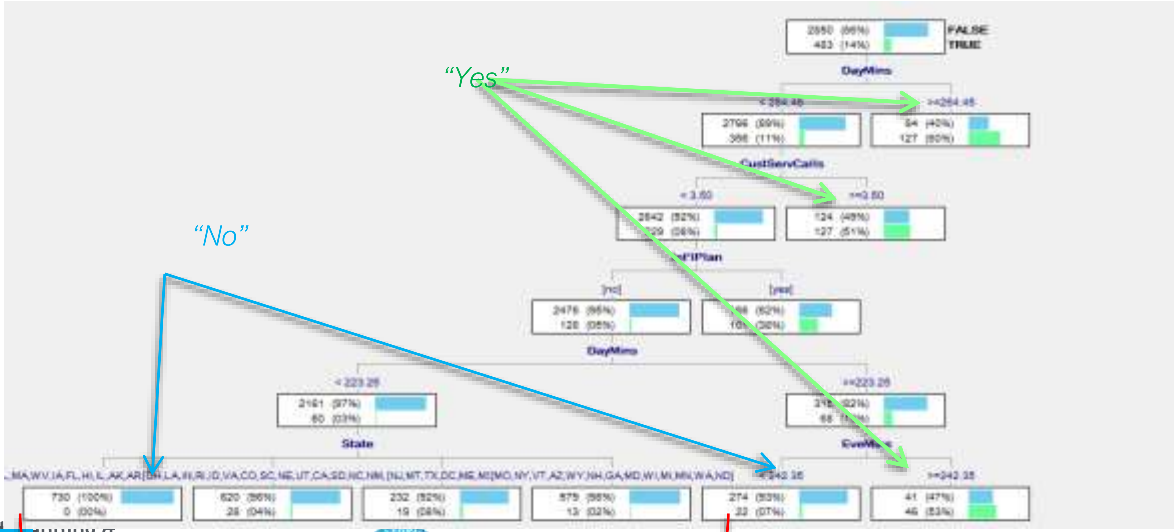
CHAID—HOW IT WORKS

3. The tree algorithm will attempt to split each node until the analyst's pre-defined criteria tell it to stop. These criteria include thresholds for statistical significance and segment size.

CHAID—HOW IT WORKS

4. Because the predictors aren't perfect, the nodes won't be 100% pure for the outcome (e.g., all "Yes" for churning, or all "No").

- But a good tree will show a final collection of nodes in which some are lopsided toward "Yes“
- Others should be lopsided toward "No“

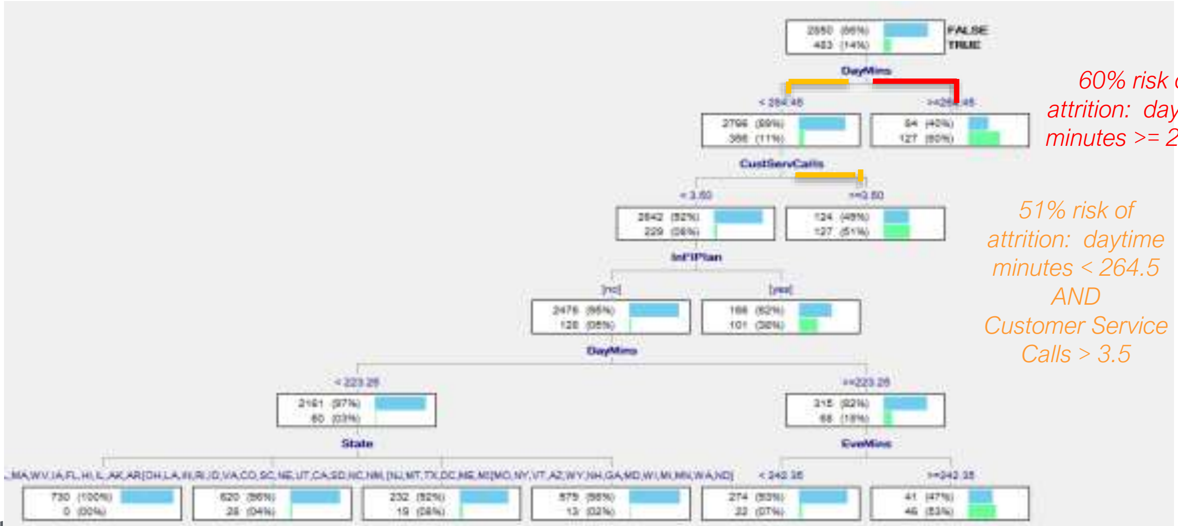


Note: Suspect over-fitting and redo with regions

CHAID—HOW IT WORKS

5. When complete, inspection of the tree discloses key, predictive interactions

- One simply traces a series of branches from the root to the terminal node
- The predictors and their values are identified on the branches
- The number of affected observations and %splits are stated on the terminal node.



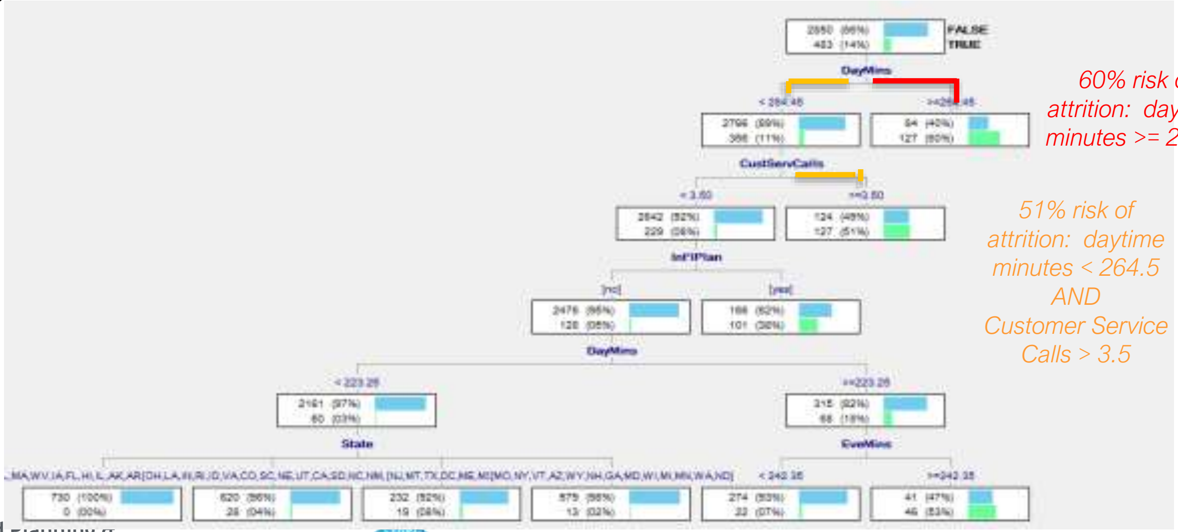
60% risk of attrition: daytime minutes >= 264.5

51% risk of attrition: daytime minutes < 264.5 AND Customer Service Calls > 3.5

CHAID—EVALUATING RESULTS

5. When complete, inspection of the tree discloses key, predictive interactions

- The number of affected observations and % splits are stated on the terminal nodes
- **Ideally, nodes will show high degrees of contrast**
- In theory, for yes/no classifications, one can construct ROC curves just as for logistic regression



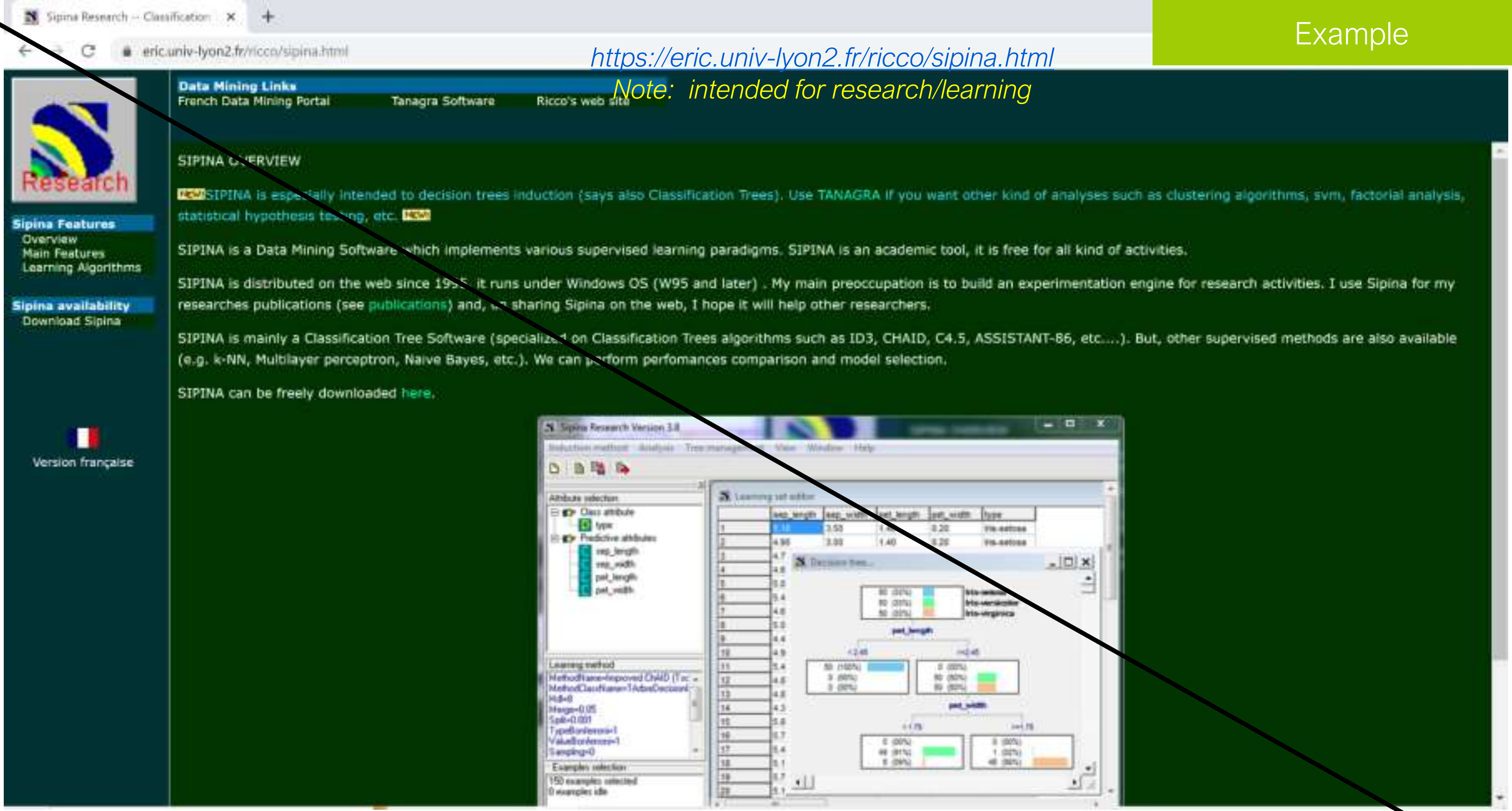
60% risk of attrition: daytime minutes >= 264.5

51% risk of attrition: daytime minutes < 264.5 AND Customer Service Calls > 3.5

Example

https://eric.univ-lyon2.fr/ricco/sipina.html

Note: intended for research/learning



Institute of Business Forecasting & Planning

https://sipina-arbres-de-decision.blogspot.com/

Sipina - Decision trees - Data Mining

Free data mining software for inducing decision trees

Welcome | Releases | Methods | Abilities | References

Connections

Home page

Download

Other links

Sipina website (EN)

Tanagra software

Course materials

Subect to check

Sipina

SIPINA is free Data Mining software specialized in the induction of decision trees. Curiously, it is one of the very rare open access tools integrating interactive features when building a decision tree. Features which, however, make this method so valuable in a data mining activity.

SIPINA also implements other supervised methods. But its interest is less in this context. Since the development and distribution of TANAGRA (January 2004), I systematically recommend using the latter. It includes not only supervised methods but also a large majority of statistical and data analysis techniques such as factorial analyses, automatic classification, etc., and the possibility of making them cooperate with each other.

The different versions of SIPINA have been available on the web since 1995. The current version has barely evolved since 2000. It is nevertheless distributed because, as I said above, there are very few free equivalents in the world. The English distribution site is still regularly consulted to this day, and the software downloaded. There must be a reason for this. So I decided to document it a little more, an aspect totally neglected at the time of its development. I am also rediscovering many features imagined, experienced, and ultimately known only to me... so much so that everyone benefits from them.

Configured judiciously, SIPINA can process very large volumes (several million observations, several thousand variables) while retaining its interactive functionalities.

This site brings together all the material concerning SIPINA. Another notable development is that it is entirely in French, the initial site having always been exclusively in English. The software remains in English, but the key words are relatively easy to understand.

SIPINA is completely free, whatever the context of use.

Ricco Rakotomalala .

Selection of documents.

Tree methodology

Video tutorials

Sipina under Linux

Spreadsheet add-ins

Excel 2007 & 2010

Open & Libre Office

Example

About the academic developer…

Ricco Rakotomalala - Data Science

"I think of data scientists as knowing more about statistics than computer scientists and more about computer science than statisticians.", Michael O'Connell (KDnuggets Interview - May, 2014).

Accueil | Publications

Favoris

- Profil Linkedin
- Master SISE
- Chaîne YouTube
- Cours Machine Learning
- Tutoriels Data Science

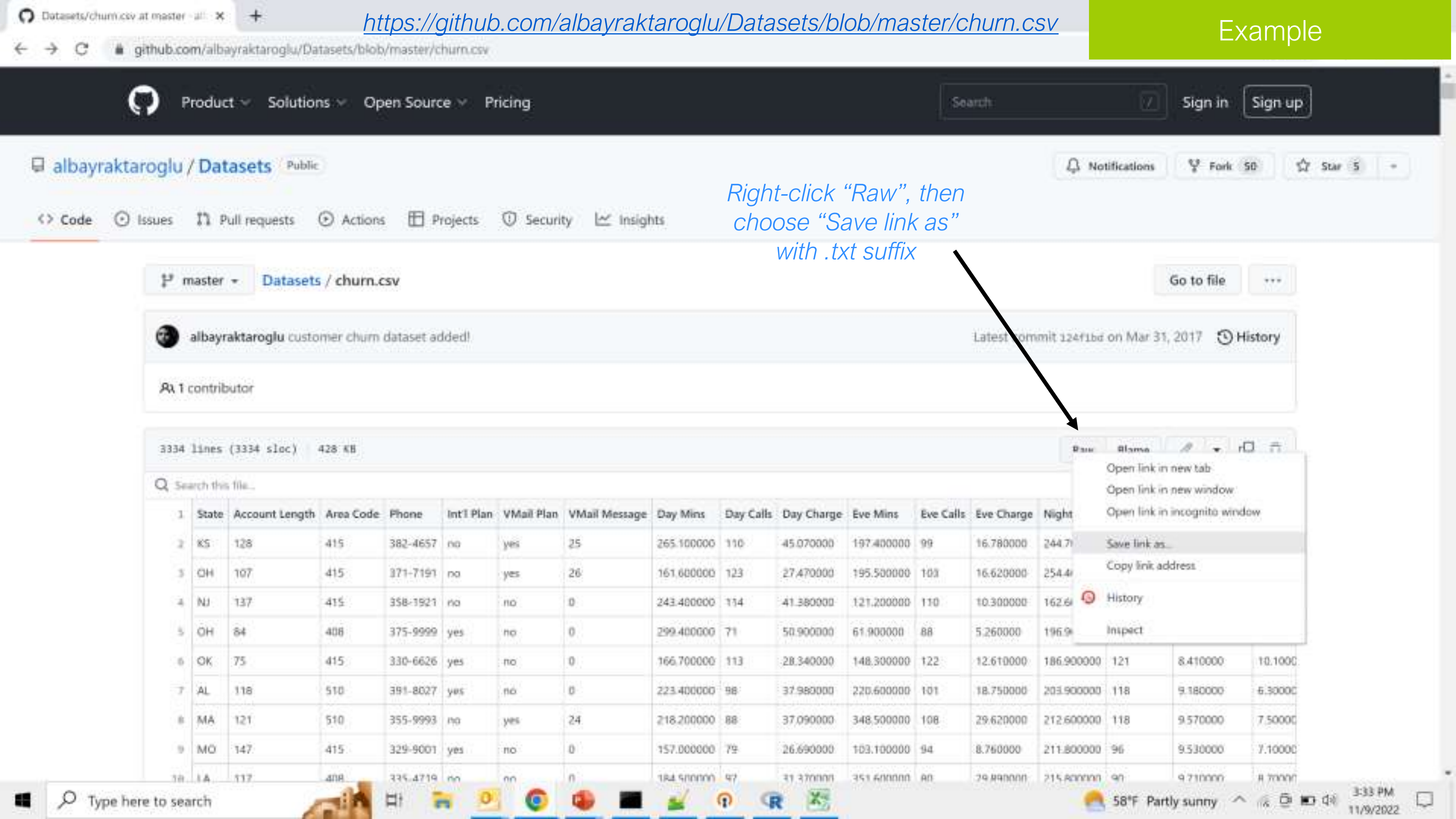
Logiciels libres

- Tanagra
- Sipina

- Maître de Conférences en Informatique - Section CNU 27
- Université Lumière Lyon 2 - Institut de la Communication (ICOM)
- Responsable du Master SISE - Master 2 - Statistique et Informatique pour la Science des donnéEs
- Enseignements (voir cours) :
  1. méthodes statistiques, modélisation statistique, économétrie, statistique exploratoire ;
  2. méthodes de data science, machine learning, data mining, analyse prédictive, big data analytics ;
  3. text mining, web mining, analyse des réseaux sociaux ;
  4. langages de programmation (vba, java, delphi, c++, c#, R, Python), programmation pour les big data, etc.
- Doctorat en apprentissage statistique (*machine learning*)
- Spécialisation en recherche : Data Science - Big Data Analytics - Data Mining - Machine Learning - Statistical Modeling - Exploratory Data Analysis
- A écrit et mis en ligne de nombreux ouvrages, tutoriels et supports de cours à destination des *data scientists* (voir publications)
- Chaîne YouTube dédiée aux principes et à la pratique de la data science, du machine learning, et de la statistique exploratoire
- A développé et mis en ligne des logiciels de statistique et de data mining, en particulier Sipina (V3) et Tanagra

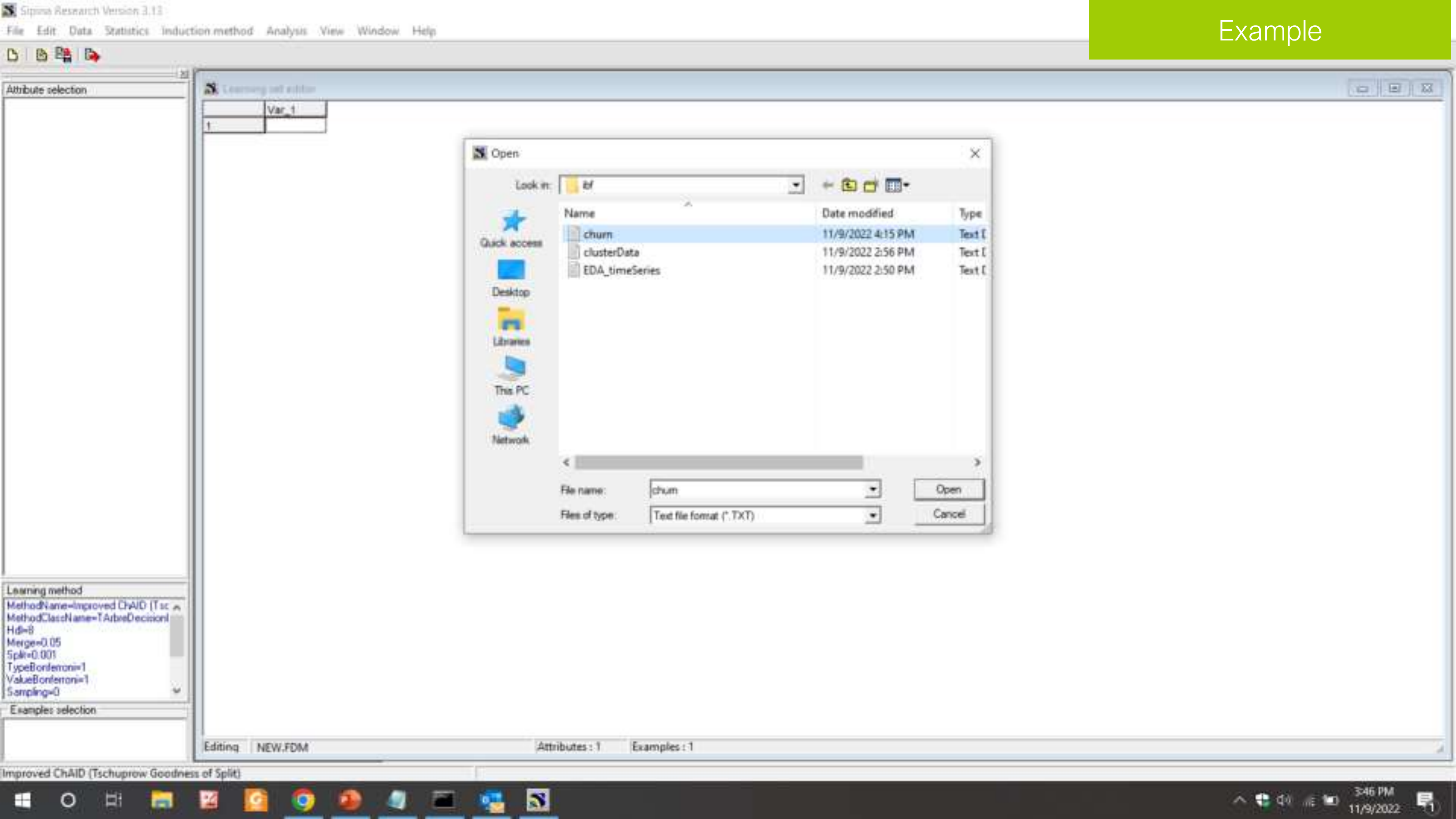
https://github.com/albayraktaroglu/Datasets/blob/master/churn.csv

Example

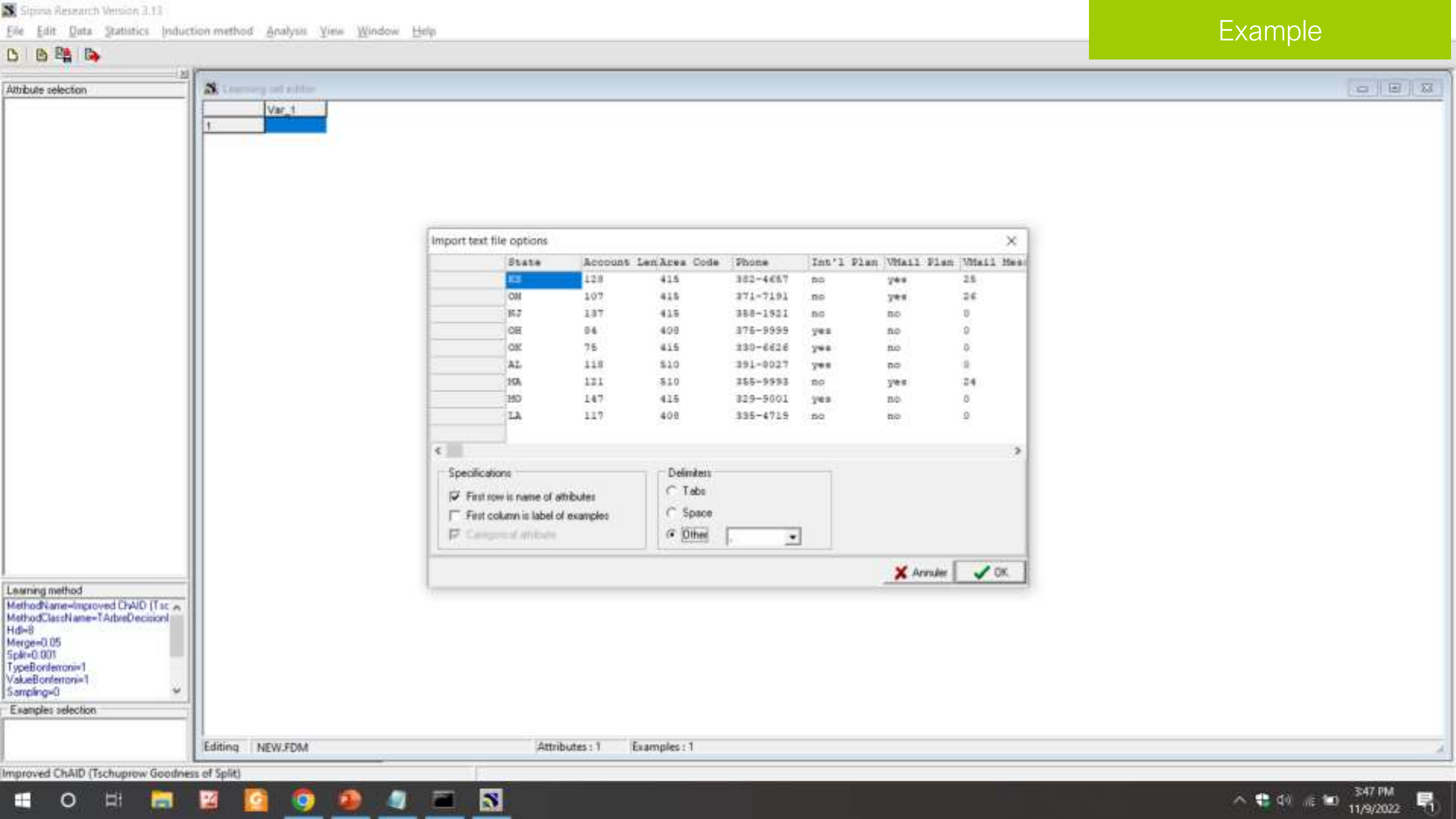


Right-click “Raw”, then choose “Save link as” with .txt suffix

Example

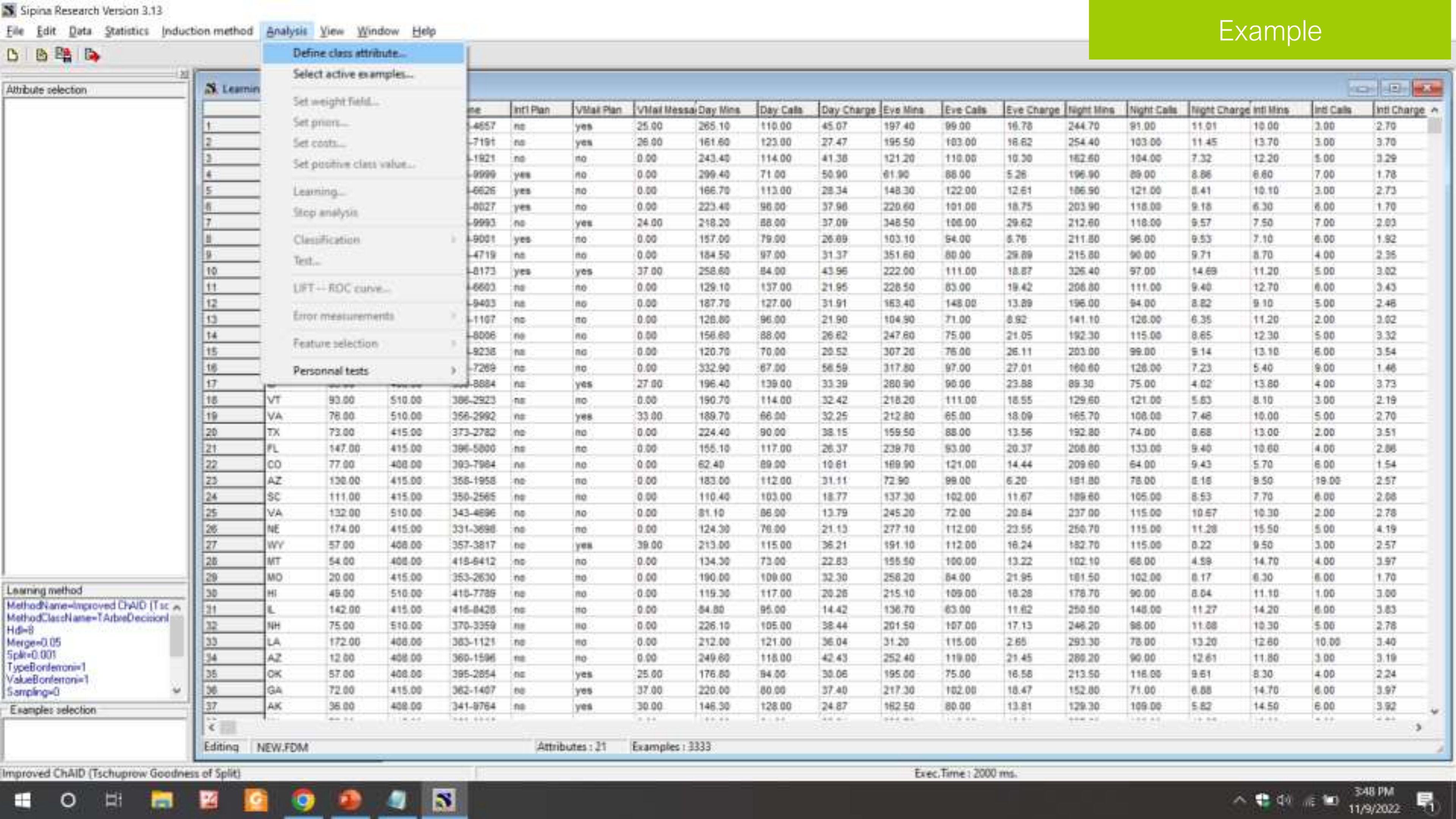

Sipina Research Version 3.13
File Edit Data Statistics Induction method Analysis View Window Help
Attribute selection
Var_1
Open
Look in: ibf
Quick access
Desktop
Libraries
This PC
Network
Name
Date modified
Type
churn 11/9/2022 4:15 PM
clusterData 11/9/2022 2:56 PM
EDA_timeSeries 11/9/2022 2:50 PM
File name: churn
Files of type: Text file format (*.TXT)
Open
Cancel
Learning method
MethodName=Improved ChAID (Tsc
MethodClassName=TArbreDecision
Hdl=8
Merge=0.05
Split=0.001
TypeBonferroni=1
ValueBonferroni=1
Sampling=0
Examples selection
Editing NEW.FDM
Attributes : 1
Examples : 1
Improved ChAID (Tschuprow Goodness of Split)
3:46 PM
11/9/2022

Example

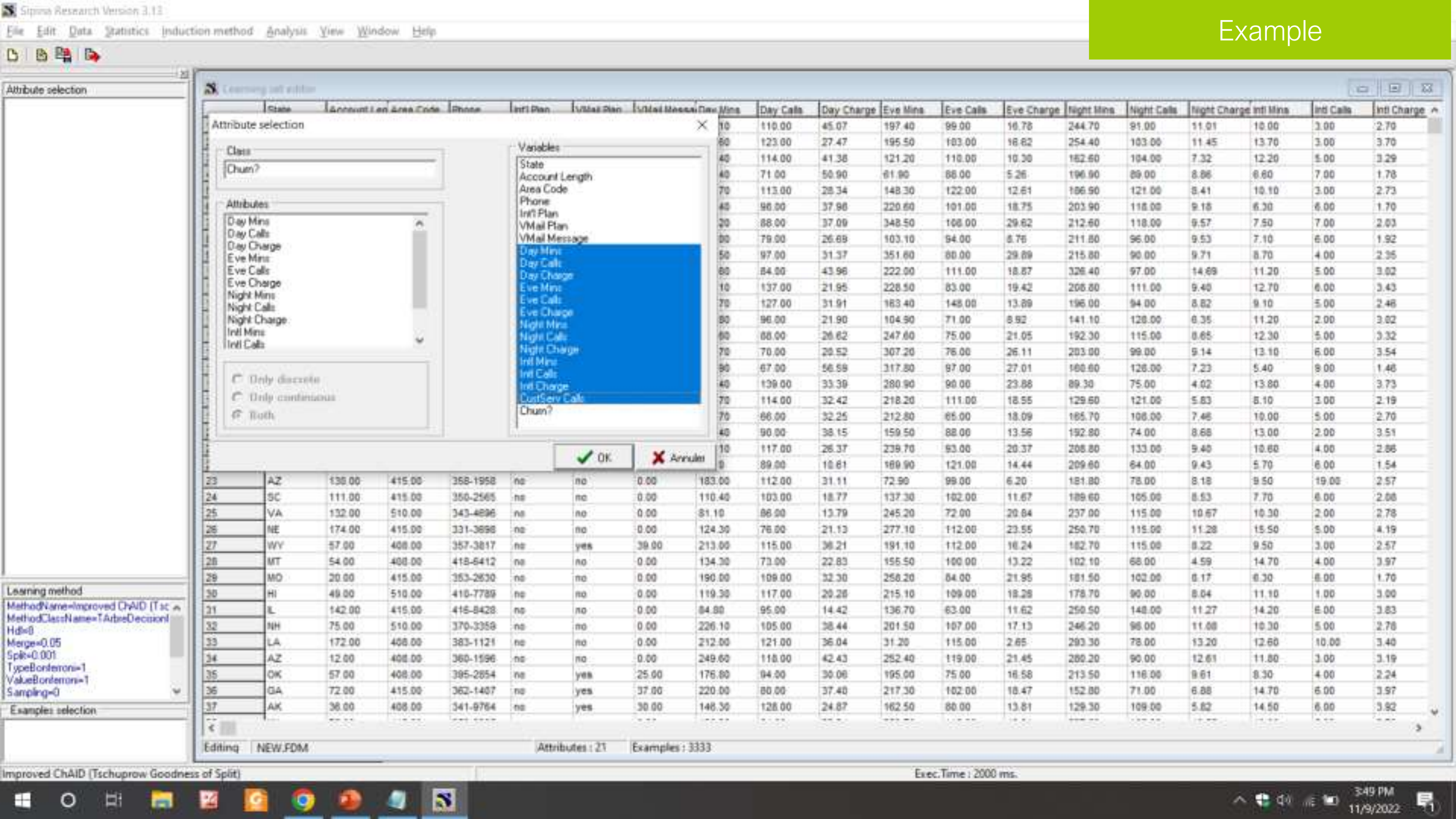


Sipina Research Version 3.13
File Edit Data Statistics Induction method Analysis View Window Help
Attribute selection
Var_1
Import text file options
State Account Len Area Code Phone Int'l Plan VMail Plan VMail Mes
KS 128 415 382-4657 no yes 25
OH 107 415 371-7191 no yes 26
NJ 137 415 358-1921 no no 0
OH 84 408 375-9999 yes no 0
OK 75 415 330-6626 yes no 0
AL 118 510 391-8027 yes no 0
MA 121 510 355-9993 no yes 24
MO 147 415 329-9001 yes no 0
LA 117 408 335-4719 no no 0
Specifications
First row is name of attributes
First column is label of examples
Categorical attribute
Delimiters
Tabs
Space
Other
Annuler
OK
Learning method
MethodName=Improved ChAID (Tsc
MethodClassName=TArbreDecision
Hdl=8
Merge=0.05
Split=0.001
TypeBonferroni=1
ValueBonferroni=1
Sampling=0
Examples selection
Editing NEW.FDM
Attributes : 1
Examples : 1
Improved ChAID (Tschuprow Goodness of Split)
3:47 PM
11/9/2022

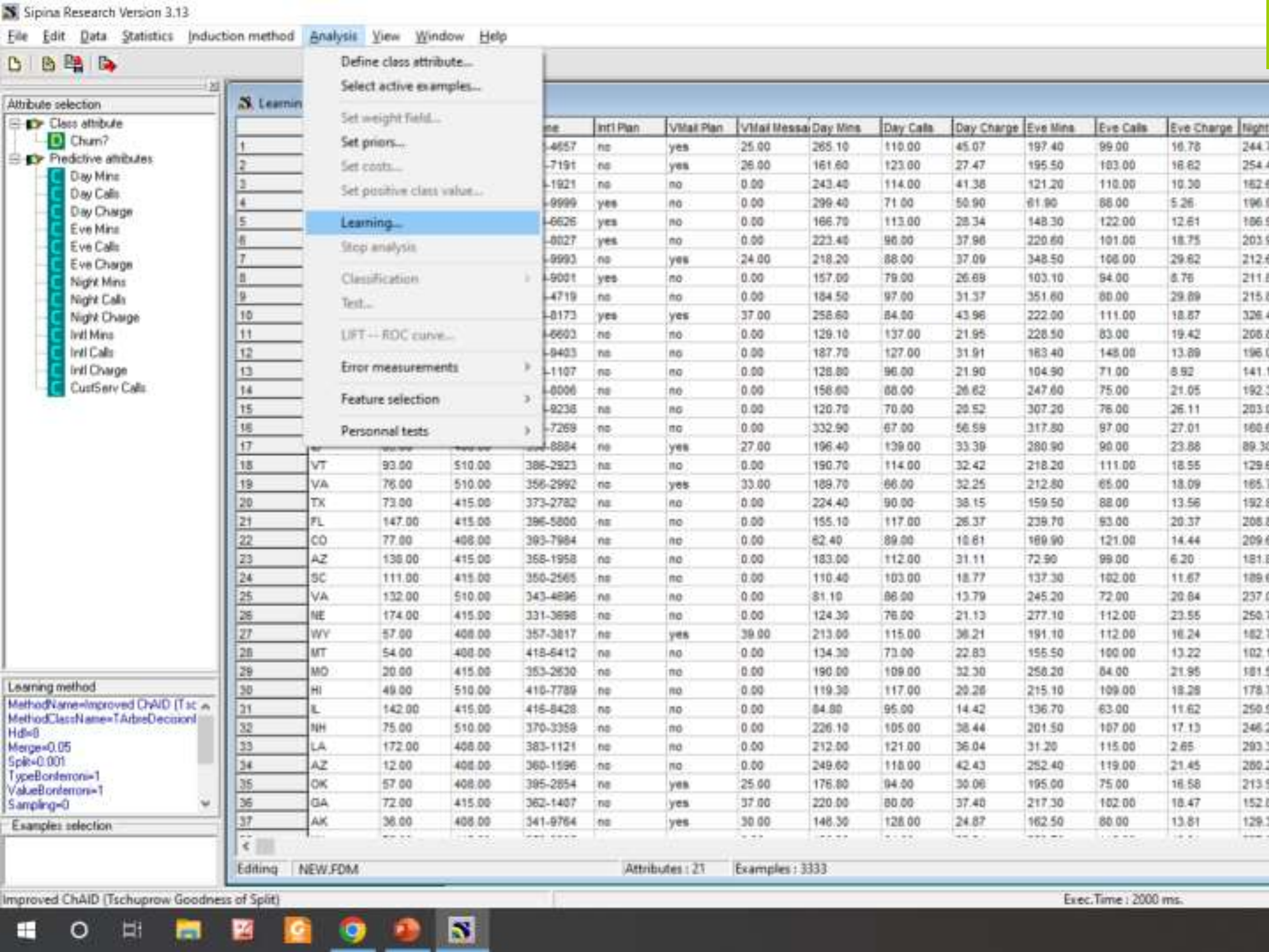
Example


Sipina Research Version 3.13
File Edit Data Statistics Induction method Analysis View Window Help
Define class attribute...
Select active examples...
Set weight field...
Set priors...
Set costs...
Set positive class value...
Learning...
Stop analysis
Classification
Test...
LIFT -- ROC curve...
Error measurements
Feature selection
Personnal tests
Attribute selection
Learning method
MethodName=Improved ChAID (Tsc
MethodClassName=TArbreDecisionI
Hdl=8
Merge=0.05
Split=0.001
TypeBonferroni=1
ValueBonferroni=1
Sampling=0
Examples selection
Editing NEW.FDM Attributes : 21 Examples : 3333
Improved ChAID (Tschuprow Goodness of Split)
Exec.Time : 2000 ms.
3:48 PM
11/9/2022

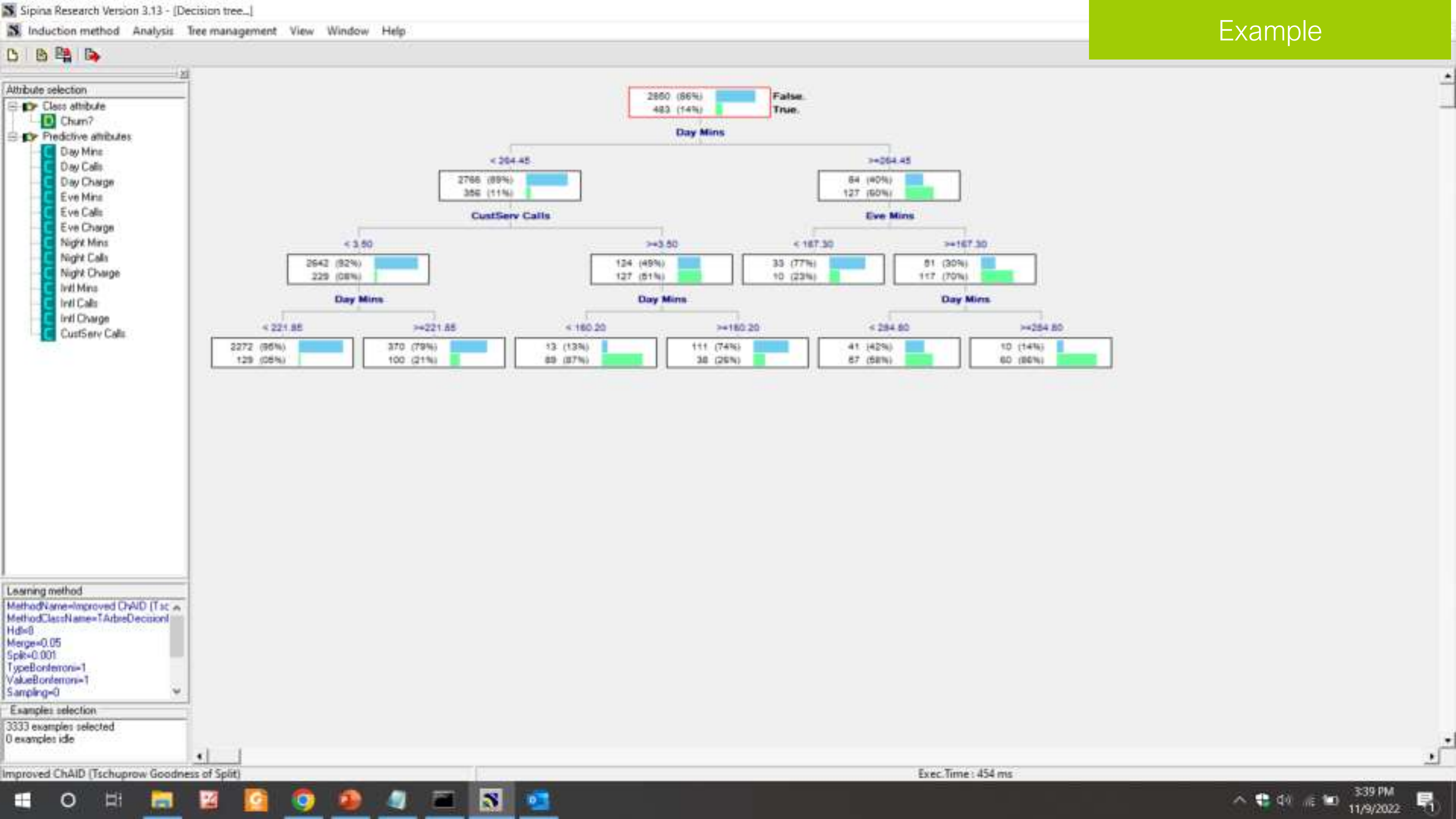
Example


Sipina Research Version 3.13
File Edit Data Statistics Induction method Analysis View Window Help
Attribute selection
Learning set editor
Attribute selection
Class
Churn?
Attributes
Day Mins
Day Calls
Day Charge
Eve Mins
Eve Calls
Eve Charge
Night Mins
Night Calls
Night Charge
Intl Mins
Intl Calls
Only discrete
Only continuous
Both
Variables
State
Account Length
Area Code
Phone
Int'l Plan
VMail Plan
VMail Message
Day Mins
Day Calls
Day Charge
Eve Mins
Eve Calls
Eve Charge
Night Mins
Night Calls
Night Charge
Intl Mins
Intl Calls
Intl Charge
CustServ Calls
Churn?
OK
Annuler
Learning method
MethodName=Improved ChAID (Tsc
MethodClassName=TArbreDecision
Hdl=8
Merge=0.05
Split=0.001
TypeBonferroni=1
ValueBonferroni=1
Sampling=0
Examples selection
Editing
NEW.FDM
Attributes : 21
Examples : 3333
Improved ChAID (Tschuprow Goodness of Split)
Exec.Time : 2000 ms.
3:49 PM
11/9/2022

Example


Sipina Research Version 3.13
File Edit Data Statistics Induction method Analysis View Window Help
Define class attribute...
Select active examples...
Set weight field...
Set priors...
Set costs...
Set positive class value...
Learning...
Stop analysis
Classification
Test...
LIFT -- ROC curve...
Error measurements
Feature selection
Personnal tests
Attribute selection
Class attribute
Churn?
Predictive attributes
Day Mins
Day Calls
Day Charge
Eve Mins
Eve Calls
Eve Charge
Night Mins
Night Calls
Night Charge
Intl Mins
Intl Calls
Intl Charge
CustServ Calls
Learning method
MethodName=Improved ChAID (Tsc
MethodClassName=TArbreDecisionI
Hdl=8
Merge=0.05
Split=0.001
TypeBonferroni=1
ValueBonferroni=1
Sampling=0
Examples selection
Int'l Plan
VMail Plan
VMail Message
Day Mins
Day Calls
Day Charge
Eve Mins
Eve Calls
Eve Charge
Night Mins
Night Calls
Night Charge
Intl Mins
Intl Calls
Intl Charge
Editing
NEW.FDM
Attributes : 21
Examples : 3333
Improved ChAID (Tschuprow Goodness of Split)
Exec.Time : 2000 ms.
3:50 PM
11/9/2022

Example



Sipina Research Version 3.13 - [Decision tree...]
Induction method Analysis Tree management View Window Help
Attribute selection
Class attribute
Churn?
Predictive attributes
Day Mins
Day Calls
Day Charge
Eve Mins
Eve Calls
Eve Charge
Night Mins
Night Calls
Night Charge
Intl Mins
Intl Calls
Intl Charge
CustServ Calls
Learning method
MethodName=Improved ChAID (Tsc
MethodClassName=TArbreDecision
Hdl=8
Merge=0.05
Split=0.001
TypeBonferroni=1
ValueBonferroni=1
Sampling=0
Examples selection
3333 examples selected
0 examples idle
Improved ChAID (Tschuprow Goodness of Split)
Exec.Time : 454 ms
2850 (86%) False.
483 (14%) True.
Day Mins
< 264.45
2766 (89%)
356 (11%)
>=264.45
84 (40%)
127 (60%)
CustServ Calls
Eve Mins
< 3.50
2642 (92%)
229 (08%)
>=3.50
124 (49%)
127 (51%)
< 187.30
33 (77%)
10 (23%)
>=187.30
51 (30%)
117 (70%)
Day Mins
Day Mins
Day Mins
< 221.85
2272 (95%)
129 (05%)
>=221.85
370 (79%)
100 (21%)
< 160.20
13 (13%)
89 (87%)
>=160.20
111 (74%)
38 (26%)
< 284.80
41 (42%)
57 (58%)
>=284.80
10 (14%)
60 (86%)
3:39 PM
11/9/2022

ASSOCIATION RULES

5

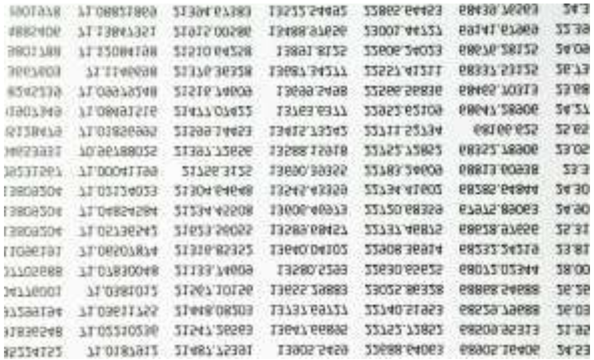
ASSOCIATION RULES

- What it is/how it works
  - Descriptive/predictive technique that—through machine learning—identifies combinations of items, within market baskets, that are most likely to be purchased together, i.e.. {peanut butter, bread} -> jelly
  - Statistics vs. Machine Learning—generally accepted as machine learning
- How to evaluate your results
  - Support, Confidence, Lift, Conviction, Rule Power Factor (RPF)
- How to apply your results
  - Add-factor SKU-level forecasts
- Example
  - Script in R (for today)

ASSOCIATION RULES

- Analyze transactions data for co-occurrence of products (events
- Consider item sets, along with “antecedent” (if) and “consequent” (then)
- Evaluate in terms of *support* and *confidence*

https://searchbusinessanalytics.techtarget.com/definition/association-rules-in-data-mining

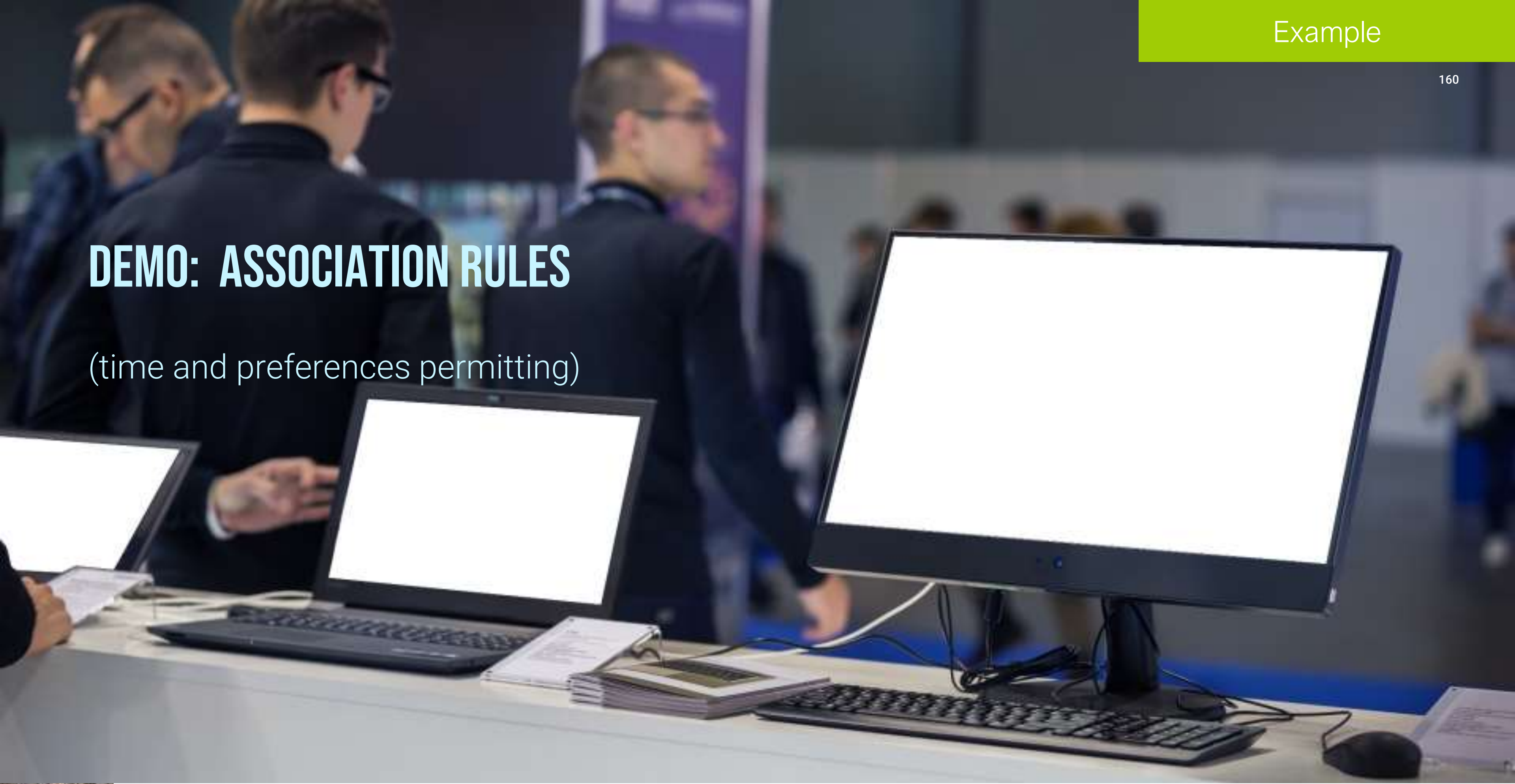


https://www.slideshare.net/aorriols/lecture13-association-rules

ASSOCIATION RULES

KEY MEASURES

Measure	Meaning
Support	• How frequently (% transactions) a set of items occurs in the dataset
Confidence	• How often a rule is true (%) • How often a given set of items (X) is associated with the "outcome" (Y: item or item set) of interest
Lift	• Ratio: how often the rule's prediction is true vs. what you'd find if items were unrelated • >1 means useful for prediction(higher is better) • < 1 means items are substitutes
(lack of) Conviction	• How frequently (%) the rule would predict incorrectly



DEMO: ASSOCIATION RULES

(time and preferences permitting)

```
> summary(Groceries)
transactions as itemMatrix in sparse format with
 9835 rows (elements/itemsets/transactions) and
 169 columns (items) and a density of 0.026

most frequent items:
      whole milk other vegetables       rolls/buns             soda           yogurt          (Other)
            2513             1903             1809             1715             1372            34055

element (itemset/transaction) length distribution:
sizes
   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16   17   18   19   20   21   22   23   24   26   27   28   29   32
2159 1643 1299 1005  855  645  545  438  350  246  182  117   78   77   55   46   29   14   14    9   11    4    6    1    1    1    1    3    1

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      1       2       3       4       6      32

includes extended item information - examples:
       labels  level2           level1
1 frankfurter sausage meat and sausage
2     sausage sausage meat and sausage
3  liver loaf sausage meat and sausage
```

Rows are transactions; items are columns

See distribution of item counts among transactions

Labeling system has more general categories, as well as specifics

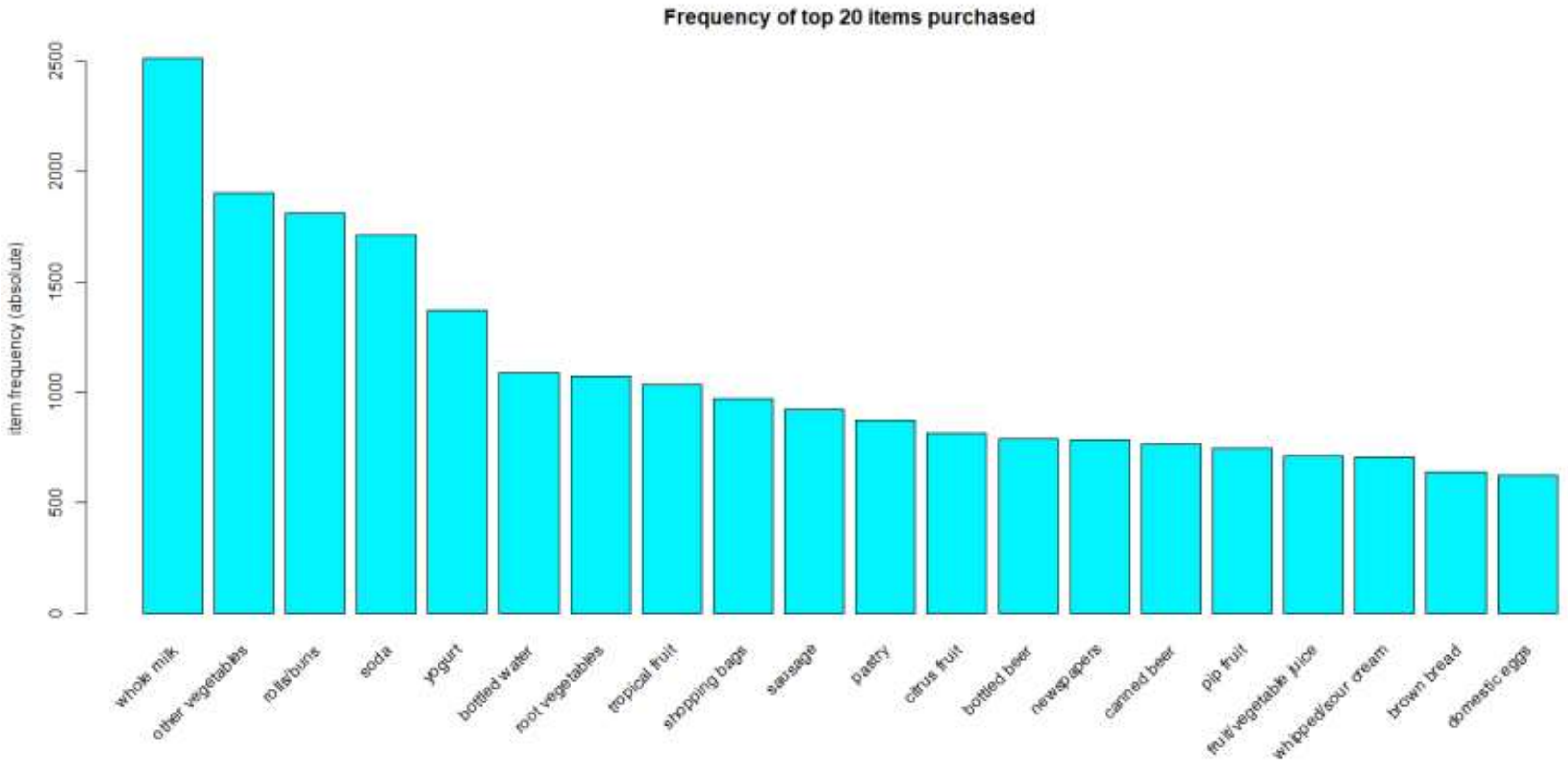
Label details (larger sample)

RGui (64-bit) - [R Console]

File Edit View Misc Packages Windows Help

```
> Groceries@itemInfo
                     labels                    level2               level1
1               frankfurter                   sausage     meat and sausage
2                   sausage                   sausage     meat and sausage
3                liver loaf                   sausage     meat and sausage
4                       ham                   sausage     meat and sausage
5                      meat                   sausage     meat and sausage
6         finished products                   sausage     meat and sausage
7           organic sausage                   sausage     meat and sausage
8                   chicken                   poultry     meat and sausage
9                    turkey                   poultry     meat and sausage
10                     pork                      pork     meat and sausage
11                     beef                      beef     meat and sausage
12           hamburger meat                      beef     meat and sausage
13                     fish                      fish     meat and sausage
14             citrus fruit                     fruit fruit and vegetables
15           tropical fruit                     fruit fruit and vegetables
16                pip fruit                     fruit fruit and vegetables
17                   grapes                     fruit fruit and vegetables
18                  berries                     fruit fruit and vegetables
19              nuts/prunes                     fruit fruit and vegetables
20          root vegetables                vegetables fruit and vegetables
21                   onions                vegetables fruit and vegetables
22                    herbs                vegetables fruit and vegetables
23         other vegetables                vegetables fruit and vegetables
24 packaged fruit/vegetables packaged fruit/vegetables fruit and vegetables
25               whole milk             dairy produce       fresh products
26                   butter             dairy produce       fresh products
27                     curd             dairy produce       fresh products
28                  dessert             dairy produce       fresh products
29              butter milk             dairy produce       fresh products
30                   yogurt             dairy produce       fresh products
31        whipped/sour cream             dairy produce       fresh products
32                beverages             dairy produce       fresh products
33                 UHT-milk        shelf-stable dairy       fresh products
34           condensed milk        shelf-stable dairy       fresh products
35                    cream        shelf-stable dairy       fresh products
36              soft cheese                    cheese       fresh products
37            sliced cheese                    cheese       fresh products
38              hard cheese                    cheese       fresh products
39             cream cheese                    cheese       fresh products
40         processed cheese                    cheese       fresh products
41            spread cheese                    cheese       fresh products
```

Explore the data



Top rules by "Support"
(% of Transactions containing item set)

```
> inspect(rules_supp[1:20])
     lhs                                                                rhs                 support confidence coverage lift count
[1]  {citrus fruit,tropical fruit,root vegetables,whole milk}        => {other vegetables} 0.0032  0.89       0.0036    4.6 31
[2]  {other vegetables,curd,domestic eggs}                            => {whole milk}       0.0028  0.82       0.0035    3.2 28
[3]  {hamburger meat,curd}                                            => {whole milk}       0.0025  0.81       0.0032    3.2 25
[4]  {herbs,rolls/buns}                                               => {whole milk}       0.0024  0.80       0.0031    3.1 24
[5]  {tropical fruit,herbs}                                           => {whole milk}       0.0023  0.82       0.0028    3.2 23
[6]  {citrus fruit,root vegetables,other vegetables,yogurt}           => {whole milk}       0.0023  0.82       0.0028    3.2 23
[7]  {pork,other vegetables,butter}                                   => {whole milk}       0.0022  0.85       0.0026    3.3 22
[8]  {tropical fruit,root vegetables,yogurt,rolls/buns}               => {whole milk}       0.0022  0.81       0.0027    3.2 22
[9]  {tropical fruit,grapes,whole milk}                               => {other vegetables} 0.0020  0.80       0.0025    4.1 20
[10] {root vegetables,other vegetables,yogurt,fruit/vegetable juice} => {whole milk}       0.0020  0.83       0.0024    3.3 20
[11] {root vegetables,whole milk,yogurt,fruit/vegetable juice}       => {other vegetables} 0.0020  0.80       0.0025    4.1 20
[12] {liquor,red/blush wine}                                          => {bottled beer}     0.0019  0.90       0.0021   11.2 19
[13] {yogurt,rice}                                                    => {other vegetables} 0.0019  0.83       0.0023    4.3 19
[14] {herbs,shopping bags}                                            => {other vegetables} 0.0019  0.83       0.0023    4.3 19
[15] {tropical fruit,whipped/sour cream,fruit/vegetable juice}       => {other vegetables} 0.0019  0.90       0.0021    4.7 19
[16] {pork,butter milk}                                               => {other vegetables} 0.0018  0.86       0.0021    4.4 18
[17] {root vegetables,other vegetables,rice}                          => {whole milk}       0.0018  0.82       0.0022    3.2 18
[18] {pip fruit,curd,whipped/sour cream}                              => {whole milk}       0.0018  0.82       0.0022    3.2 18
[19] {pip fruit,butter,whipped/sour cream}                            => {whole milk}       0.0018  0.90       0.0020    3.5 18
[20] {tropical fruit,whipped/sour cream,domestic eggs}                => {whole milk}       0.0018  0.90       0.0020    3.5 18
```

Top rules by "Confidence"
(% of occasions rule is true)

```
> inspect(rules_conf[1:20]) # show the support, lift and confidence for all rules
     lhs                                                    rhs                support confidence coverage lift count
[1]  {rice,sugar}                                        => {whole milk}       0.0012  1          0.0012   3.9  12
[2]  {canned fish,hygiene articles}                      => {whole milk}       0.0011  1          0.0011   3.9  11
[3]  {root vegetables,butter,rice}                       => {whole milk}       0.0010  1          0.0010   3.9  10
[4]  {root vegetables,whipped/sour cream,flour}          => {whole milk}       0.0017  1          0.0017   3.9  17
[5]  {butter,soft cheese,domestic eggs}                  => {whole milk}       0.0010  1          0.0010   3.9  10
[6]  {citrus fruit,root vegetables,soft cheese}          => {other vegetables} 0.0010  1          0.0010   5.2  10
[7]  {pip fruit,butter,hygiene articles}                 => {whole milk}       0.0010  1          0.0010   3.9  10
[8]  {root vegetables,whipped/sour cream,hygiene articles} => {whole milk}     0.0010  1          0.0010   3.9  10
[9]  {pip fruit,root vegetables,hygiene articles}        => {whole milk}       0.0010  1          0.0010   3.9  10
[10] {cream cheese ,domestic eggs,sugar}                 => {whole milk}       0.0011  1          0.0011   3.9  11
[11] {curd,domestic eggs,sugar}                          => {whole milk}       0.0010  1          0.0010   3.9  10
[12] {cream cheese ,domestic eggs,napkins}               => {whole milk}       0.0011  1          0.0011   3.9  11
[13] {pip fruit,whipped/sour cream,brown bread}          => {other vegetables} 0.0011  1          0.0011   5.2  11
[14] {tropical fruit,grapes,whole milk,yogurt}           => {other vegetables} 0.0010  1          0.0010   5.2  10
[15] {ham,tropical fruit,pip fruit,yogurt}               => {other vegetables} 0.0010  1          0.0010   5.2  10
[16] {ham,tropical fruit,pip fruit,whole milk}           => {other vegetables} 0.0011  1          0.0011   5.2  11
[17] {tropical fruit,root vegetables,yogurt,oil}         => {whole milk}       0.0011  1          0.0011   3.9  11
[18] {root vegetables,other vegetables,yogurt,oil}       => {whole milk}       0.0014  1          0.0014   3.9  14
[19] {root vegetables,other vegetables,butter,white bread} => {whole milk}     0.0010  1          0.0010   3.9  10
[20] {pork,other vegetables,butter,whipped/sour cream}   => {whole milk}       0.0010  1          0.0010   3.9  10
```

Top rules by “Lift”
(How often a predicted result is true vs. random occurrence)

```
> inspect(rules_lift[1:20]) # show the support, lift and confidence for all rules
     lhs                                                       rhs                support confidence coverage lift count
[1]  {liquor,red/blush wine}                                => {bottled beer}     0.0019  0.90       0.0021   11.2 19
[2]  {citrus fruit,other vegetables,soda,fruit/vegetable juice} => {root vegetables} 0.0010  0.91    0.0011    8.3 10
[3]  {tropical fruit,other vegetables,whole milk,yogurt,oil}  => {root vegetables} 0.0010  0.91       0.0011    8.3 10
[4]  {citrus fruit,grapes,fruit/vegetable juice}            => {tropical fruit}   0.0011  0.85       0.0013    8.1 11
[5]  {other vegetables,whole milk,yogurt,rice}              => {root vegetables}  0.0013  0.87       0.0015    8.0 13
[6]  {tropical fruit,other vegetables,whole milk,oil}       => {root vegetables}  0.0013  0.87       0.0015    8.0 13
[7]  {ham,pip fruit,other vegetables,yogurt}                => {tropical fruit}   0.0010  0.83       0.0012    7.9 10
[8]  {beef,citrus fruit,tropical fruit,other vegetables}    => {root vegetables}  0.0010  0.83       0.0012    7.6 10
[9]  {root vegetables,butter,cream cheese }                 => {yogurt}           0.0010  0.91       0.0011    6.5 10
[10] {tropical fruit,whole milk,butter,sliced cheese}       => {yogurt}           0.0010  0.91       0.0011    6.5 10
[11] {other vegetables,curd,whipped/sour cream,cream cheese } => {yogurt}         0.0010  0.91       0.0011    6.5 10
[12] {tropical fruit,other vegetables,butter,white bread}   => {yogurt}           0.0010  0.91       0.0011    6.5 10
[13] {sausage,pip fruit,sliced cheese}                      => {yogurt}           0.0012  0.86       0.0014    6.1 12
[14] {tropical fruit,whole milk,butter,curd}                => {yogurt}           0.0012  0.86       0.0014    6.1 12
[15] {tropical fruit,butter,white bread}                    => {yogurt}           0.0011  0.85       0.0013    6.1 11
[16] {tropical fruit,butter,margarine}                      => {yogurt}           0.0011  0.85       0.0013    6.1 11
[17] {whole milk,curd,whipped/sour cream,cream cheese }     => {yogurt}           0.0011  0.85       0.0013    6.1 11
[18] {whipped/sour cream,cream cheese ,margarine}           => {yogurt}           0.0010  0.83       0.0012    6.0 10
[19] {beef,tropical fruit,butter}                           => {yogurt}           0.0010  0.83       0.0012    6.0 10
[20] {pork,tropical fruit,fruit/vegetable juice}            => {yogurt}           0.0010  0.83       0.0012    6.0 10
```

If lift=1, lhs and rhs are independent
If lift < 0, lhs and rhs are negatively correlated
If lift > 0, lhs and rhs are positivey correlated

PUTTING IT ALL TOGETHER...

Summary

SUMMARY

PREDICTIVE ANALYTICS IN FORECASTING

- Data extraction and aggregation are key to predictive analytics and forecasting
  - Transaction data aggregate to customer-level measures
  - Customer-level measures aggregate to class-level measures

- Descriptive techniques can support predictor selection and class formation (customers, products)

- Predictive techniques can be applied at the customer level, class level, product level and higher to enrich relevance and accuracy of forecasts

- Illustrations of both Descriptive and Predictive techniques can increase the value of forecasting presentations

- Statistics and Machine Learning often overlap, offering opportunities for forecast precision and skill development