Module 3: Vantage Functions across R & Python

Day on the life of a Data Scientist Workshop

Copyright © 2007–2022 by Teradata. All Rights Reserved.

Objectives

After completing this module, you will be able to:

- Write queries in SQL, Python, and R using the main analytic functions

- Visualize query results in Vantage, Python and R

Topics

- Introduction
- Red Bull Scenario Association Analysis
  - Data Science Process
    - SQL
    - Python
    - R

Introduction

- We will use various analytic functions in **SQL**, **Python**, and **R** in one case study with real information
- For the Red Bull dataset, we will be using the Association function to determine which products co-occur within the same transactions as when Red Bull is present.
- We also will replicate the process in R and Python, to show the differences and similarities.

Current Topic –
Red Bull Scenario using Data Science Process

- Introduction
- Red Bull Scenario Association Analysis
  - **Data Science Process**
    - SQL
    - Python
    - R

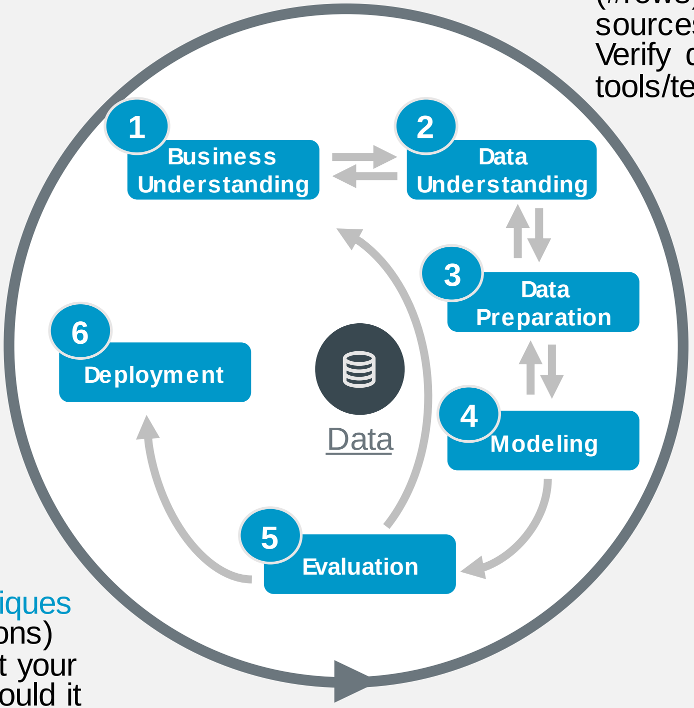
teradata.

Data Science Process (CRISP DM) – Six Steps

1. You must be clear on the business goal of the analytic request, the success criteria and Data Science goal(s)

6. Operationalize and revisit over time. Should it be operationalized? Once you have followed your companies process for operationalizing, you need to consider these monitoring and maintenance questions: Has data changed? Has new data become available? Is the model still predicting accurately? Etc.

5. Compare models / analysis techniques (e.g. using Vantage statistical functions) and select best results. Does it meet your business goals/success criteria? Should it be visualized and/or operationalized?



CRISP-DM

2. You must know the underlying data. Size (#rows)? What's in the data (Columns)? Data sources? Schema type - structured or unstructured? Verify data quality. Use Vantage functions and other tools/techniques to examine the data

3. Does the data need preparing? Yes/No? Remove outliers? Missing data? Scale? Transform? Organize by geolocation? Perform data preparation using Vantage data preparation functions

4. Which predictive/analytic functions? Which arguments? Based on function, does data need further cleaning/preparing? Execute Vantage ML functions to create models/ analyses

– Build supervised/predictive model(s) on training set, validate on test set, and use Vantage statistical functions to assess accuracy of results
– Perform unsupervised/descriptive analytics on full data set and assess reasonableness of results visually
– Make sure to experiment. Use visualization to assist in model/analysis assessment. Keep track of peripheral findings

Data Science Process for the Red Bull Scenario

Step	Description	Activities	Comments
1	**Business Understanding**	Business goal? Success criteria? DS goal?	We must build a Model that accurately predicts the top 10 products with the highest affinity to Red Bull (purchased in same transaction)
2	**Data Understanding**	What does the data show? Where is it located? Is it complete/correct?	Data resides in **sales_detail1**, which contains detailed transaction information in Vantage, showing *who* bought *what*, *when*, *where*, *how much*, etc. The data is complete and clean
3	**Data Preparation**	Outliers? Scale? Organize by geolocation?	Not needed
4	**Modeling/Analysis**	Which functions? Which arguments? Experiment! Assess and compare models/analyses	**Collaborative Filtering** SQL: CFilter Python: CFilter R: td_cfilter_mle
5	**Evaluation**	Are your business goals and criteria being met? Visualize? Pick best performers	Analyze the results and determine the strongest associated products with Red Bull
6	**Deployment**	Operationalize and revisit over time.	Plan to operationalize the analytic results varies by customer and is not covered in this course. Revisit process over time.

Step 1. Business Understanding
Here's the Red Bull Scenario

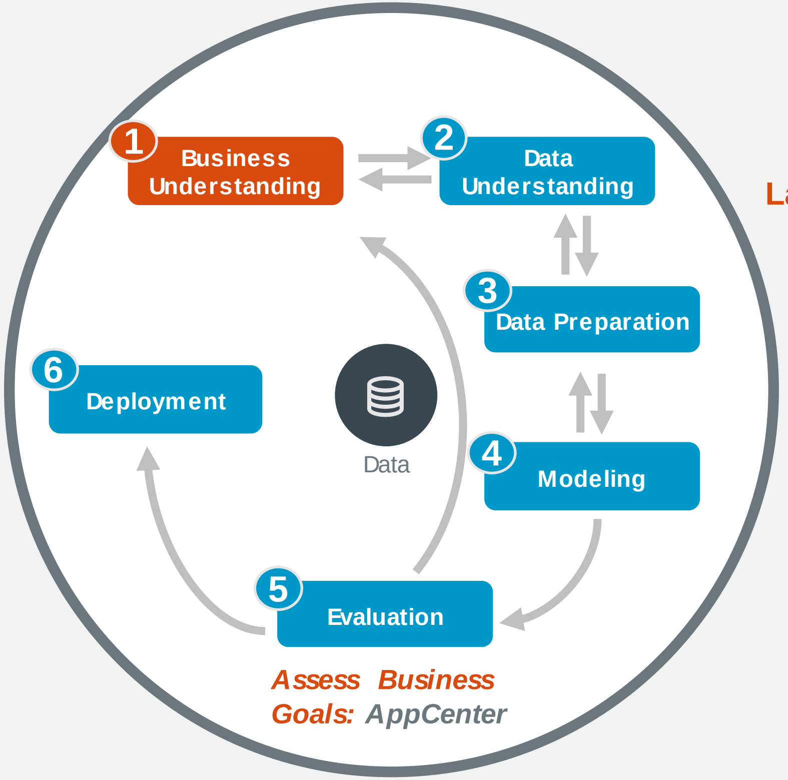
Goal – Red Bull® has given promotional dollars to fund a grocery advertisement. To maximize sales, use Collaborative Filtering to find which products have strongest affinity with Red Bull. Promote the top 10 affinity products along with Red Bull

Current Topic – Red Bull Scenario with SQL

- Introduction
- Red Bull Scenario Association Analysis
  - Data Science Process
    - **SQL**
    - Python
    - R

Step 1. Business Understanding
Map Out Data Science Process – Tools and Functions

Goal: Which products have the strongest affinity with Red Bull, and advertise these products with Red Bull



CRISP-DM

Language: SQL in Teradata Studio

Functions: N/A – No data prep needed

Association Function : VAL Association

Model Accuracy: Association statistics

Step 2. Data Understanding
Lab 01: View the Data (1 of 2)

Answer these questions:

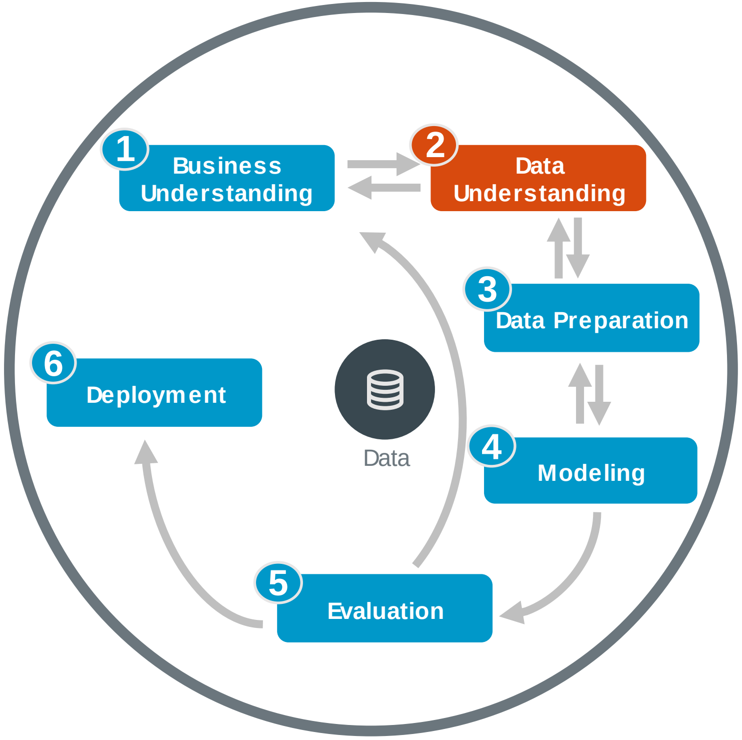
a) Multiple data sources? No

b) Vendor data source? Teradata only

c) Object type? SQL table

d) Is it accessible? Yes, Read permissions

e) Has Schema? Yes (structured data)



CRISP-DM

Step 2. Data Understanding
Lab 01: View the Data (2 of 2)

```
SELECT * FROM sales_detail1 SAMPLE 100;
```

	product_name	product_category...	store_name	region_name	city_name	sales_date	customer_id	basket_id	store_id
64	Peaches	Fruits	Seattle	Western	Seattle	2008-07-12 00:0...	28	65288	8
65	Rice Krispie treats	Other Snacks	Seattle	Western	Seattle	2008-03-26 00:0...	28	173288	8
66	Cola	Drinks	New York	Eastern	New York	2007-11-08 00:0...	30	3123010	10
67	Fairy bread	Ethnic Snacks	New York	Eastern	New York	2008-03-26 00:0...	30	1733010	10
68	Jelly Beans	Candy	New York	Eastern	New York	2008-05-01 00:0...	30	1373010	10
69	Bugles	Chips	New York	Eastern	New York	2008-03-26 00:0...	30	1733010	10
70	Mamee	Ethnic Snacks	New York	Eastern	New York	2008-08-17 00:0...	30	293010	10
71	Red Bull	Drinks	New York	Eastern	New York	2008-08-29 00:0...	30	173010	10
72	Cola	Drinks	New York	Eastern	New York	2008-05-01 00:0...	30	1373010	10
73	Candy Bars	Candy	New York	Eastern	New York	2008-03-14 00:0...	30	1853010	10
74	Sprite	Drinks	New York	Eastern	New York	2008-08-05 00:0...	30	[illegible]	10

Business Objective: Which products have the strongest affinity with Red Bull, and advertise these products with Red Bull

f) Describe and explore data — 100 rows displaying transaction data
product_name displays the product that was purchased
basket_id is a unique identifier for the transaction

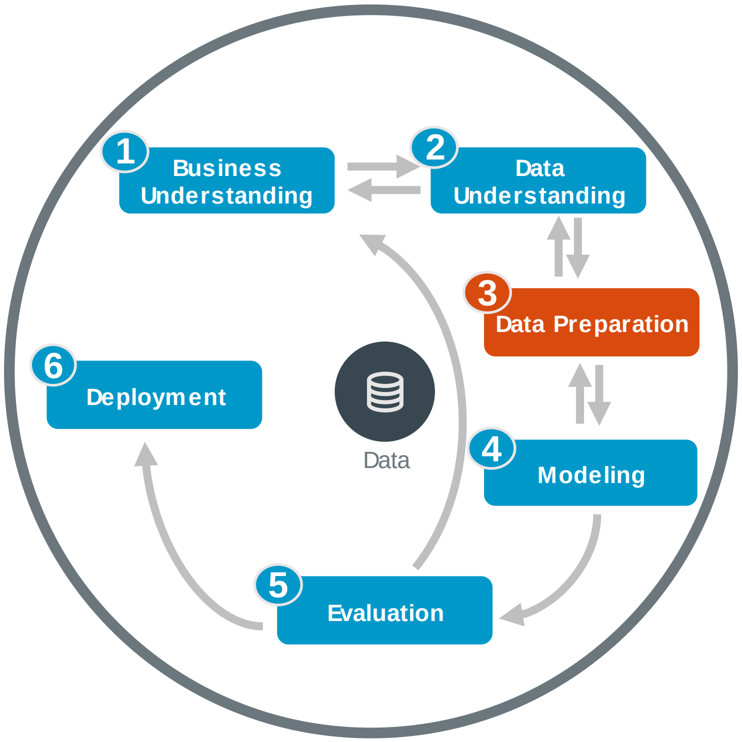
g) Verify data adequacy — Yes, data is adequate for analysis task

h) Verify data quality — Complete? – Yes, covers all cases required
Correct? – missing values or errors

Step 3. Data Preparation

a) Does data require Cleaning? Does data need to be Scaled? Do Outliers need to be removed?

No, the data has already been prepared



CRISP-DM

Step 4. Modeling

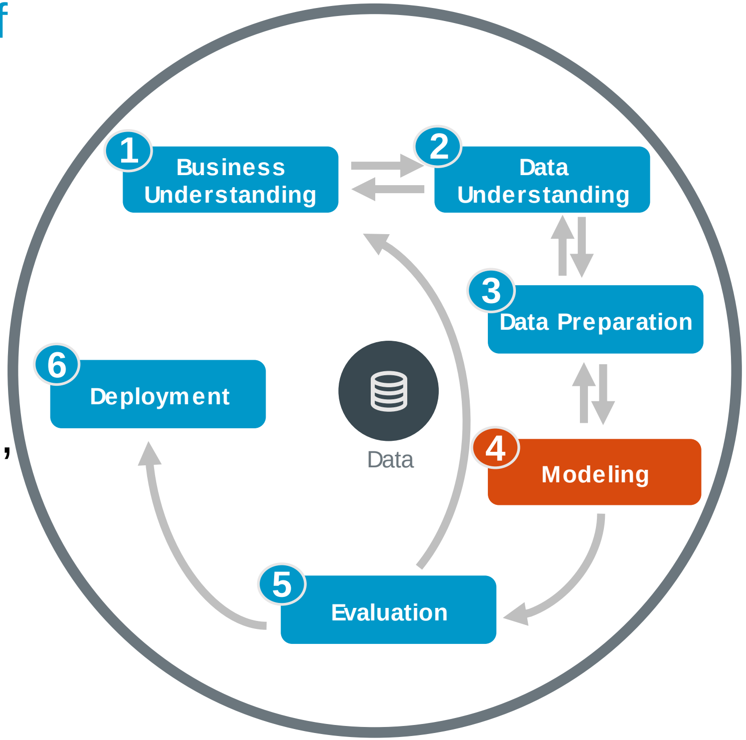
a) Which Function (algorithms) and which order of execution?

Market Basket (**Association**) only

Association: A function that allows you to understand the extent to which disparate product domains do or do not co-occur together within the same grouping; e.g., same transaction, same household, same timeframe, same geography, etc.

b) Does data require more data preparation for **Association** function?

No



CRISP-DM

Step 4. Modeling
Lab 02: Association Syntax


```
SELECT * FROM CFilter (
ON sales_detail1 AS InputTable
OUT Table OutputTable (cf_redbull_output)
USING
TargetColumns ('product_name')
JoinColumns ('basket_id') ) AS dt;

SELECT * cf_redbull_output;
```

	col1_item1	col1_item2	cntb	cnt1	cnt2	score	support	confidence	lift	z_score
1	Red Bull	Cup noodles	2	22	9	0.020202020	0.002958580	0.090909091	6.828282828	4.165072159
2	Red Bull	Licorice	1	22	5	0.009090909	0.001479290	0.045454545	6.145454545	-0.240091879
3	Red Bull	Pistachio nuts	1	22	5	0.009090909	0.001479290	0.045454545	6.145454545	-0.240091879
4	Red Bull	Cola	2	22	12	0.015151515	0.002958580	0.090909091	5.121212121	4.165072159
5	Red Bull	Toaster pastries	2	22	12	0.015151515	0.002958580	0.090909091	5.121212121	4.165072159
6	Red Bull	Cheese puffs	1	22	7	0.006493506	0.001479290	0.045454545	4.389610390	-0.240091879
7	Red Bull	Geplak	1	22	7	0.006493506	0.001479290	0.045454545	4.389610390	-0.240091879
8	Red Bull	Confections	1	22	8	0.005681818	0.001479290	0.045454545	3.840909091	-0.240091879
9	Red Bull	Meze	1	22	8	0.005681818	0.001479290	0.045454545	3.840909091	-0.240091879
10	Red Bull	Nachos	1	22	8	0.005681818	0.001479290	0.045454545	3.840909091	-0.240091879

- **OutputTable**: Specify the name of the output table that the function creates. Here, we are writing the results to **cf_redbull_output**
- **TargetColumns**: Specify the names of the input table columns for which the function seeks out co-occurrences; e.g., prod_id, upc_num, cat_id, etc. Here, we are seeking out product co-occurrences for **product_name** values
- **JoinColumns**: Specify the names of join columns. This determines the "level" at which co-occurrences are sought out; e.g., trans_id, hh_id, etc. In our case, we seek out co-occurrences at the **basket_id** level

Step 4. Modeling / Step 5. Evaluation

Lab 03: Output – Is It Actionable? Yes

Based on metrics, pick product pairings having highest affinity with Red Bull. In this case, we have opted to base this on **Lift** (the higher, the better)

```
SELECT * FROM cf_redbull_output
WHERE col1_item1 = 'Red Bull'
AND lift >= 3.84
ORDER BY lift DESC;
```

In this small sample, Red Bull and Cup noodles has highest '**lift**'

Metrics

	col1_item1	col1_item2	cntb	cnt1	cnt2	score	support	confidence	lift	z_score
1	Red Bull	Cup noodles	2	22	9	0.020202020...	0.002958579...	0.090909090...	6.828282828...	4.165072158...
2	Red Bull	Licorice	1	22	5	0.009090909...	0.001479289...	0.045454545...	6.145454545...	-0.240091878...
3	Red Bull	Pistachio nuts	1	22	5	0.009090909...	0.001479289...	0.045454545...	6.145454545...	-0.240091878...
4	Red Bull	Cola	2	22	12	0.015151515...	0.002958579...	0.090909090...	5.121212121...	4.165072158...
5	Red Bull	Toaster pastries	2	22	12	0.015151515...	0.002958579...	0.090909090...	5.121212121...	4.165072158...
6	Red Bull	Cheese puffs	1	22	7	0.006493506...	0.001479289...	0.045454545...	4.389610389...	-0.240091878...
7	Red Bull	Geplak	1	22	7	0.006493506...	0.001479289...	0.045454545...	4.389610389...	-0.240091878...
8	Red Bull	Confections	1	22	8	0.005681818...	0.001479289...	0.045454545...	3.840909090...	-0.240091878...
9	Red Bull	Meze	1	22	8	0.005681818...	0.001479289...	0.045454545...	3.840909090...	-0.240091878...
10	Red Bull	Nachos	1	22	8	0.005681818...	0.001479289...	0.045454545...	3.840909090...	-0.240091878...

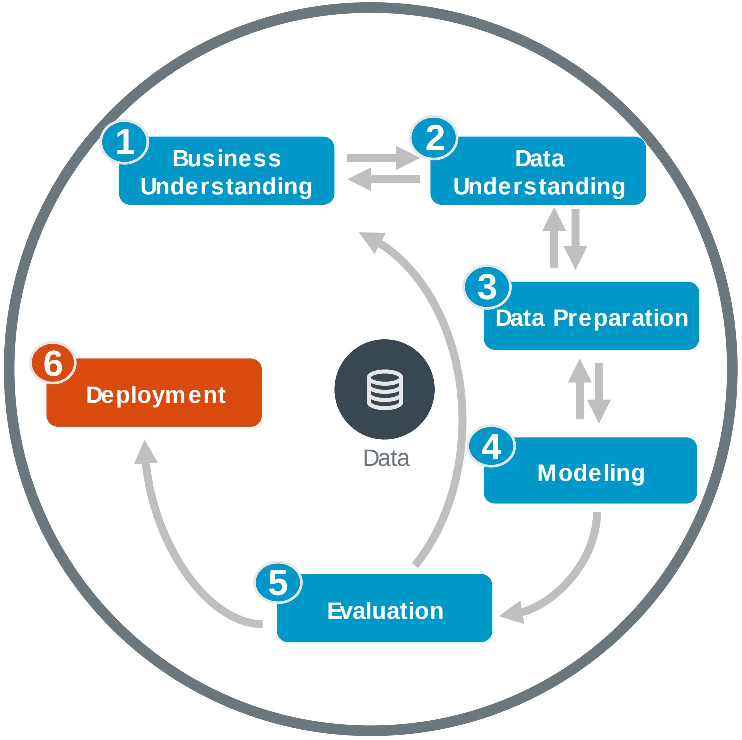
Step 6a. Deployment (Operationalizing/Monitoring/Maintenance)

teradata.

6. **Deployment** – The end goal is to "operationalize" the analytic findings. Taking analytics from insight to impact – the process of getting analytics out to business stakeholders for use/reuse to meet business goals

   a) **Plan deployment (how to operationalize)**

   Note: This varies by customer and is not covered in this course



CRISP-DM

Step 6b. Deployment (Operationalizing/Monitoring/Maintenance)

b) **Plan monitoring and maintenance** – Once it's operationalized, it's important to monitor and maintain it. Does your process need to be revisited as time marches on? Consider the following:

1. Do you still carry Red Bull?
2. Do you still carry the affinity products to Red Bull?
3. Are there any new products that you started carrying since you last ran your analysis? Might any of these products have an affinity to Red Bull?
4. Have the purchasing habits of your customers changed?

Note: This step is the same regardless of language; i.e., Python, R, or SQL

Current Topic – Red Bull Scenario with Python

- Introduction
- Red Bull Scenario Association Analysis
  - Data Science Process
    - SQL
    - **Python**
    - R

Step 1. Business Understanding
Map Out Data Science Process – Tools and Functions

Goal: Which products have the strongest affinity with Red Bull, and advertise these products with Red Bull

CRISP-DM

Language: Python in JupyterLab

Functions: N/A – no data prep needed

Association Function : CFilter

Visualization apps: NA

Model Accuracy: CFilter statistics

Lab 00: Import Python Libraries

1. Highlight Cell 1 (you'll get a blue vertical bar for that Cell)
2. Click Run button . Kernel indicator circle will fill in. When finished it will be White again (sometimes it happens so fast won't see circle fill in)
3. Run Cell 2 to Display all Python code as SQL code

Lab 00: Import the teradataml and pandas packages

[1]:

```
import teradataml as tdml
from teradataml import *
import pandas
```

Lab 00: Connect JupyterLab to Vantage

Run 'create_context' method to connect Python client to Vantage Cluster via JDBC

Lab 00: Connect to Vantage

```
1  # WARNING: Edit Line 5.  'username' = Your QuickLook ID
2  # WARNING: Once run, type YOUR password in 'Caption' box to proceed
3
4  create_context(host='tdprd2.td.teradata.com',
5  use                r=getpass.getpass(), logmech='LDAP')
```

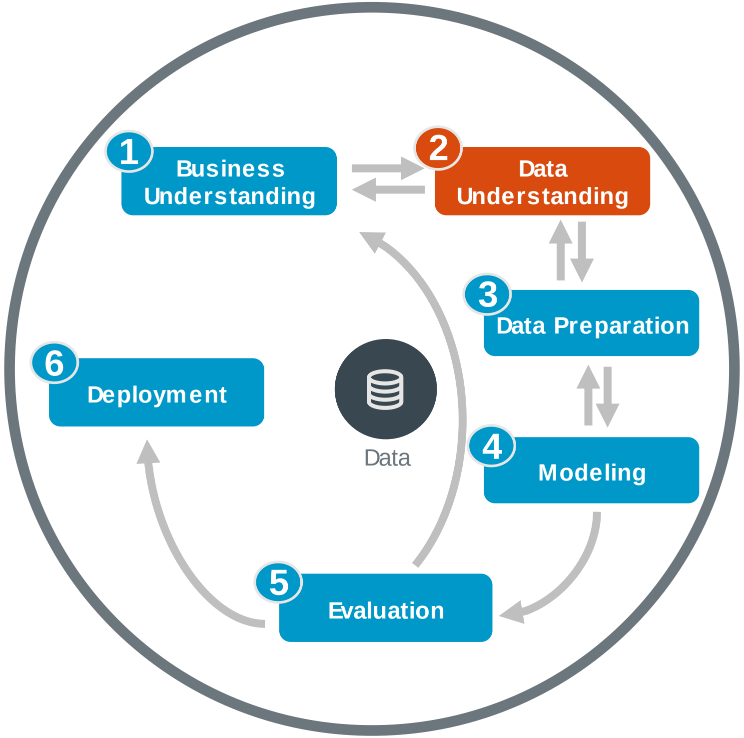
td01 / td01

Step 2. Data Understanding
Answer following questions

teradata.

Answer these questions

a) Multiple data sources?:	No
b) Vendor data source?	Teradata only
c) Object type?	SQL table
d) Is it accessible?	Yes
e) Has Schema?	Yes



CRISP-DM

Step 2. Data Understanding
Lab 01: Load/View Sales Detail Dataframe


```
## Load salesdetail1 table into a TD DataFrame
red_bull_df = tdml.DataFrame('sales_detail1')
print(red_bull_df)
```

Business Objective: Which products have the strongest affinity with Red Bull, and advertise these products with Red Bull

[9]:

customer_id	product_name	product_category_name	store_name	region_name	city_name	sales_date	basket_id	store_id	sales_quantity	discount_amount
40	Cashews	Nuts	New York	Eastern	New York	2008-08-05 00:00:00.000000	414010	10	9	.160
40	Licorice	Candy	New York	Eastern	New York	2008-03-26 00:00:00.000000	1734010	10	1	.050
40	Root Beer	Drinks	New York	Eastern	New York	2008-03-14 00:00:00.000000	1854010	10	5	.010
40	Sunflower seeds	Nuts	New York	Eastern	New York	2008-08-17 00:00:00.000000	294010	10	8	.080
40	Lollipops	Candy	New York	Eastern	New York	2008-03-14 00:00:00.000000	1854010	10	2	.190
40	Gummy Bears	Candy	New York	Eastern	New York	2008-07-24 00:00:00.000000	534010	10	1	.050
40	French Fries	Other Snacks	New York	Eastern	New York	2007-12-14 00:00:00.000000	2764010	10	1	.170
40	Bonda	Ethnic Snacks	New York	Eastern	New York	2008-08-17 00:00:00.000000	294010	10	1	.050
40	Pork Rind	Chips	New York	Eastern	New York	2008-08-17 00:00:00.000000	294010	10	4	.180
40	Ice cream	Other Snacks	New York	Eastern	New York	2008-04-07 00:00:00.000000	1614010	10	2	.080

f) Describe and explore data — View rows displaying transaction data
- product_name displays the product that was purchased
- basket_id is a unique identifier for the transaction

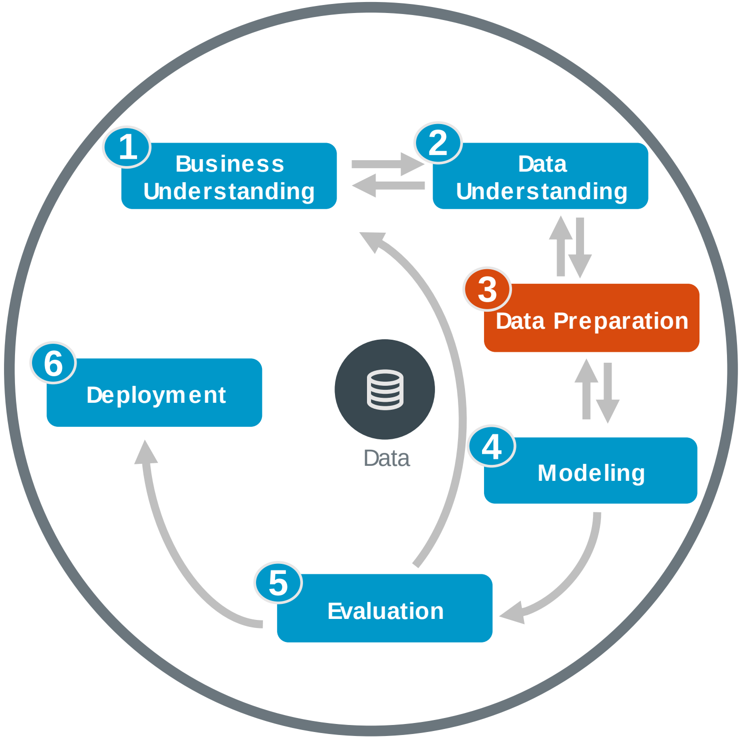
g) Verify data adequacy — Yes, data is adequate for analysis task

h) Verify data quality — Complete? – Yes, covers all cases required
Correct? – Yes, no missing values or errors

Step 3. Data Preparation

a) Does data require Cleaning? Does data need to be Scaled? Do Outliers need to be removed?

No, the data has already been prepared



CRISP-DM

Step 4. Modeling
Lab 02: CFilter Function


```
## Run CFilter function
red_bull_cf = CFilter (data = red_bull_df,
                       input_columns = ["product_name"],
                       join_columns = ["basket_id"])
```

- Here, we are running the **CFilter** function to discover which products co-occur together within the same transaction
- We are creating a DataFrame of the results, and then displaying the output
- **input_columns**: Specify the names of the input table columns that contain the data to filter. Since we have specified "product_name", the output will display "product_name" values that co-occur together
- **join_columns**: Specify the names of join columns. This will determine the level at which co-occurrences are sought out. In this case, we are looking for co-occurrences at the "basket_id" level

Step 5. Evaluation
Lab 03: CFilter Output – Is It Actionable? Yes

teradata.

- Based on metrics, pick product pairings having highest affinity with Red Bull
- In this case, we are merely eyeballing the output as a start
- Review the **Notes** page for column definitions

```
# Review output
print(red_bull_cf)
```

Metrics

	col1_item1	col1_item2	cntb	cnt1	cnt2	score	support	confidence	lift	z_score
0	Sun Chips	Cola	1	16	12	0.005208	0.001479	0.062500	3.520833	-0.240092
1	Pita chips	Tapas	1	13	7	0.010989	0.001479	0.076923	7.428571	-0.240092
2	Smores	Fairy bread	1	8	17	0.007353	0.001479	0.125000	4.970588	-0.240092
3	Cola	Sun Chips	1	12	16	0.005208	0.001479	0.083333	3.520833	-0.240092
4	Coconut	Mike and Ikes	1	6	15	0.011111	0.001479	0.166667	7.511111	-0.240092
5	Peaches	Rice Krispie treats	2	9	9	0.049383	0.002959	0.222222	16.691358	4.165072
6	Corn chips	Candy Bars	1	8	15	0.008333	0.001479	0.125000	5.633333	-0.240092
7	Pretzels	Dolma	1	14	13	0.005495	0.001479	0.071429	3.714286	-0.240092
8	Cola	Jelly Beans	1	12	16	0.005208	0.001479	0.083333	3.520833	-0.240092
9	Cola	Pretzels	1	12	14	0.005952	0.001479	0.083333	4.023810	-0.240092

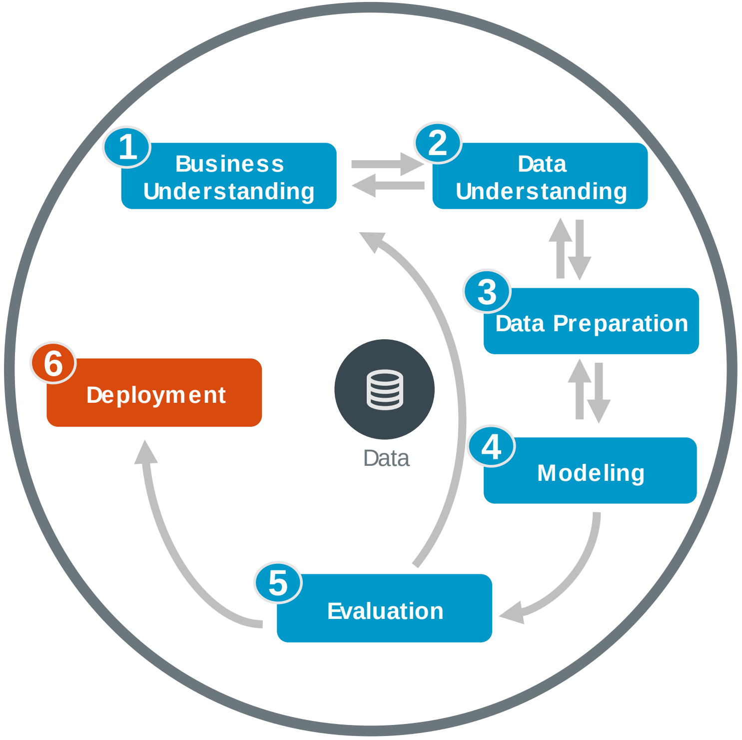
Step 6a. Deployment (Operationalizing/Monitoring/Maintenance)

teradata.

6. **Deployment** – The end goal is to "operationalize" the analytic findings. Taking analytics from insight to impact – the process of getting analytics out to business stakeholders for use/reuse to meet business goals

   a) **Plan deployment (how to operationalize)**

   Note: This varies by customer and is not covered in this course



CRISP-DM

Step 6b. Deployment (Operationalizing/Monitoring/Maintenance)

b) **Plan monitoring and maintenance** – Once it's operationalized, it's important to monitor and maintain it. Does your process need to be revisited as time marches on? Consider the following:

1. Do you still carry Red Bull?
2. Do you still carry the affinity products to Red Bull?
3. Are there any new products that you started carrying since you last ran your analysis? Might any of these products have an affinity to Red Bull?
4. Have the purchasing habits of your customers changed?

Note: This step is the same regardless of language; i.e., Python, R, or SQL

Current Topic – Red Bull Scenario with R

- Introduction
- Red Bull Scenario Association Analysis
  - Data Science Process
    - SQL
    - Python
    - **R**

Step 1. Business Understanding
Map Out Data Science Process – Tools and Functions

Goal: Which products have the strongest affinity with Red Bull, and advertise these products with Red Bull

CRISP-DM

Language: R in RStudio

Functions: N/A – no data prep needed

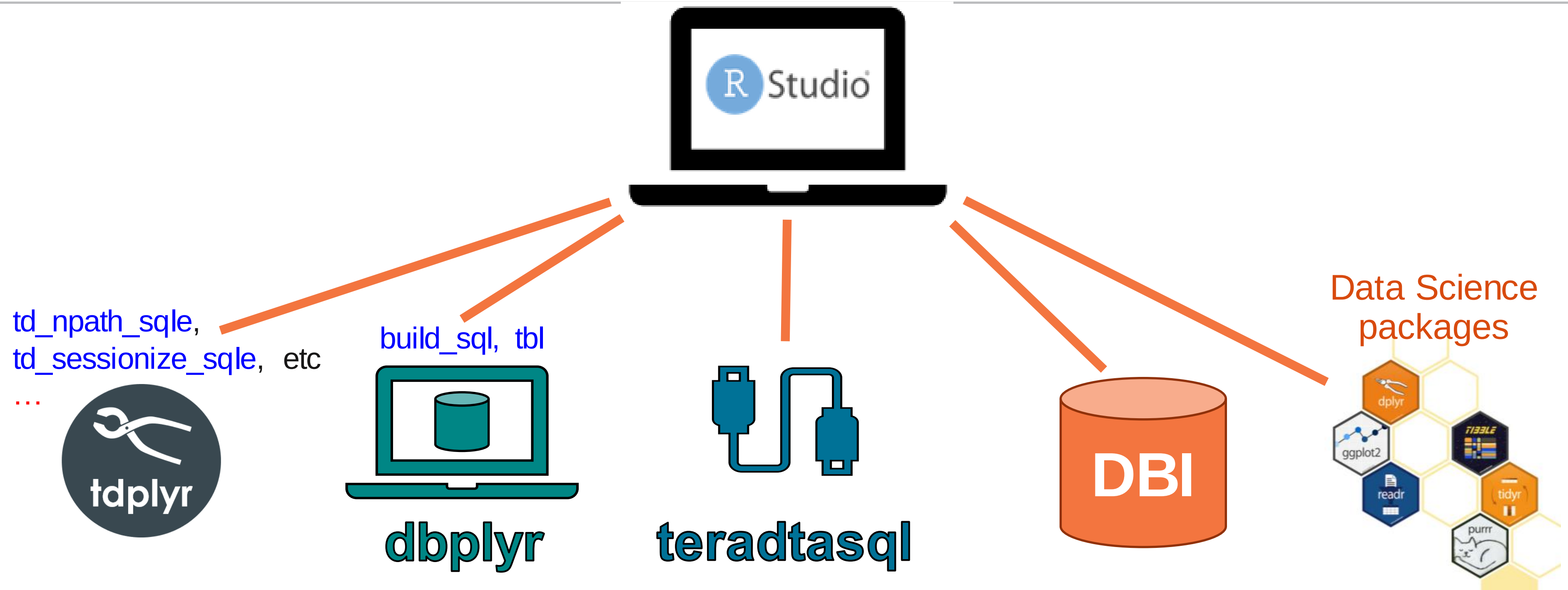
Association Function : td_cfilter_mle

Visualization apps: NA

Model Accuracy: CFilter statistics

teradata.

Dependent Packages You Will Need for Teradata Vantage



Teradata Vantage functions

Work with remote database tables

Enables R app to connect to Teradata Database

Communication between R and RDBMS

Tidyverse functions

Lab 00: Load Libraries

Load Dependent R Libraries followed by 'tdplyr'

```
# Load Libraries
LoadPackages <- function() {
library(getPass)
library(dbplyr)
library(DBI)
library(tidyverse)
library(teradatasql)
library(tdplyr)
}

# Suppress Package Detailed Information
"suppressPackageStartupMessages(LoadPackages())
```

See next pages for details on these two Libraries

Using 'dplyr' with Vantage

The Grammar of Data Manipulation

- One of the packages within the **tidyverse**
  - What is the sum of the values, grouped by product ID?
  - What are the most common car mechanical problems?
  - Which are the products with more than 10,000 reviews?
  - How do I see my data in descending order?
- Like base **SQL** and **Pandas** in **Python**

List of Helpful 'dplyr' Verbs

top_n()

Select the top *n* number of rows

mutate()

Adds new variable that are functions of existing variable

filter()

Picks cases based on their values

dplyr

select()

Picks variables based on their names

summarize()

Reduces multiple values down to a single summary

arrange()

Changes the ordering of the rows

'tdplyr' Package Compared to 'dplyr' Package

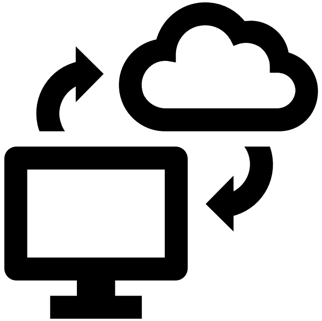
tdplyr

- td_cfilter_mle()
- td_glm_mle()
- td_ngramsplitter()

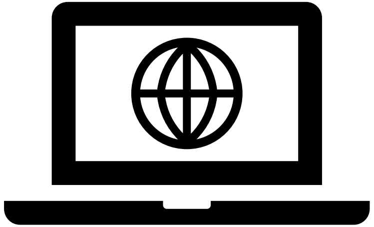
dplyr

- arrange()
- filter()
- top_n()
- summarize()
- select()
- mutate()

Create and Set Teradata Vantage Context



tdplyr



td_create_context

Create a Context to perform analytic functions on Teradata Vantage

td_set_context

Initialize a Context to perform analytic functions on Teradata Vantage

Lab 00: Create and Set Teradata Vantage Context

```
# Create Vantage Context
con <- td_create_context (
            host = "host_name",
            uid = "user_id",
            pwd = getpass(),
            dType = "native",
            logmech = "LDAP")

# Connect to Vantage
td_set_context(con)
```

Your code may vary slightly from this Generic example

Create a variable name **con**

1. Use the **td_create_context** function
2. Input the appropriate information for the remaining arguments.

3. Input the **con** variable as the parameter using the **td_set_context** function

Step 2. Data Understanding
Lab 01: Load/View Sales Detail Tibble

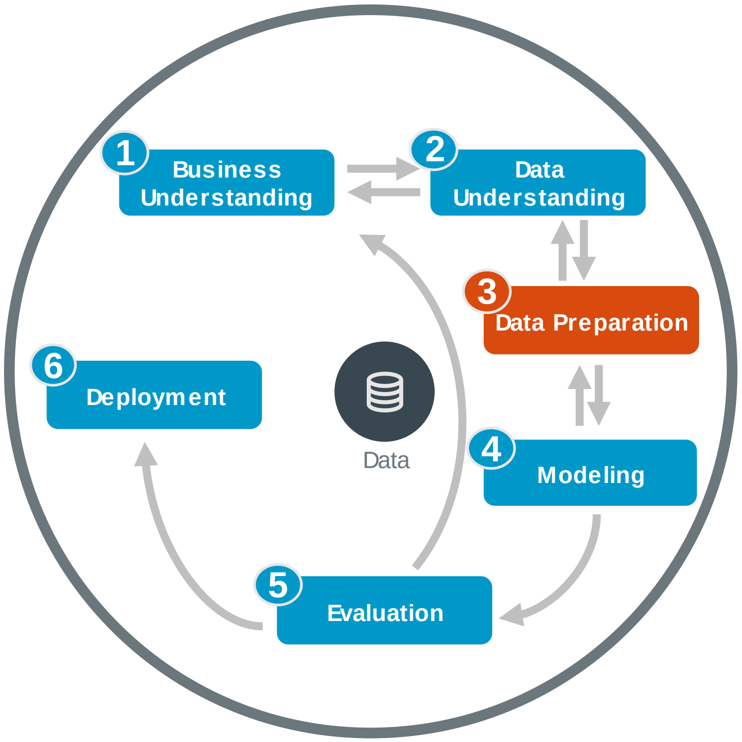

```
# Create Tibble and Display
sales_detail <- tbl(con, dplyr::sql("SELECT * FROM td01.sales_detail1"))
print(sales_detail)

# Count records in Tibble
count(sales_detail)
```

product_name <chr>	product_category_name <chr>	store_name <chr>	region_name <chr>	city_name <chr>	sales_date <dttm>	customer_id <int>	basket_id <int>	store_id <int>
Pixi Stix	Candy	Seattle	Western	Seattle	2007-11-08 00:00:00	38	312388	8
Cup noodles	Ethnic Snacks	San Diego	Western	San Diego	2007-11-08 00:00:00	244	3122444	4
Marshmallows	Candy	Atlanta	Eastern	Atlanta	2008-03-26 00:00:00	156	1731566	6
Snack Mix	Chips	Atlanta	Eastern	Atlanta	2008-04-07 00:00:00	156	1611566	6
Tortilla chips	Chips	New York	Eastern	New York	2008-03-14 00:00:00	120	18512010	10
Red Bull	Drinks	New York	Eastern	New York	2008-03-14 00:00:00	200	18520010	10
Bagel chips	Chips	New York	Eastern	New York	2008-08-29 00:00:00	200	1720010	10
Pita chips	Chips	New York	Eastern	New York	2007-11-08 00:00:00	200	31220010	10
Bluberries	Fruits	Denver	Western	Denver	2008-08-05 00:00:00	122	411222	2
Bajji	Ethnic Snacks	Los Angeles	Western	Los Angeles	2008-07-12 00:00:00	263	652633	3

Step 3. Data Preparation

a) Does data require Cleaning? Does data need to be Scaled? Do Outliers need to be removed?

No, the data has already been prepared



CRISP-DM

Step 4. Modeling
Lab 02: Create 'td_cfilter_mle' Object


```
# Run the td_cfilter_mle function
 cf_redbull_output <- td_cfilter_mle (
        data = sales_detail,
        input.columns = ("product_name"),
        join.columns = ("basket_id"))
```

- Here, we are running the **td_cfilter_mle** function to discover which products co-occur together within the same transaction
- We are creating a DataFrame of the results, and then displaying the output
- **input.columns**: Specify the names of the input table columns that contain the data to filter. Since we have specified "product_name", the output will display "product_name" values that co-occur together
- **join.columns**: Specify the names of join columns. This will determine the level at which co-occurrences are sought out. In this case, we are looking for co-occurrences at the "basket_id" level

Step 4. Modeling and 5. Evaluation
Lab 03: Review CFilter Output

- Based on metrics, pick product pairings having highest affinity with Red Bull. In this case, we have opted to base this on **Lift** (the higher, the better)
- Review the **Notes** page for column definitions

```
# Review output
  cf_redbull_output$output.table  %>%
  filter(col1_item1 == 'Red Bull')  %>%
  arrange(desc(lift))
```

Metrics

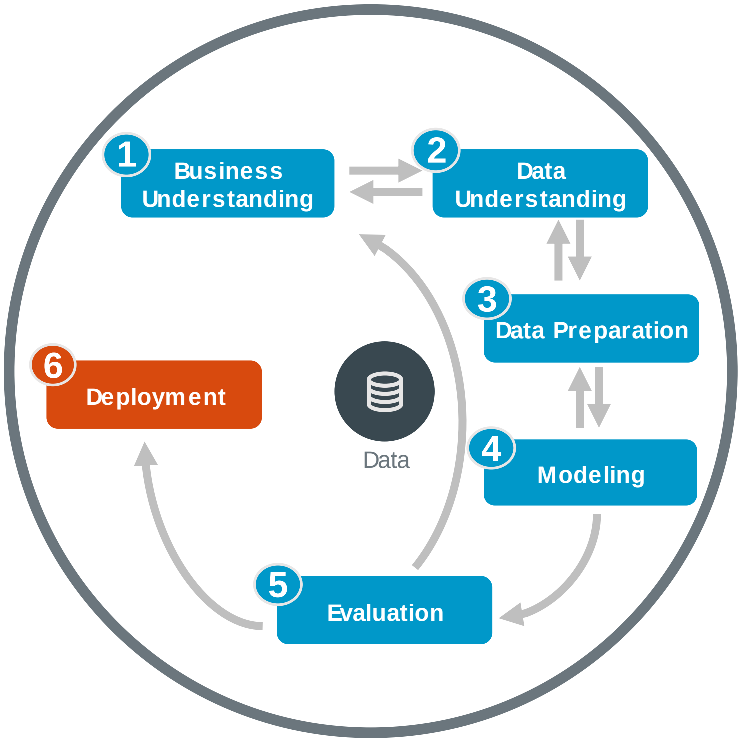
	col1_item1	col1_item2	cntb	cnt1	cnt2	score	support	confidence	lift	z_score
	<chr>	<chr>	<S3: integer64>	<S3: integer64>	<S3: integer64	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	Red Bull	Cup noodles	2	22	9	0.0202	0.00296	0.0909	6.83	4.17
2	Red Bull	Pistachio nuts	1	22	5	0.00909	0.00148	0.0455	6.15	-0.240
3	Red Bull	Licorice	1	22	5	0.00909	0.00148	0.0455	6.15	-0.240
4	Red Bull	Toaster pastries	2	22	12	0.0152	0.00296	0.0909	5.12	4.17
5	Red Bull	Cola	2	22	12	0.0152	0.00296	0.0909	5.12	4.17
6	Red Bull	Cheese puffs	1	22	7	0.00649	0.00148	0.0455	4.39	-0.240
7	Red Bull	Geplak	1	22	7	0.00649	0.00148	0.0455	4.39	-0.240
8	Red Bull	Nachos	1	22	8	0.00568	0.00148	0.0455	3.84	-0.240
9	Red Bull	Meze	1	22	8	0.00568	0.00148	0.0455	3.84	-0.240
10	Red Bull	Confections	1	22	8	0.00568	0.00148	0.0455	3.84	-0.240

Step 6a. Deployment (Operationalizing/Monitoring/Maintenance)

6. **Deployment** – The end goal is to "operationalize" the analytic findings. Taking analytics from insight to impact – the process of getting analytics out to business stakeholders for use/reuse to meet business goals

   a) **Plan deployment (how to operationalize)**

   Note: This varies by customer and is not covered in this course



CRISP-DM

Step 6b. Deployment (Operationalizing/Monitoring/Maintenance)

b) Plan monitoring and maintenance – Once operationalized, it's important to monitor and maintain it. Does your process need to be revisited as time marches on? Consider the following:

1. Do you still carry Red Bull?
2. Do you still carry the affinity products to Red Bull?
3. Are there any new products that you started carrying since you last ran your analysis? Might any of these products have an affinity to Red Bull?
4. Have the purchasing habits of your customers changed?

Note: This step is the same regardless of language; i.e., Python, R, or SQL

Thank you.

©2022 Teradata