



# Big Data Mining

## Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data

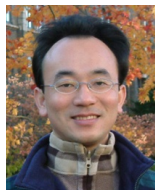
1071BDM03

TLVXM1A (M2244) (8619) (Fall 2018)

(MBA, DBETKU) (3 Credits, Required) [Full English Course]

(Master's Program in Digital Business and Economics)

Mon, 9, 10, 11, (16:10-19:00) (B206)



Min-Yuh Day, Ph.D.

Assistant Professor

Department of Information Management

Tamkang University

<http://mail.tku.edu.tw/myday>

2018-10-01



# Course Schedule (1/2)



Tamkang  
University

Week	Date	Subject/Topics
1	2018/09/10	Course Orientation for Big Data Mining
2	2018/09/17	ABC: AI, Big Data, Cloud Computing
3	2018/09/24	Mid-Autumn Festival (Day off)
4	2018/10/01	Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data
5	2018/10/08	Fundamental Big Data: MapReduce Paradigm, Hadoop and Spark Ecosystem
6	2018/10/15	Foundations of Big Data Mining in Python
7	2018/10/22	Supervised Learning: Classification and Prediction
8	2018/10/29	Unsupervised Learning: Cluster Analysis
9	2018/11/05	Unsupervised Learning: Association Analysis



# Course Schedule (2/2)



Tamkang  
University

Week    Date    Subject/Topics

10    2018/11/12    Midterm Project Report

11    2018/11/19    Machine Learning with Scikit-Learn in Python

12    2018/11/26    Deep Learning for Finance Big Data with  
TensorFlow

13    2018/12/03    Convolutional Neural Networks (CNN)

14    2018/12/10    Recurrent Neural Networks (RNN)

15    2018/12/17    Reinforcement Learning (RL)

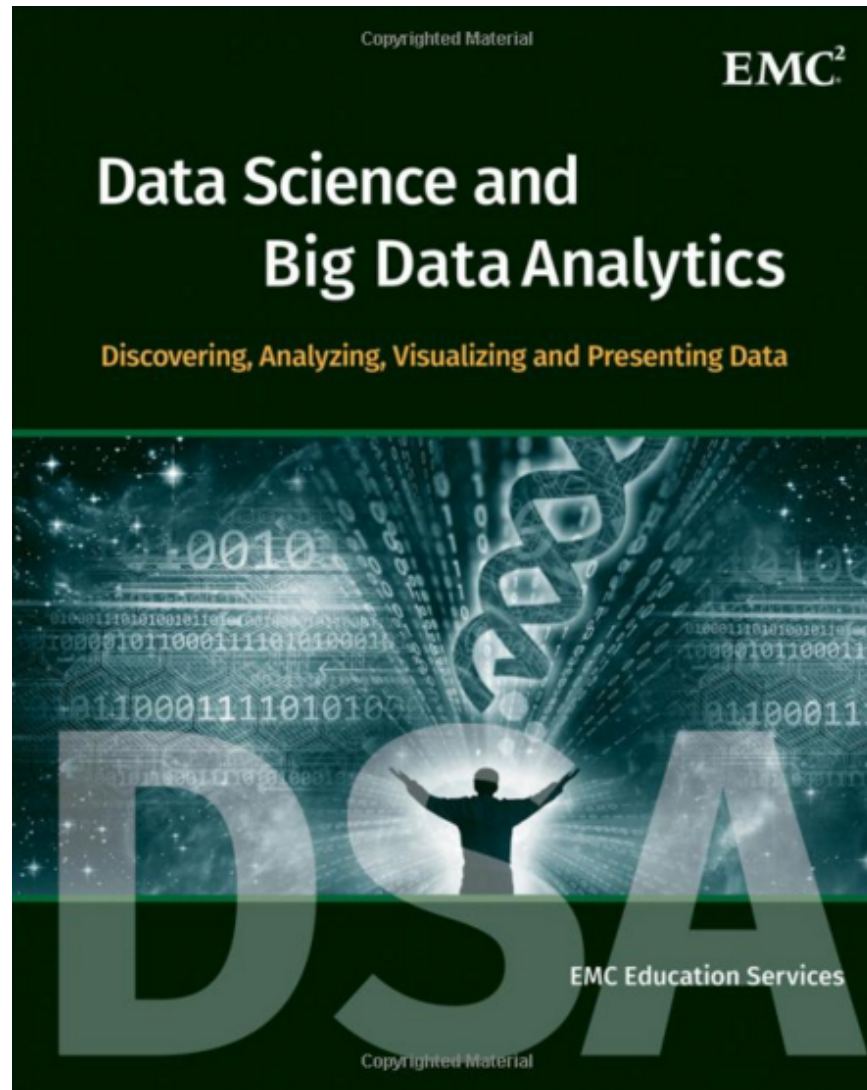
16    2018/12/24    Social Network Analysis (SNA)

17    2018/12/31    Bridge Holiday (Extra Day Off)

18    2019/01/07    Final Project Presentation

# **Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data**

**EMC Education Services,**  
**Data Science and Big Data Analytics:**  
**Discovering, Analyzing, Visualizing and Presenting Data,**  
**Wiley, 2015**



# Data Scientist:

## *The Sexiest Job of the 21st Century*

**Meet the people who  
can coax treasure out of  
messy, unstructured data.**

*by Thomas H. Davenport  
and D.J. Patil*

**W**hen Jonathan Goldman arrived for work in June 2006 at LinkedIn, the business networking site, the place still felt like a start-up. The company had just under 8 million accounts, and the number was growing quickly as existing members invited their friends and colleagues to join. But users weren't seeking out connections with the people who were already on the site at the rate executives had expected. Something was apparently missing in the social experience. As one LinkedIn manager put it, "It was like arriving at a conference reception and realizing you don't know anyone. So you just stand in the corner sipping your drink—and you probably leave early."

# Data Science

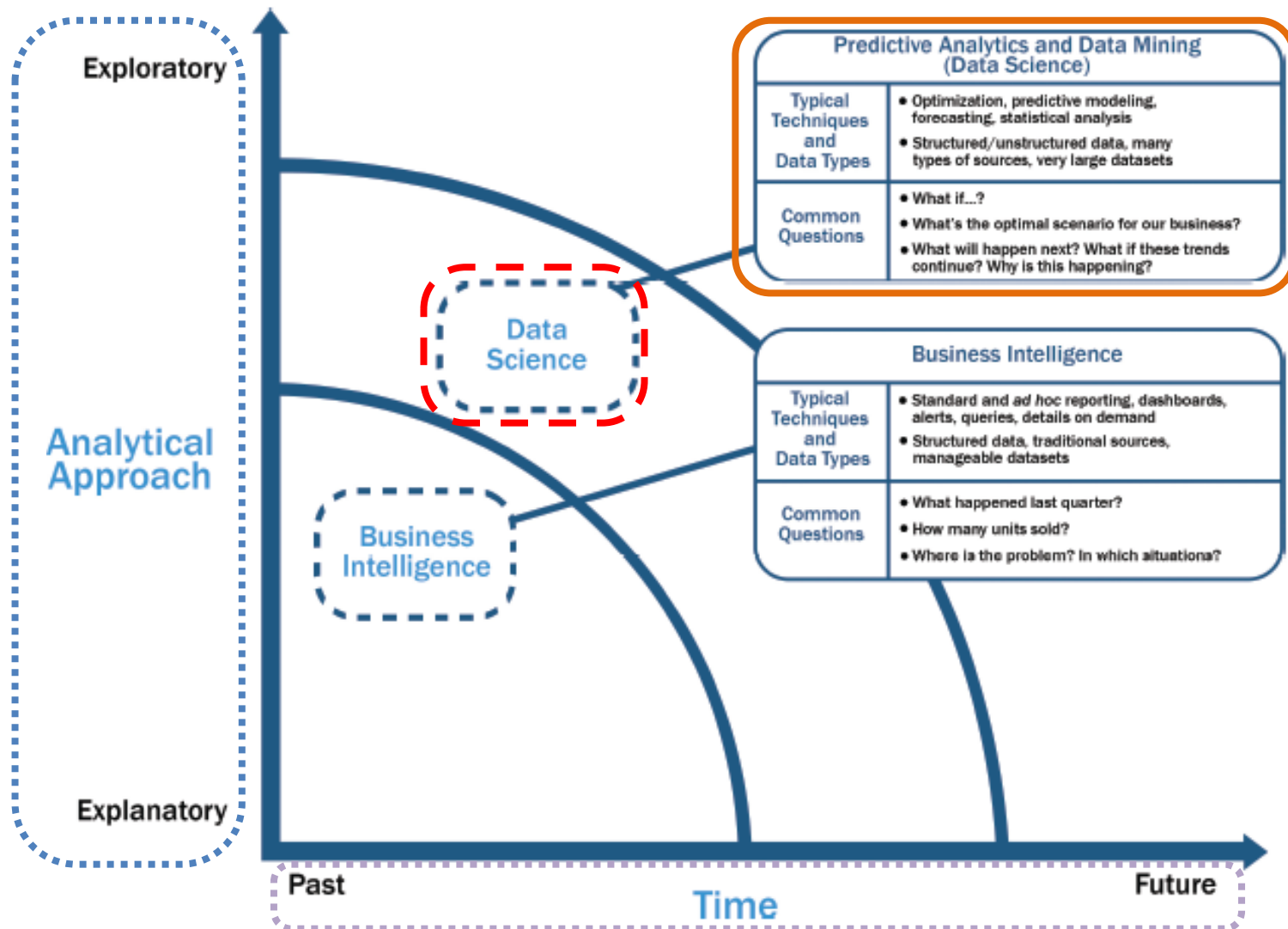
# Data Analyst

- Data analyst is just another term for professionals who were doing **BI** in the form of **data compilation, cleaning, reporting**, and perhaps some **visualization**.
- Their skill sets included Excel, some SQL knowledge, and reporting.
- You would recognize those capabilities as **descriptive** or **reporting analytics**.

# Data Scientist

- Data scientist is responsible for **predictive analysis, statistical analysis**, and more **advanced analytical tools and algorithms**.
- They may have a deeper knowledge of algorithms and may recognize them under various labels—**data mining, knowledge discovery, or machine learning**.
- Some of these professionals may also need deeper programming knowledge to be able to write code for data cleaning/analysis in current Web-oriented languages such as Java or Python and statistical languages such as R.
- Many analytics professionals also need to build significant expertise in **statistical modeling, experimentation, and analysis**.

# Data Science and Business Intelligence





# Data Science and Business Intelligence



## Predictive Analytics and Data Mining (Data Science)

# Predictive Analytics and Data Mining (Data Science)

Structured/unstructured data, many types of sources,  
very large datasets

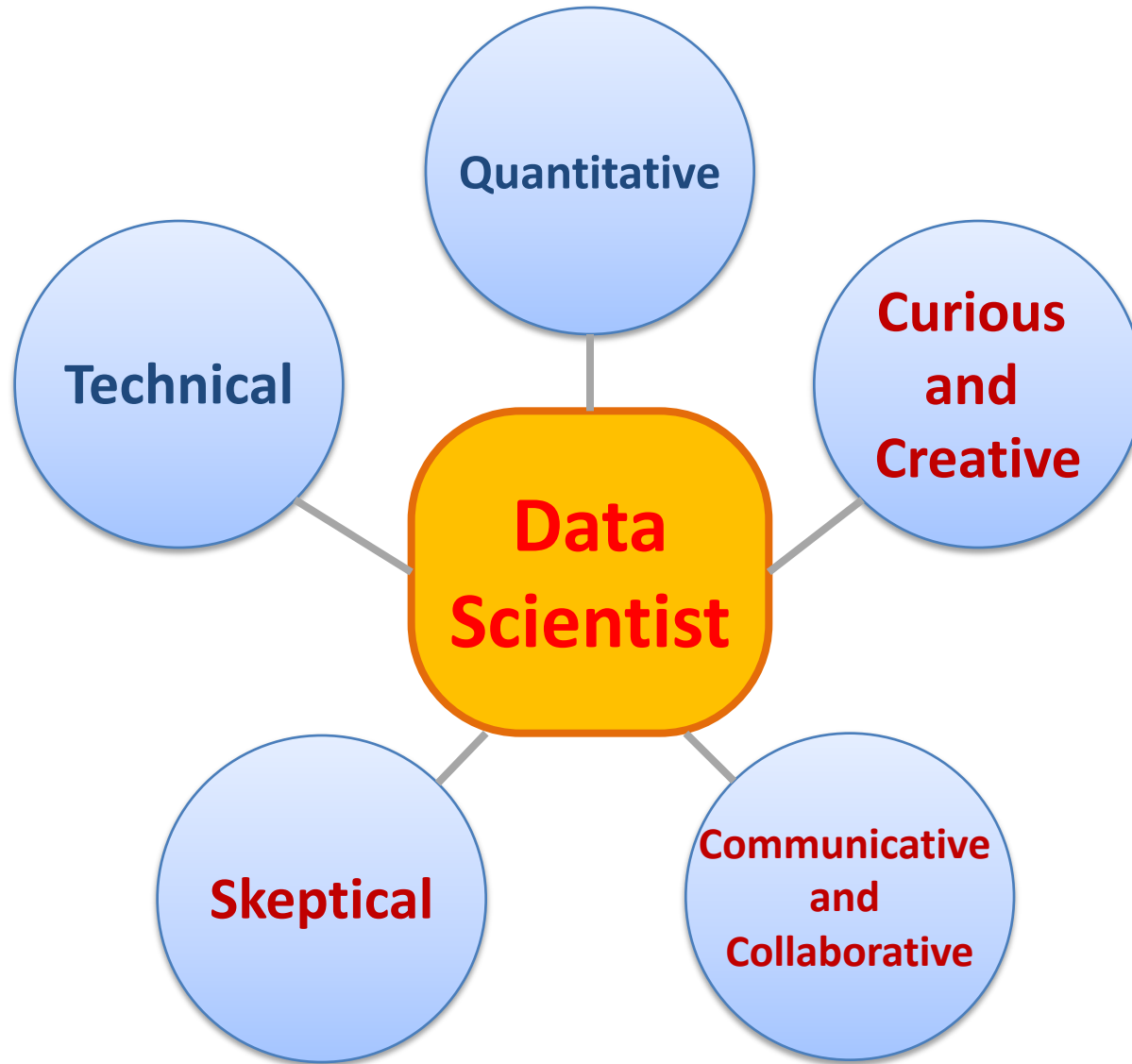
Optimization, predictive modeling, forecasting statistical analysis

What if...?  
What's the optimal scenario for our business?  
What will happen next?  
What if these trends continue?  
Why is this happening?

# Profile of a Data Scientist

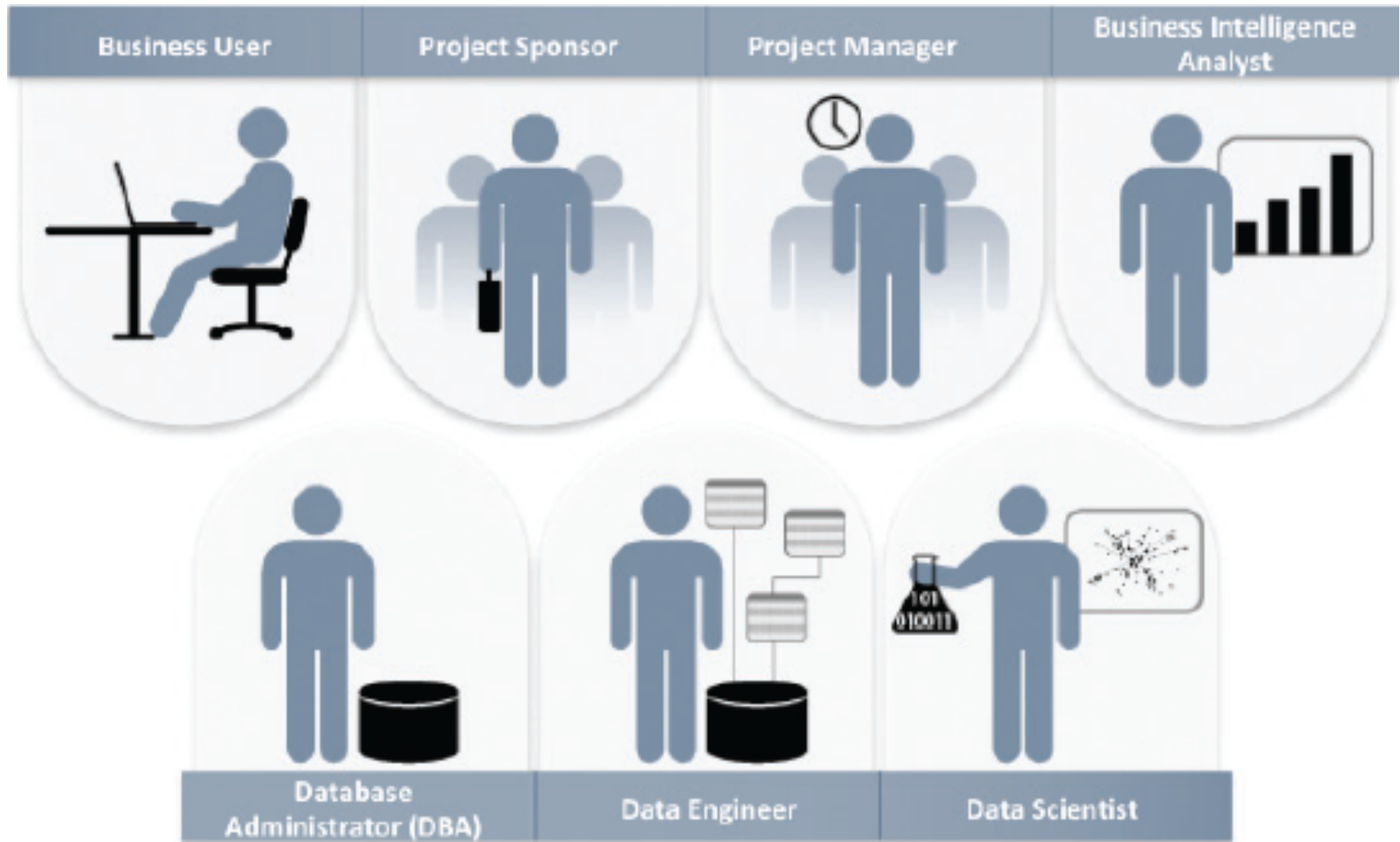
- **Quantitative**
  - mathematics or statistics
- **Technical**
  - software engineering, machine learning, and programming skills
- **Skeptical mind-set** and **critical thinking**
- **Curious** and **creative**
- **Communicative** and **collaborative**

# Data Scientist Profile

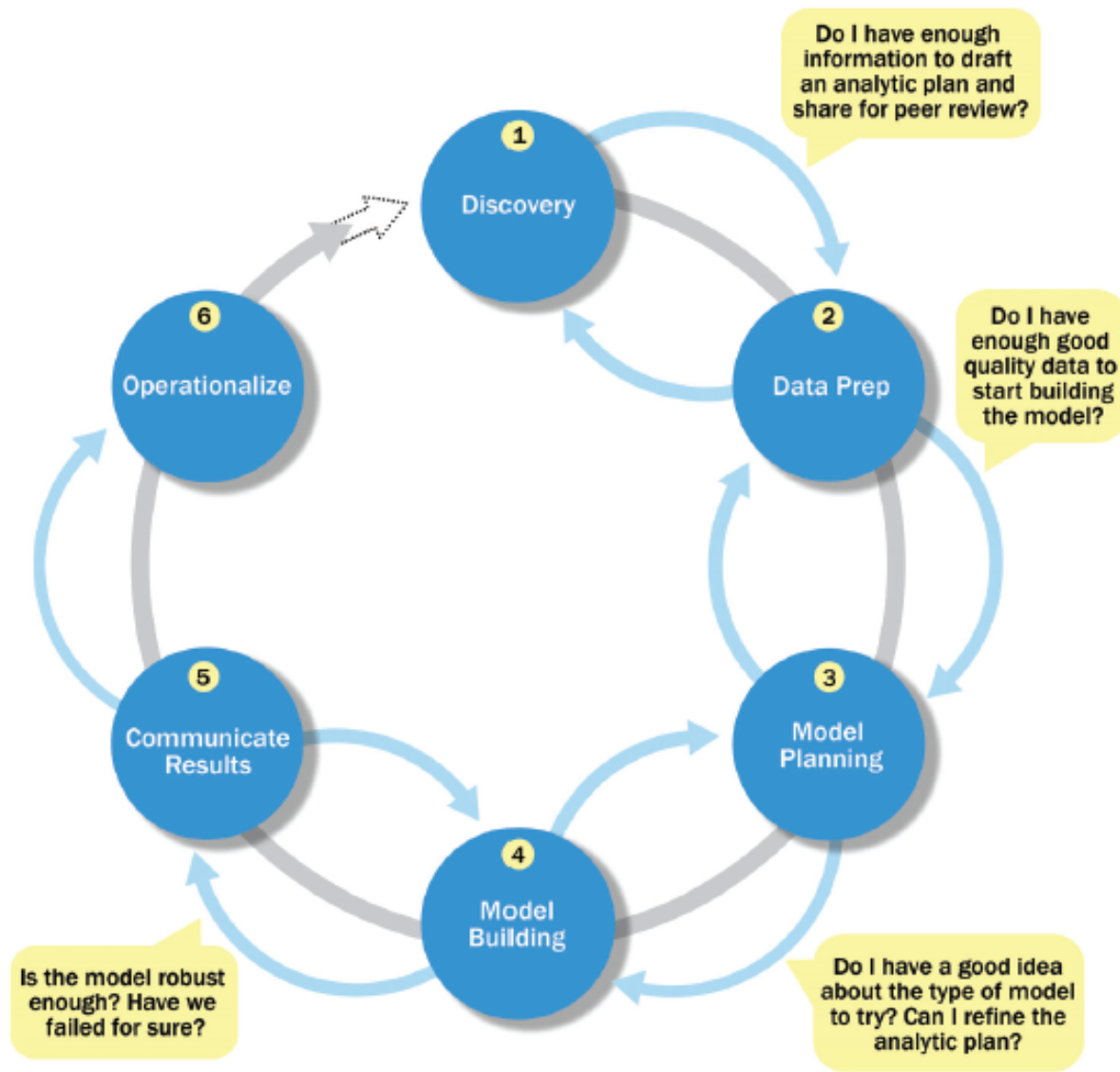


# **Big Data Analytics Lifecycle**

# Key Roles for a Successful Analytics Project



# Overview of Data Analytics Lifecycle

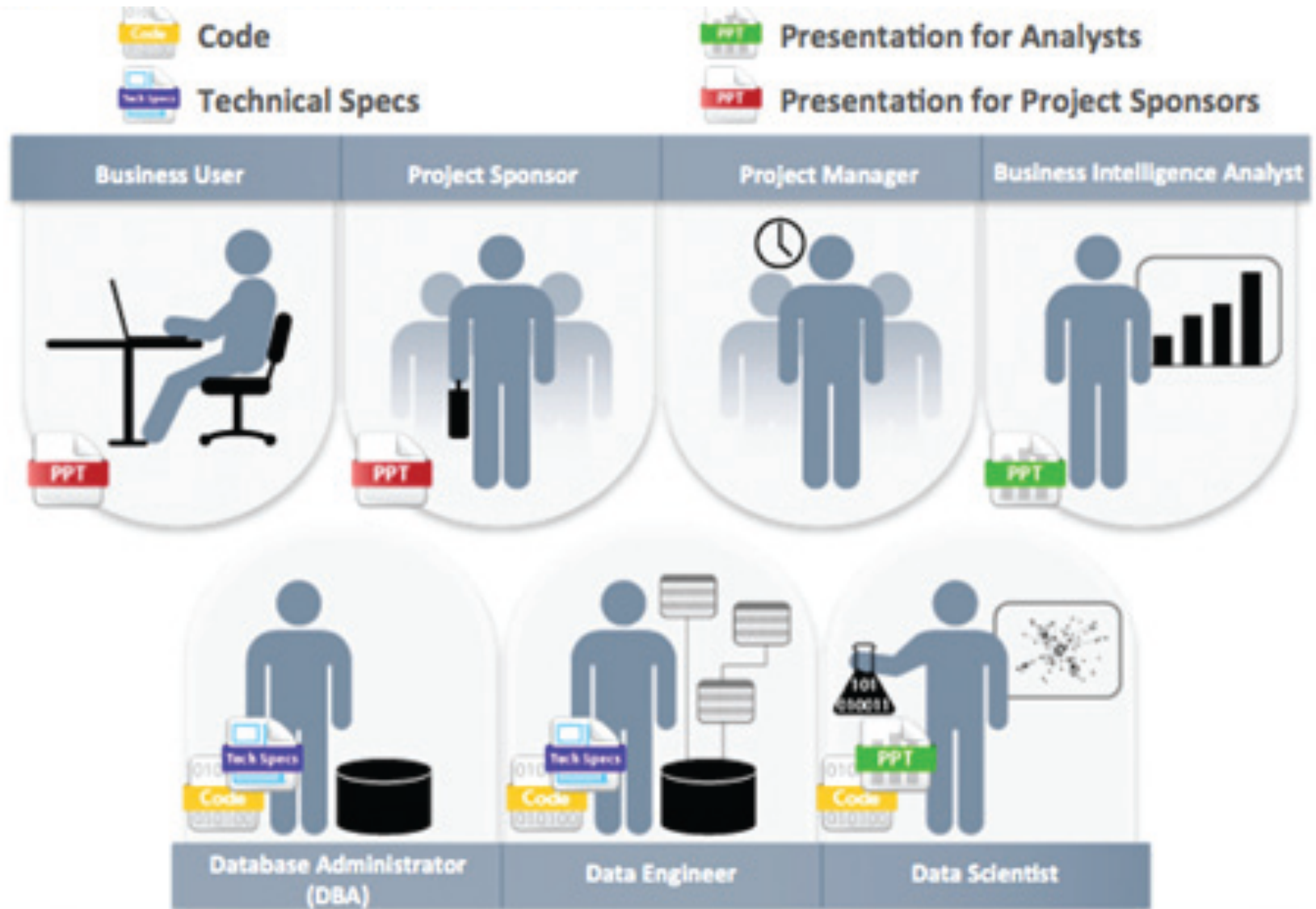


# Overview of Data Analytics Lifecycle

1. Discovery
2. Data preparation
3. Model planning
4. Model building
5. Communicate results
6. Operationalize



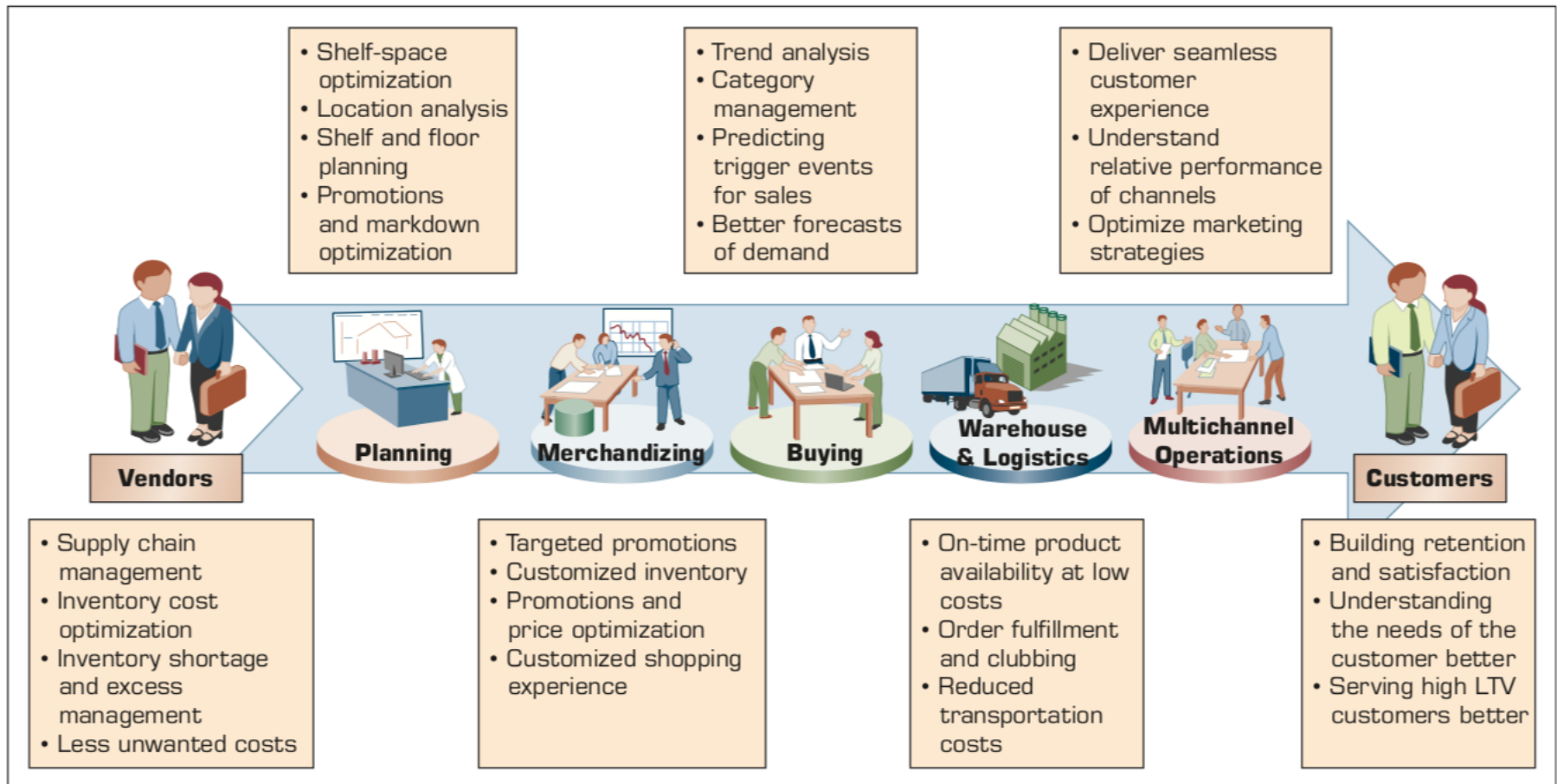
# Key Outputs from a Successful Analytics Project



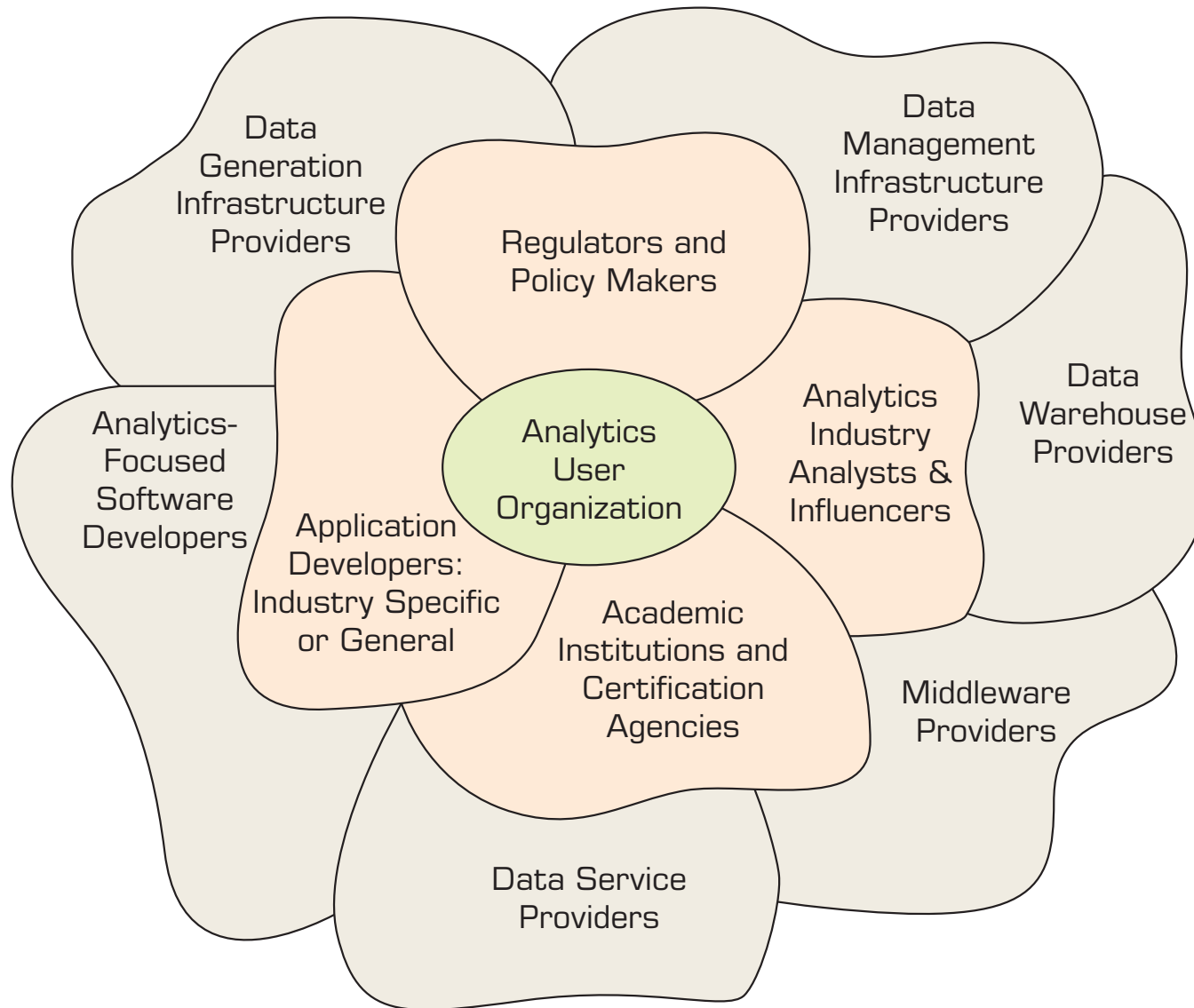
# Example of Analytics Applications in a Retail Value Chain

## Retail Value Chain

Critical needs at every touch point of the Retail Value Chain



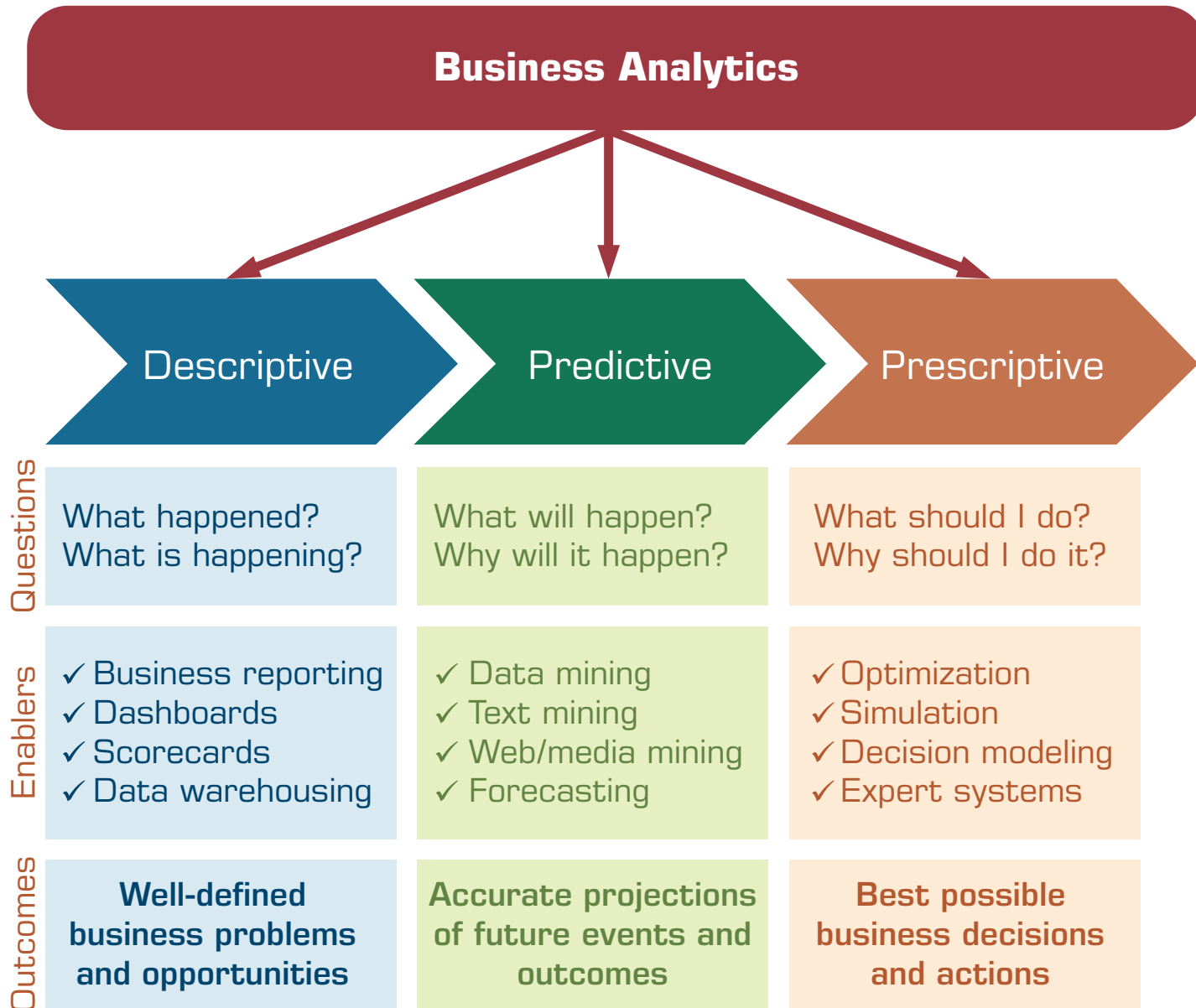
# Analytics Ecosystem



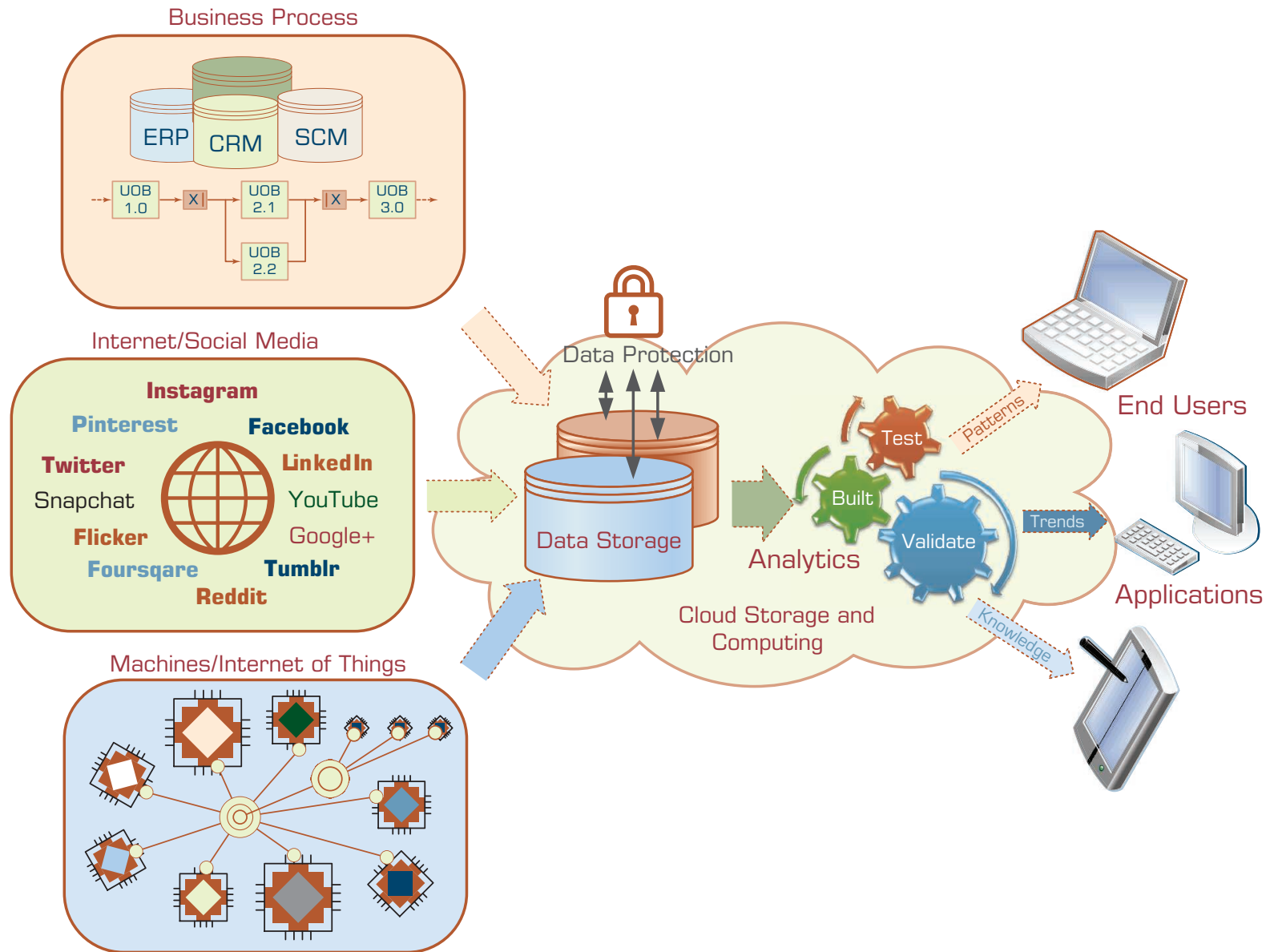
# Job Titles of Analytics



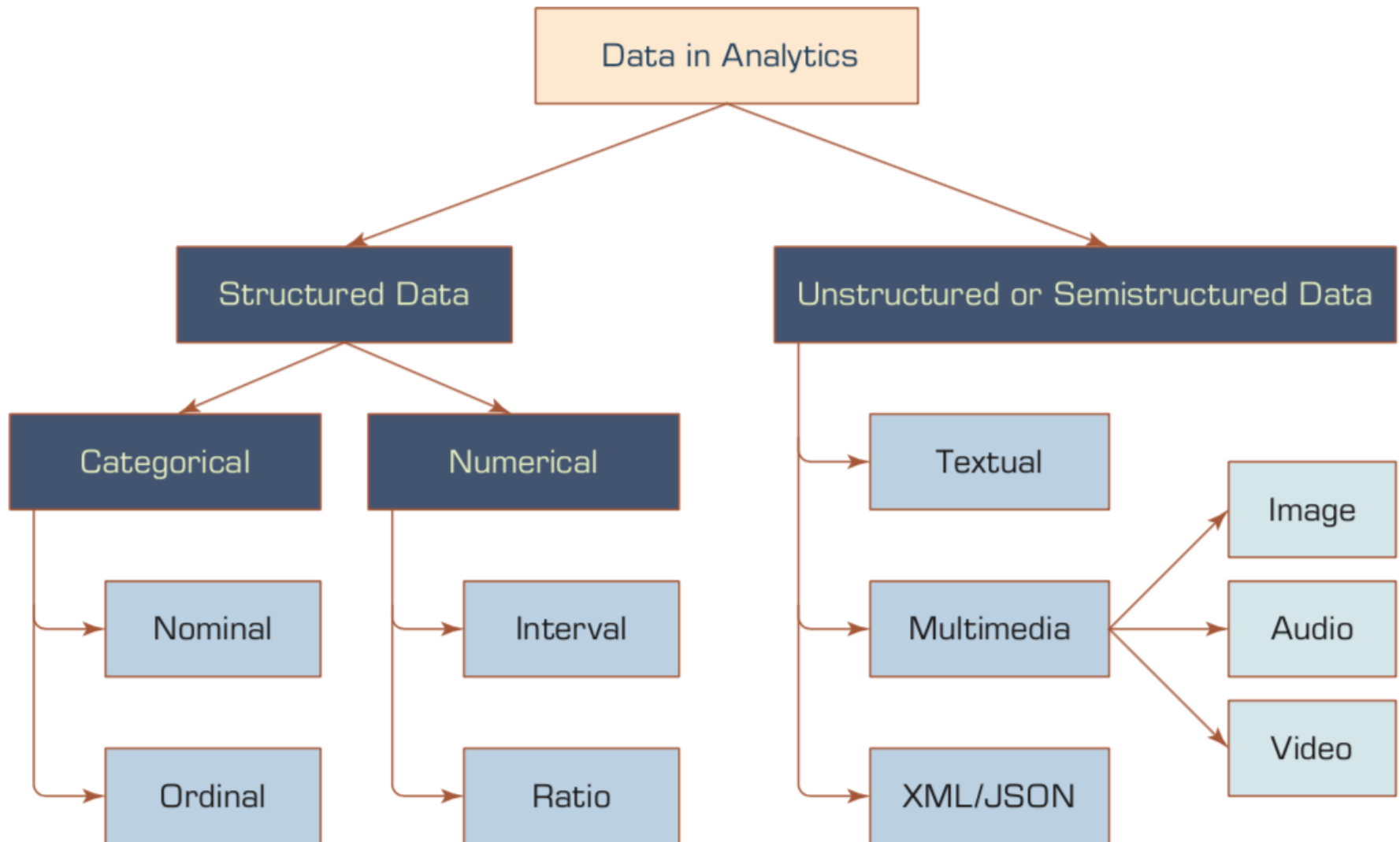
# Three Types of Analytics



# A Data to Knowledge Continuum

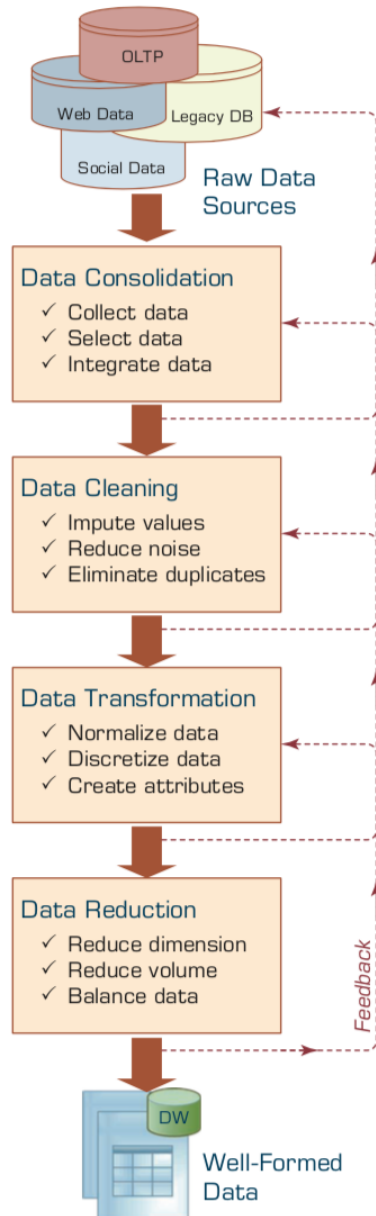


# A Simple Taxonomy of Data



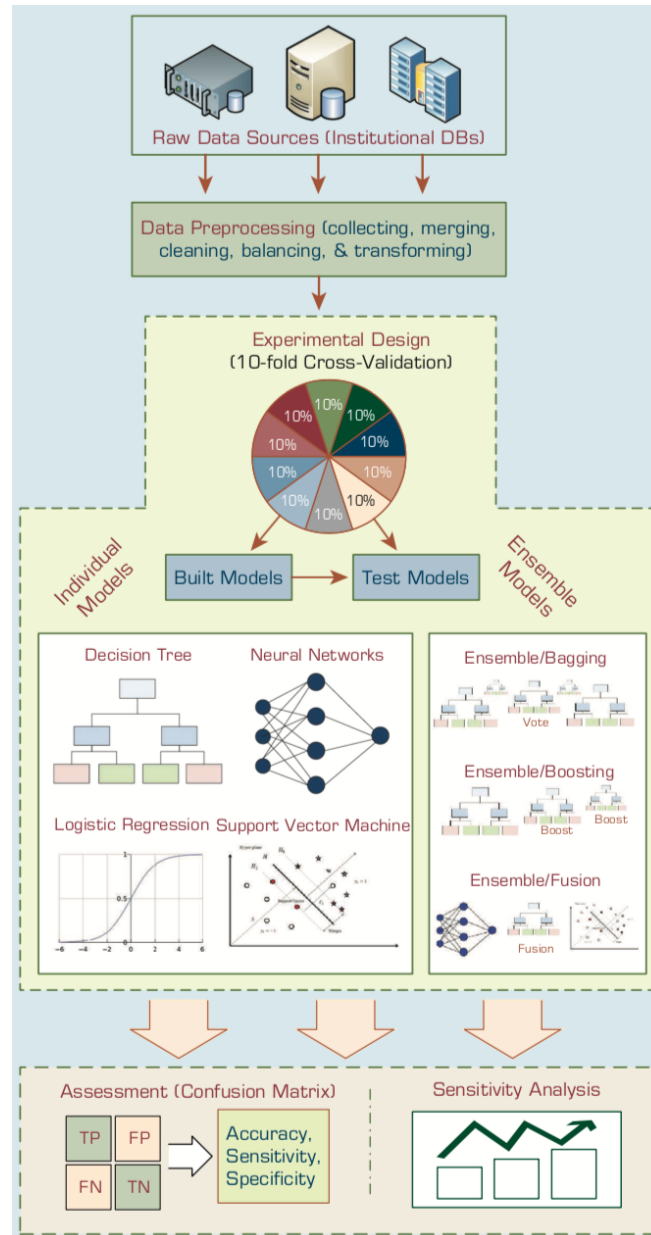


# Data Preprocessing Steps

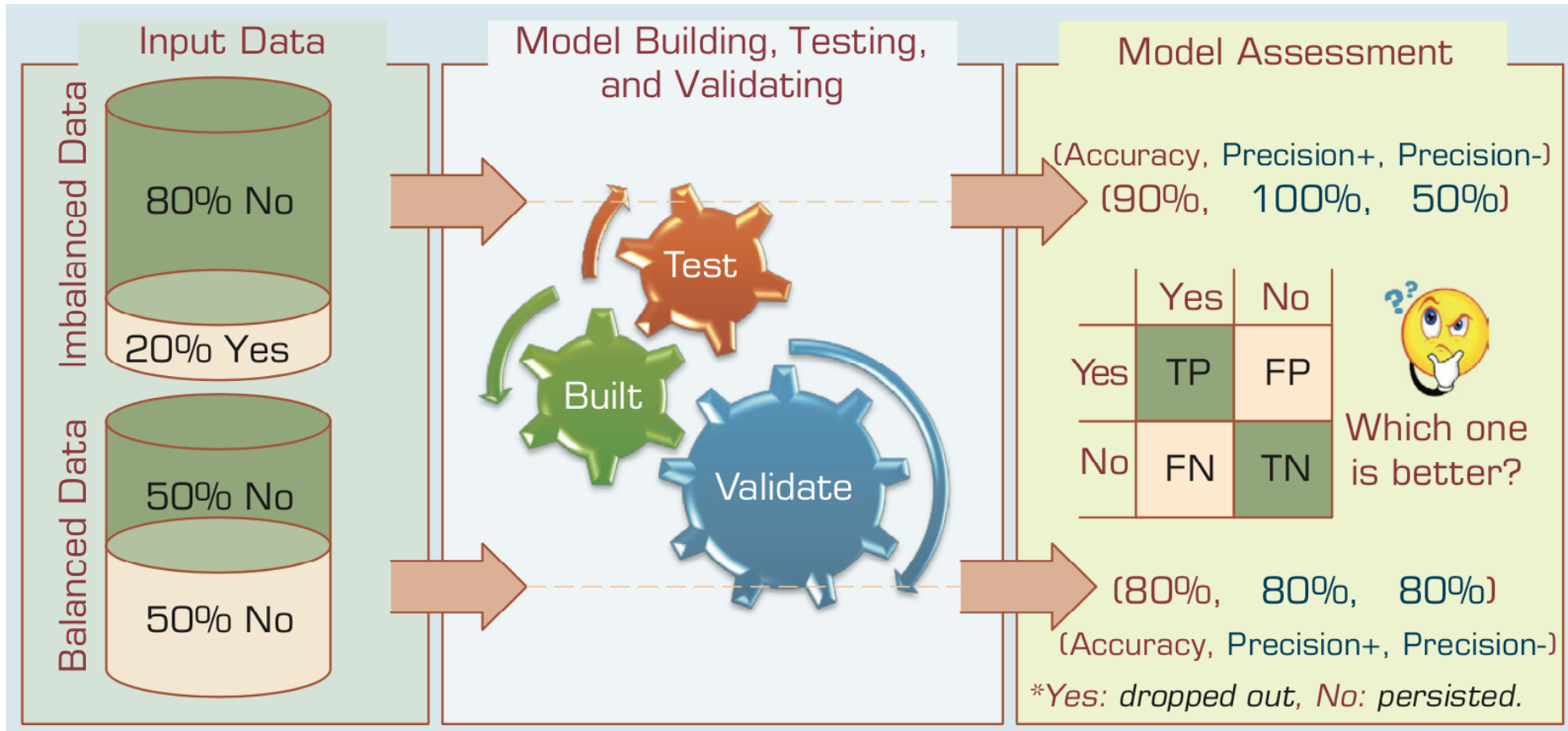




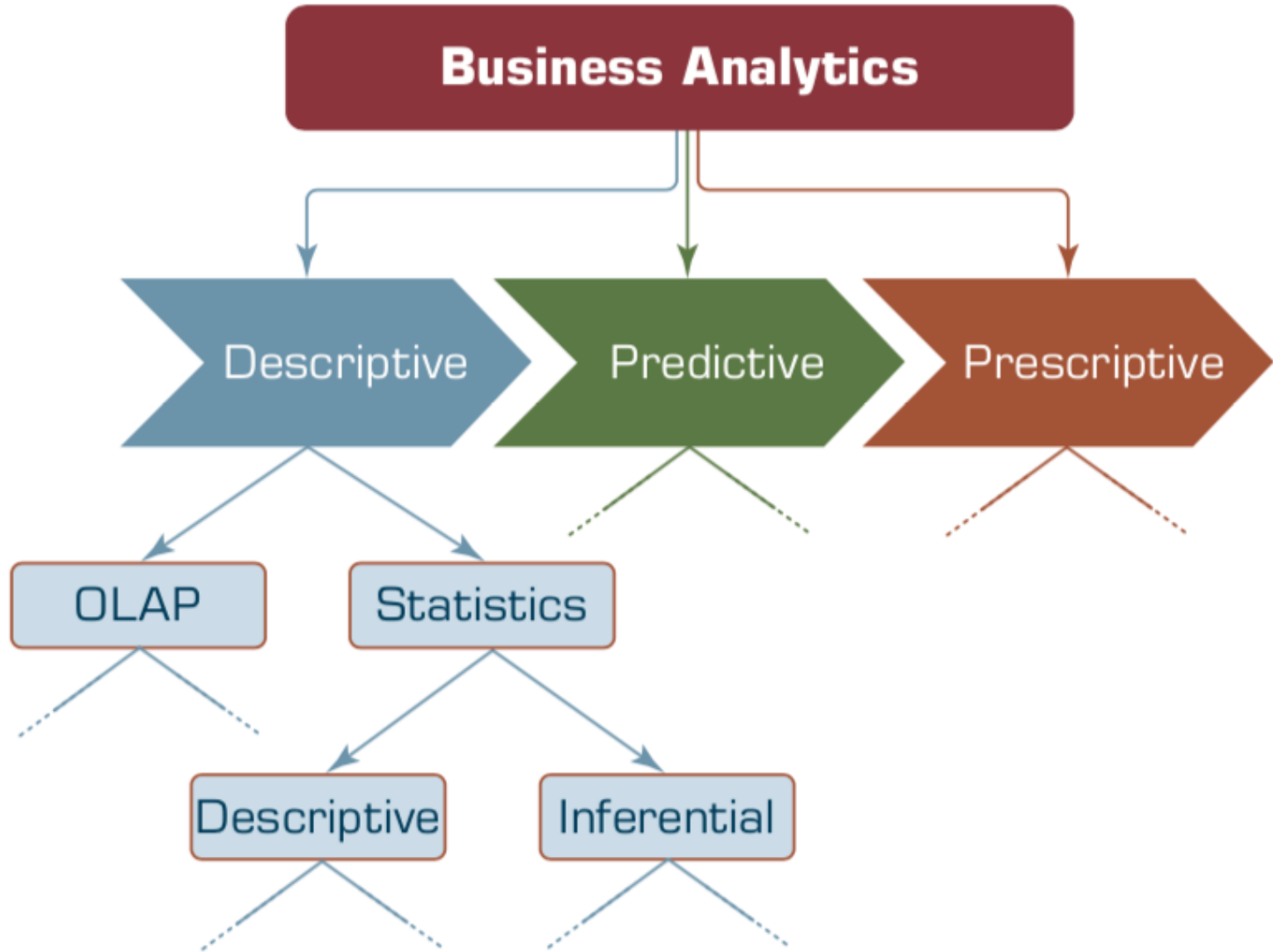
# An Analytics Approach to Predicting Student Attrition



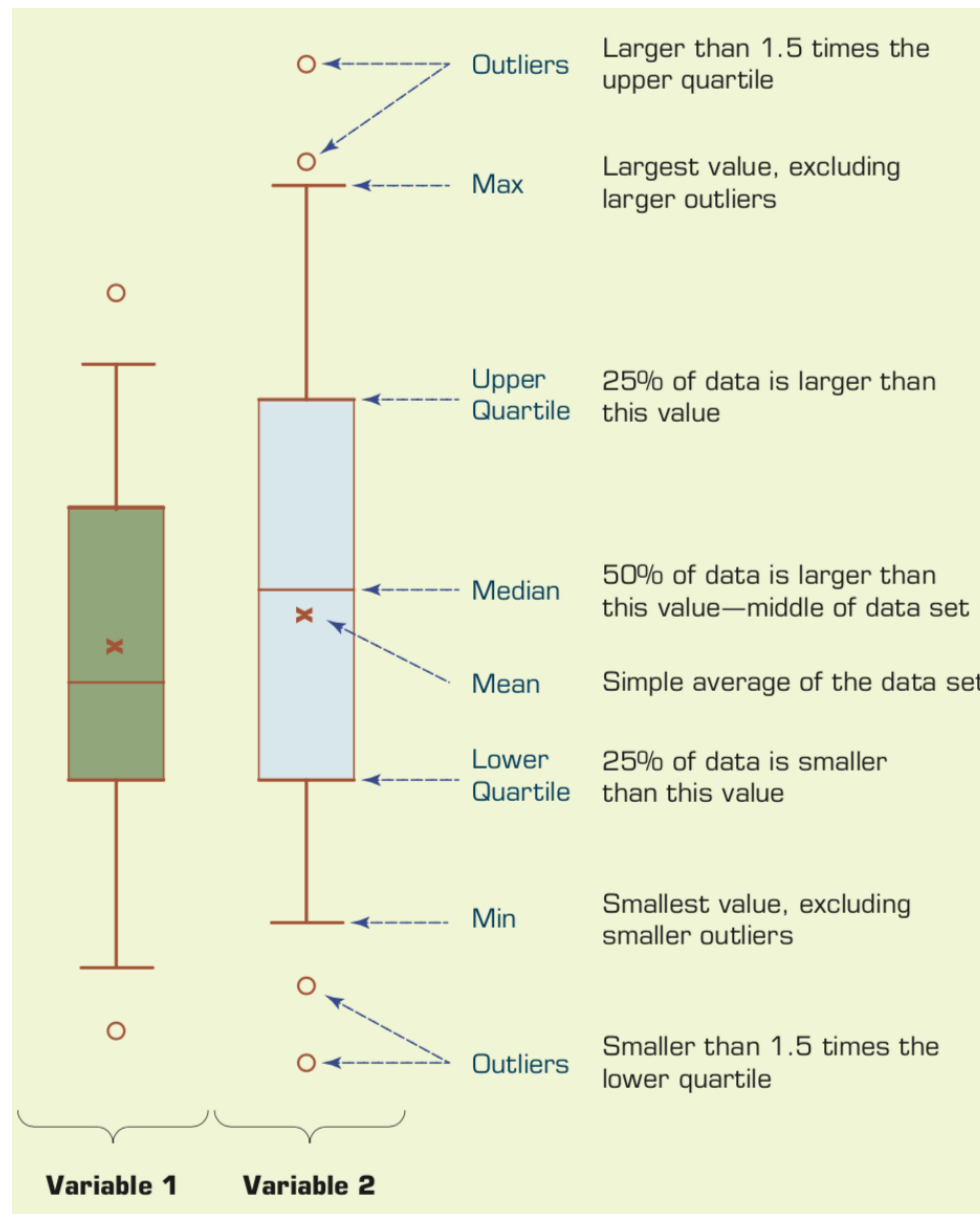
# A Graphical Depiction of the Class Imbalance Problem



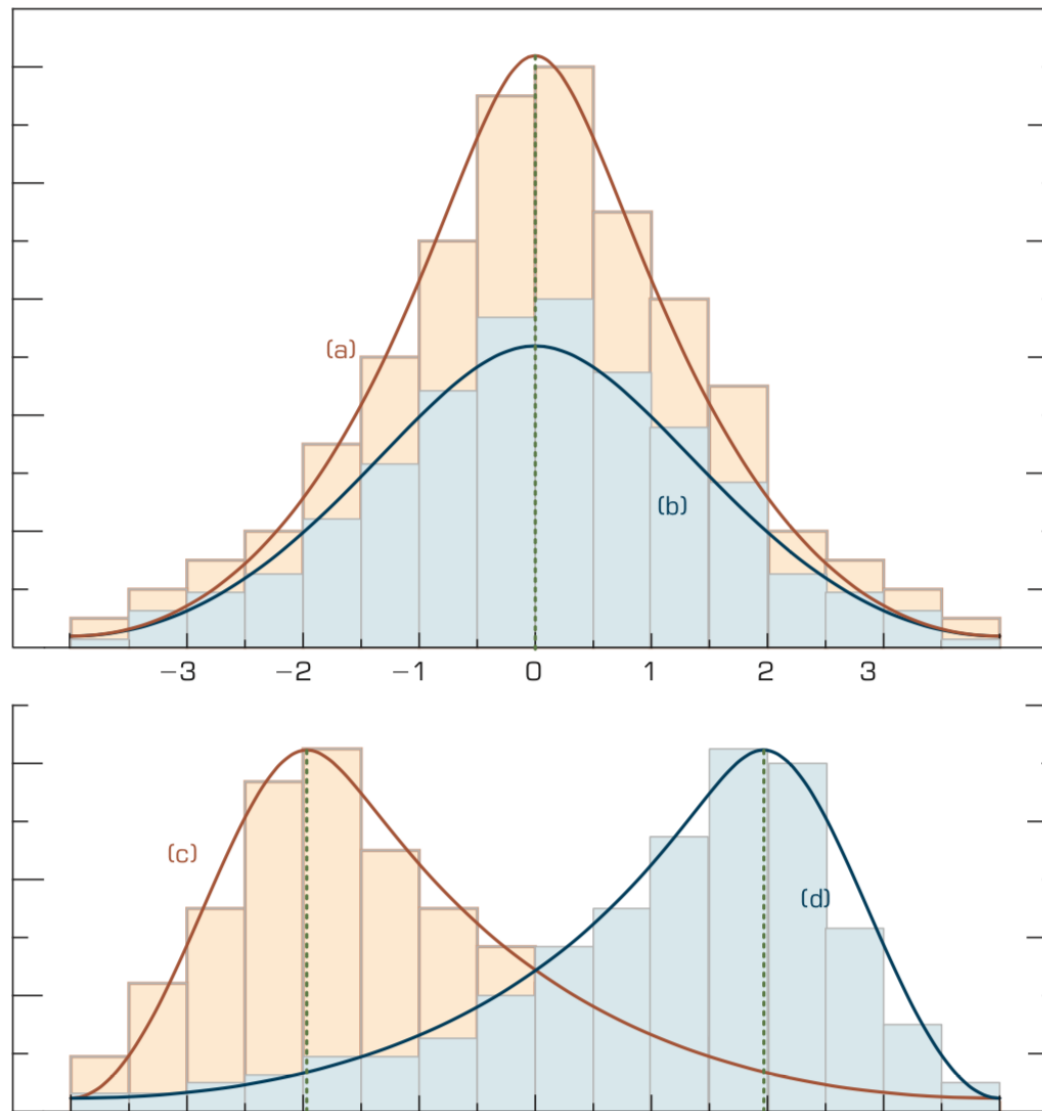
# Relationship between Statistics and Descriptive Analytics



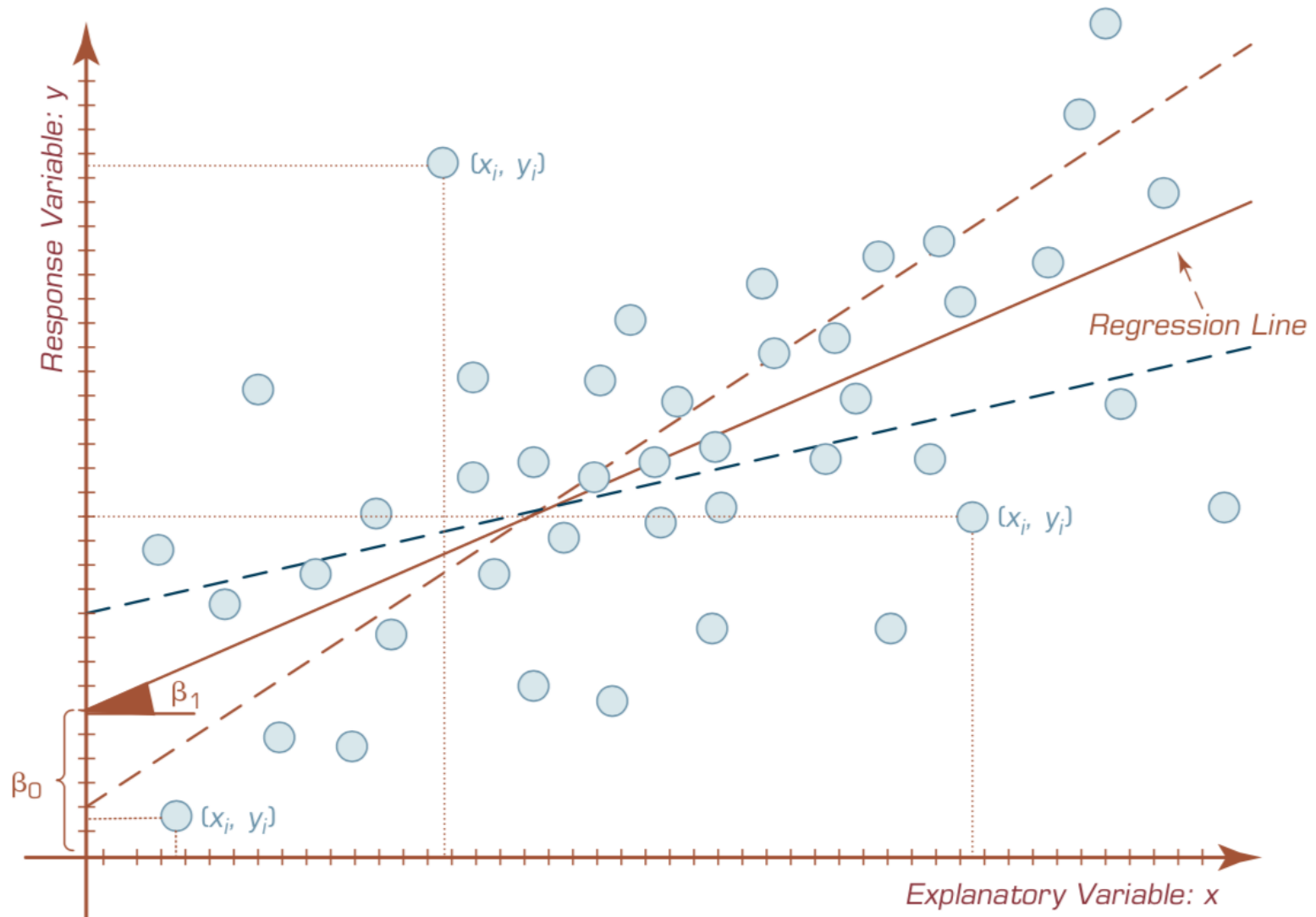
# Understanding the Specifics about Box-and-Whiskers Plots



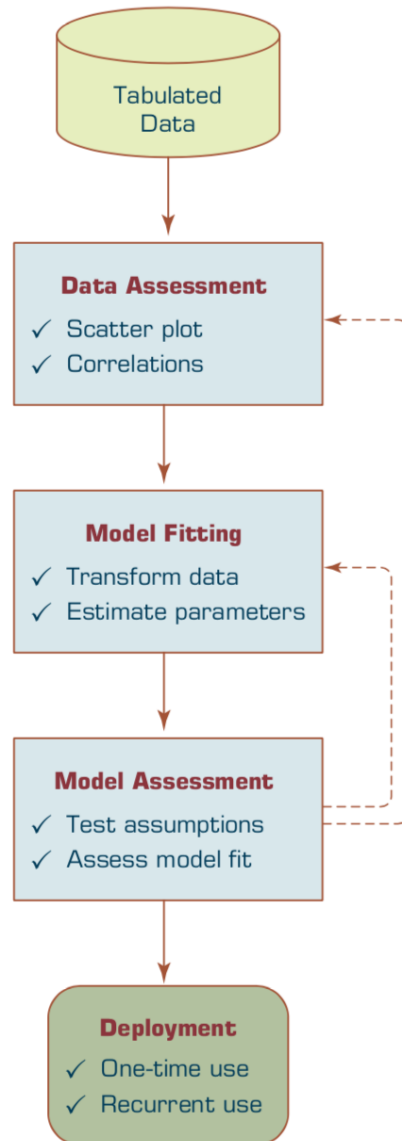
# Relationship between Dispersion and Shape Properties.



# A Scatter Plot and a Linear Regression Line

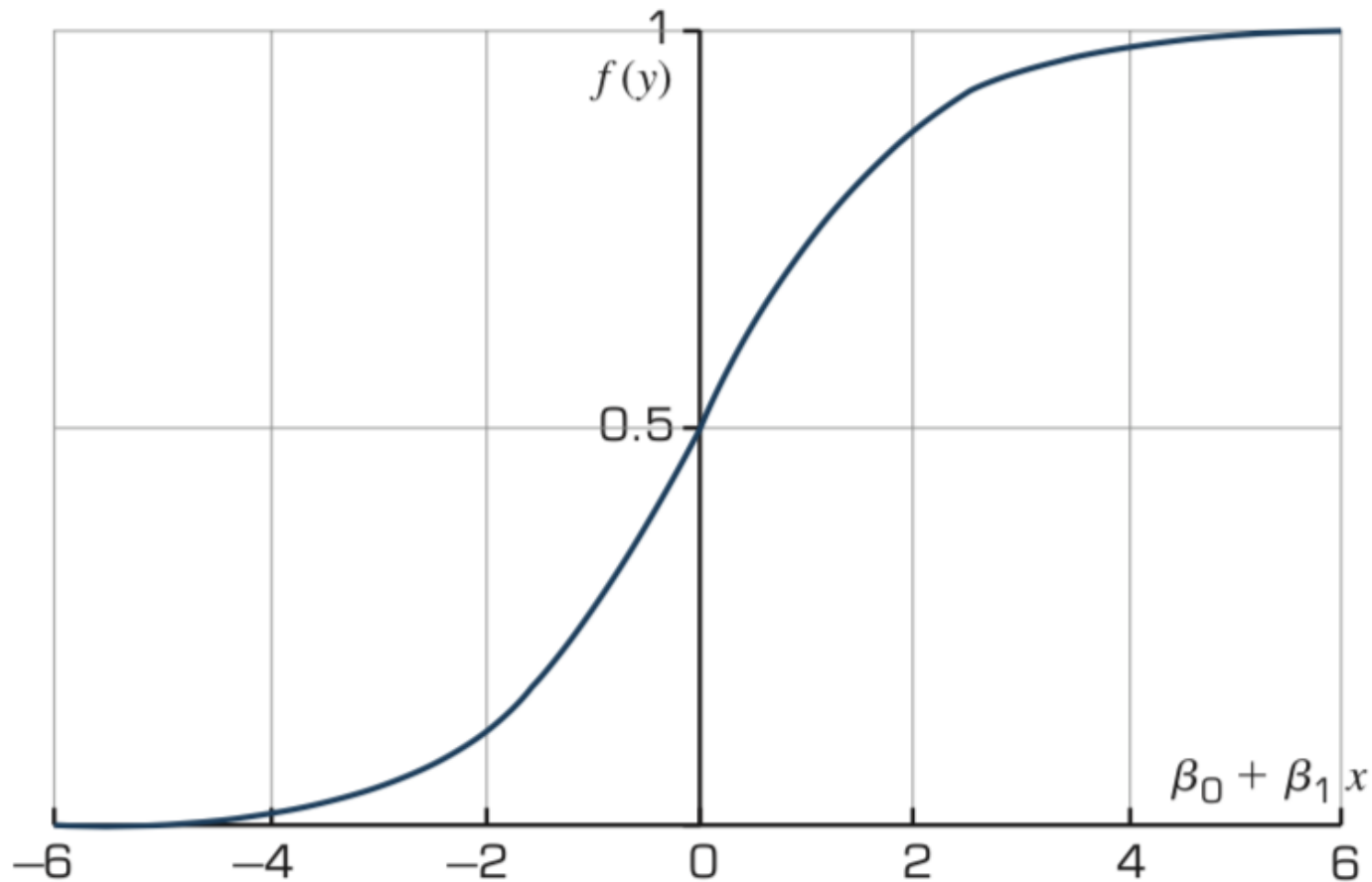


# A Process Flow for Developing Regression Models.



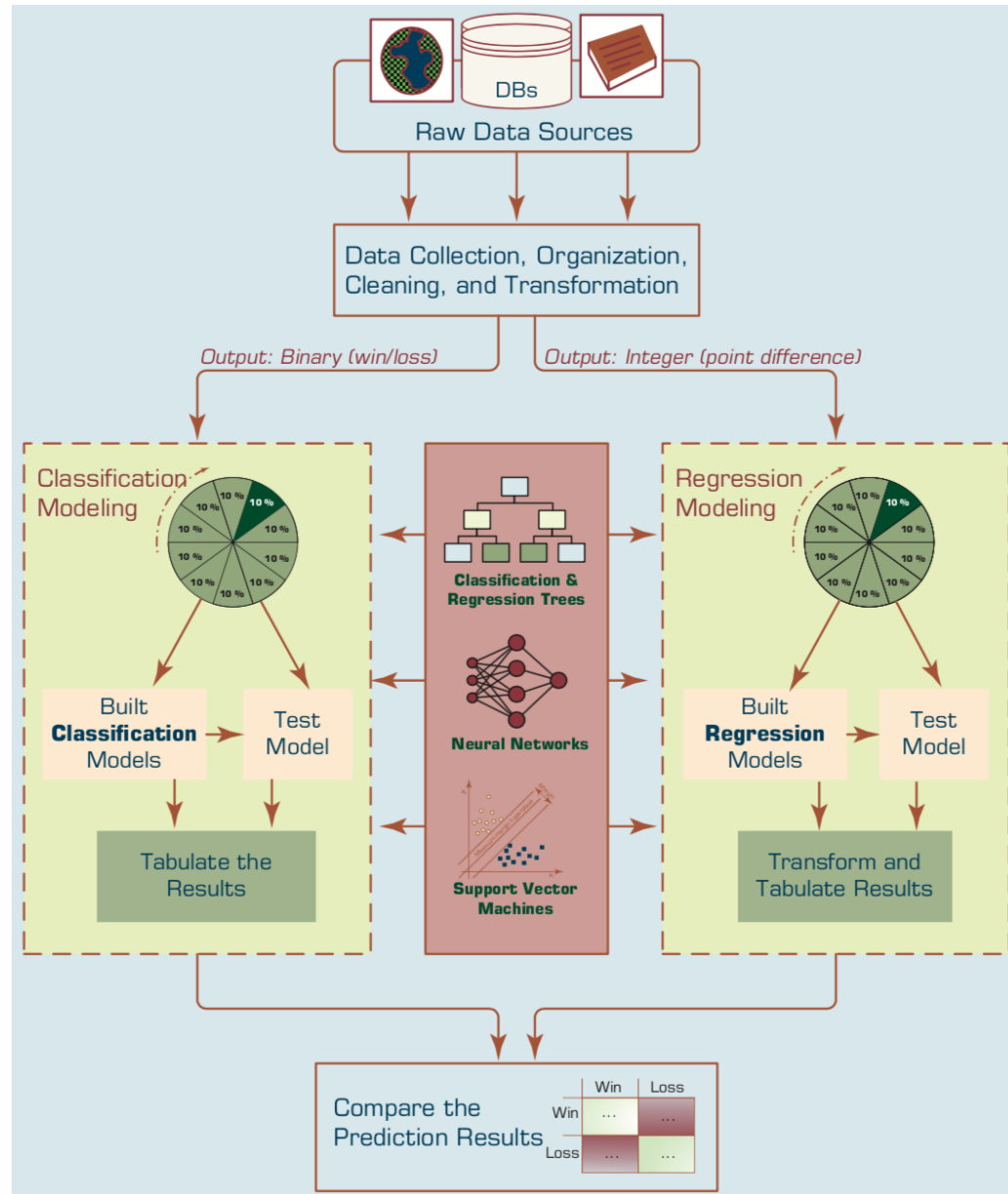
# The Logistic Function

$$f(y) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$$

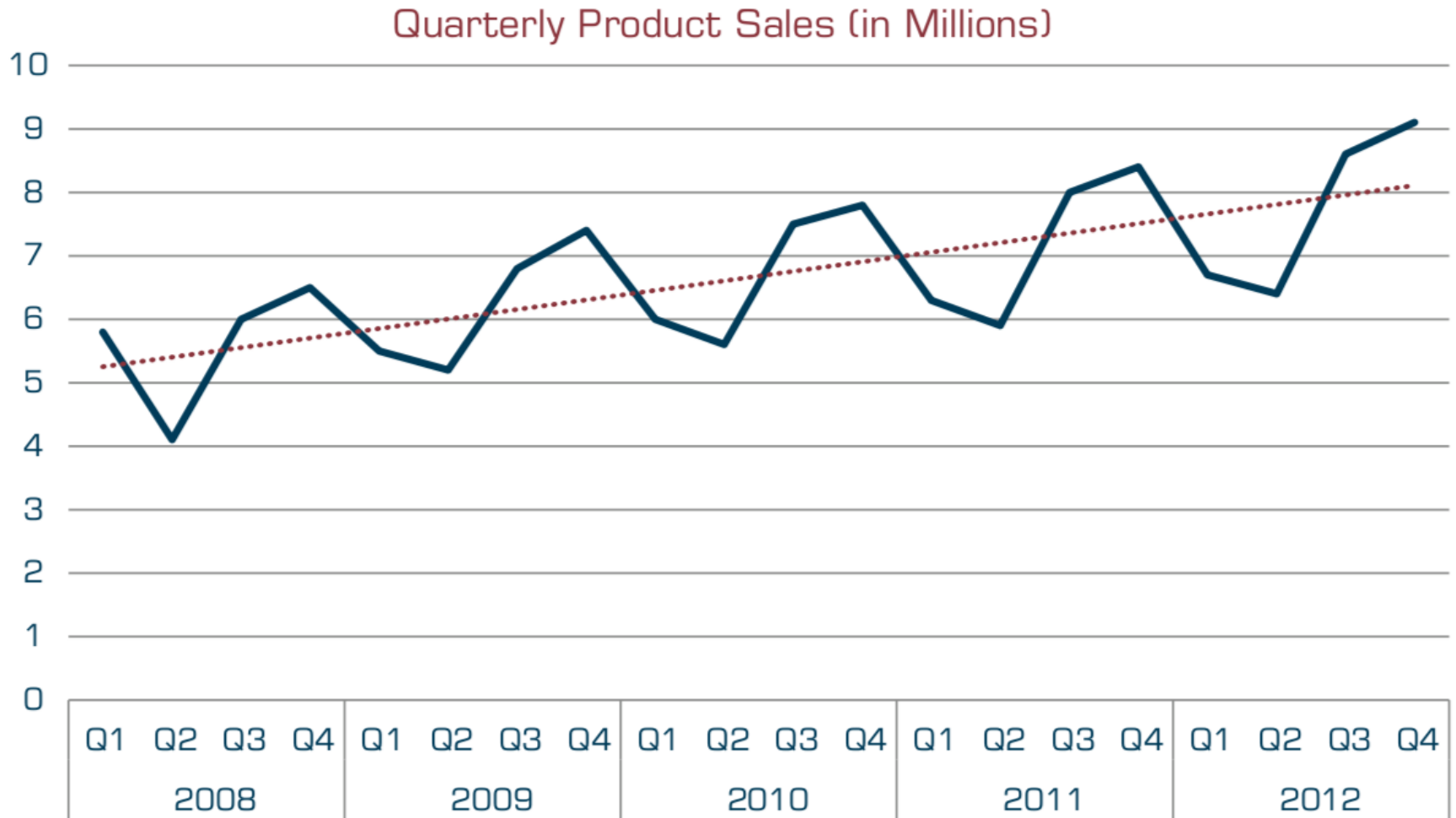




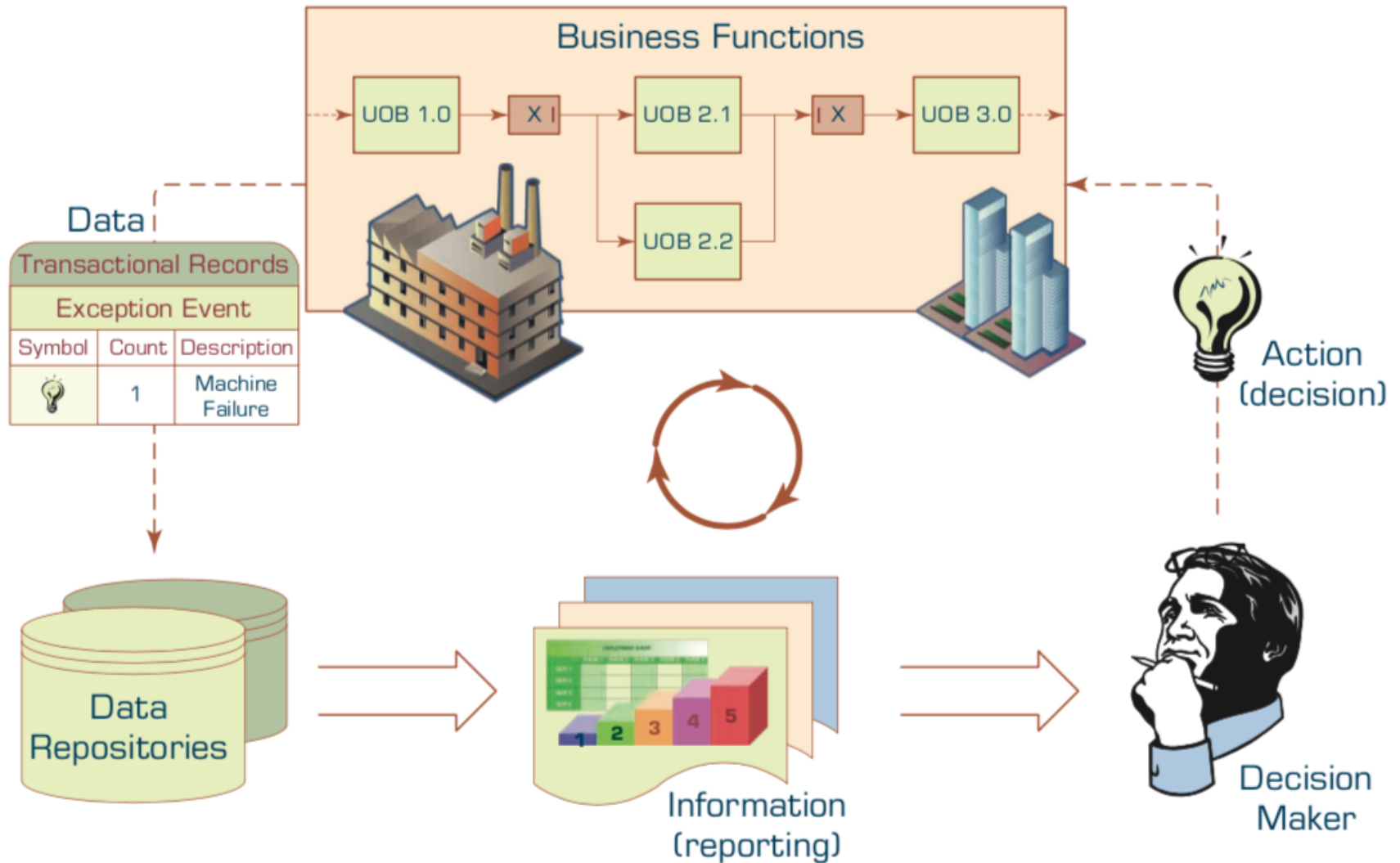
# Predicting NCAA Bowl Game Outcomes



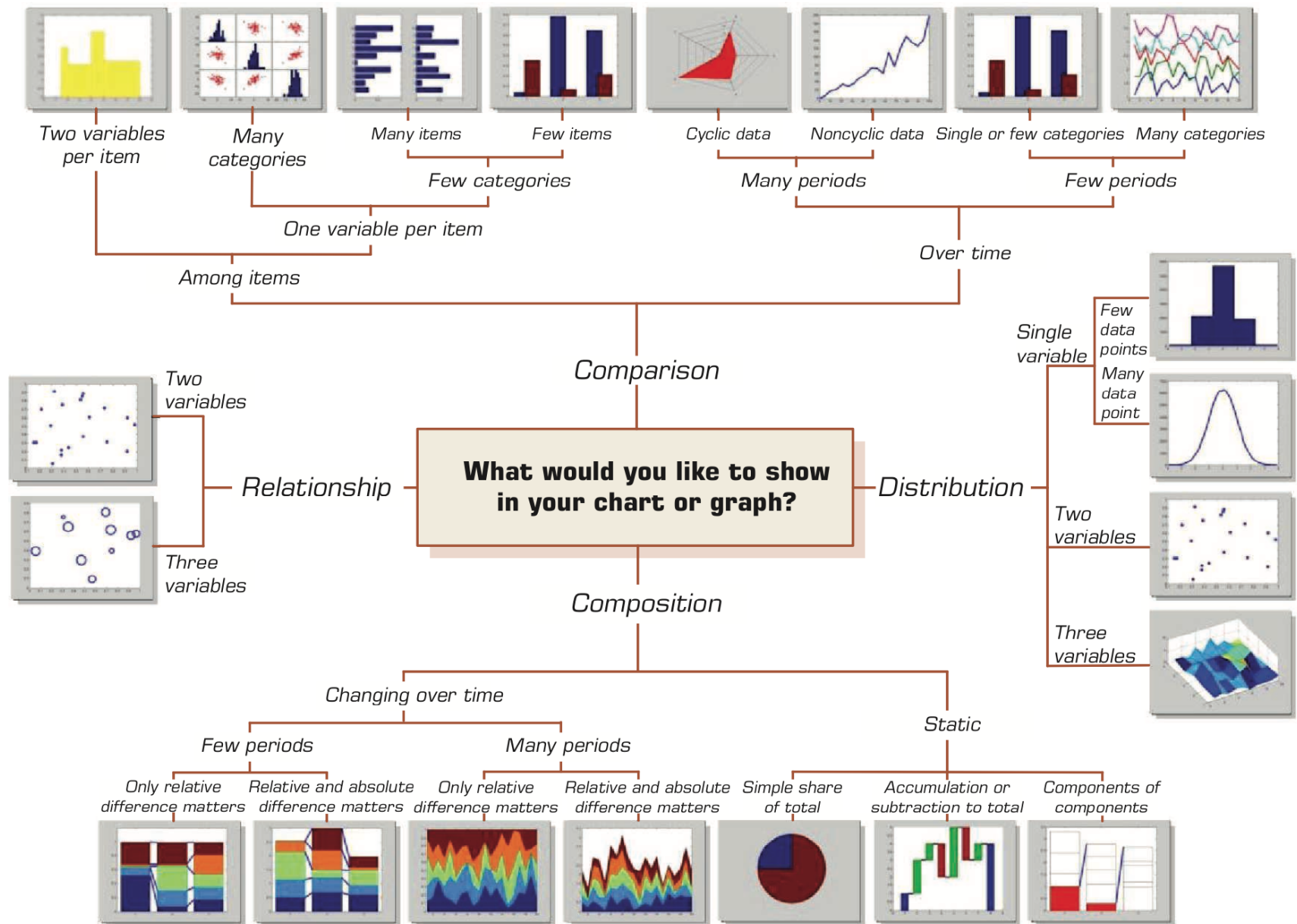
# A Sample Time Series of Data on Quarterly Sales Volumes



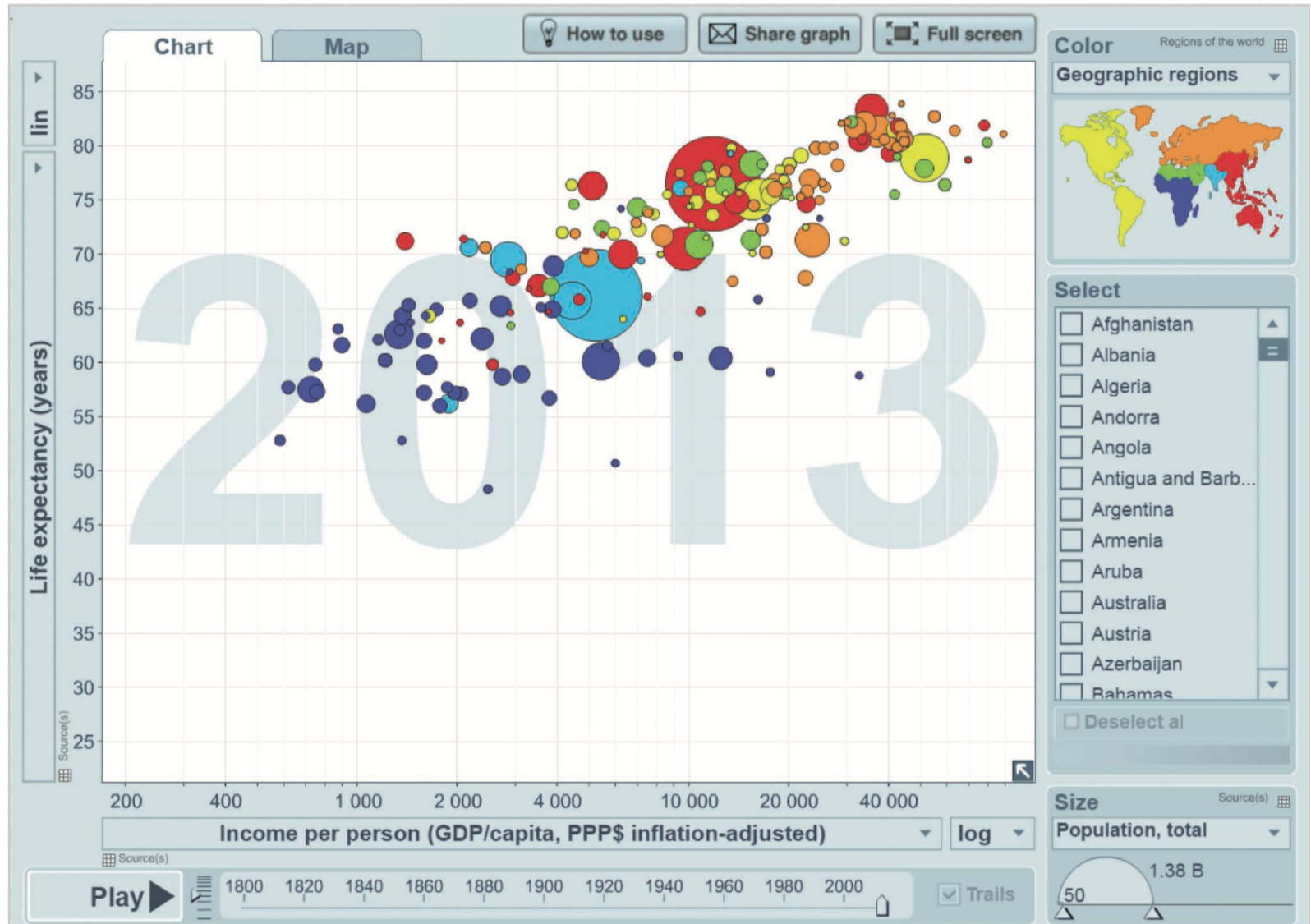
# The Role of Information Reporting in Managerial Decision Making



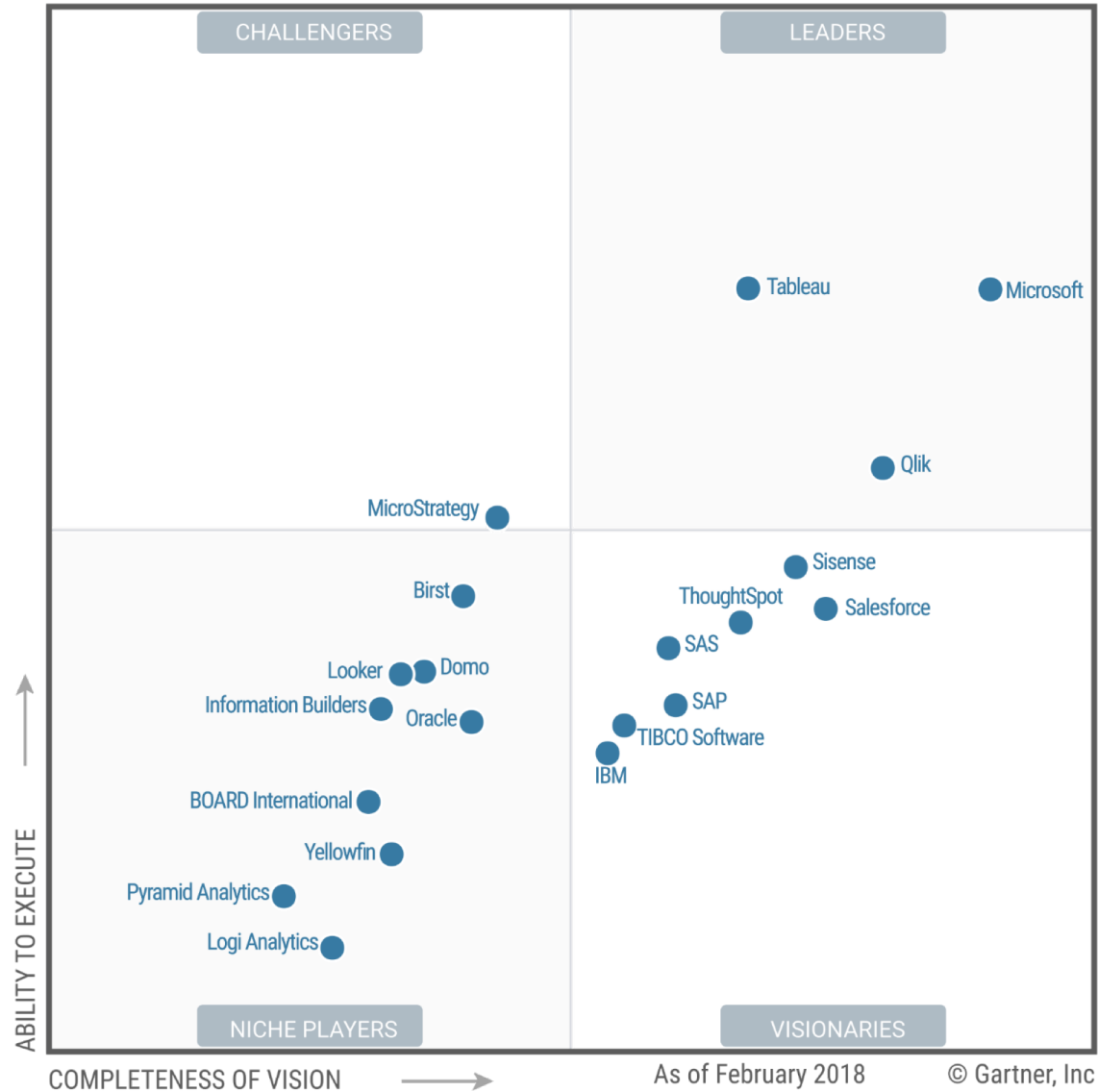
# A Taxonomy of Charts and Graphs



# A Gapminder Chart That Shows the Wealth and Health of Nations

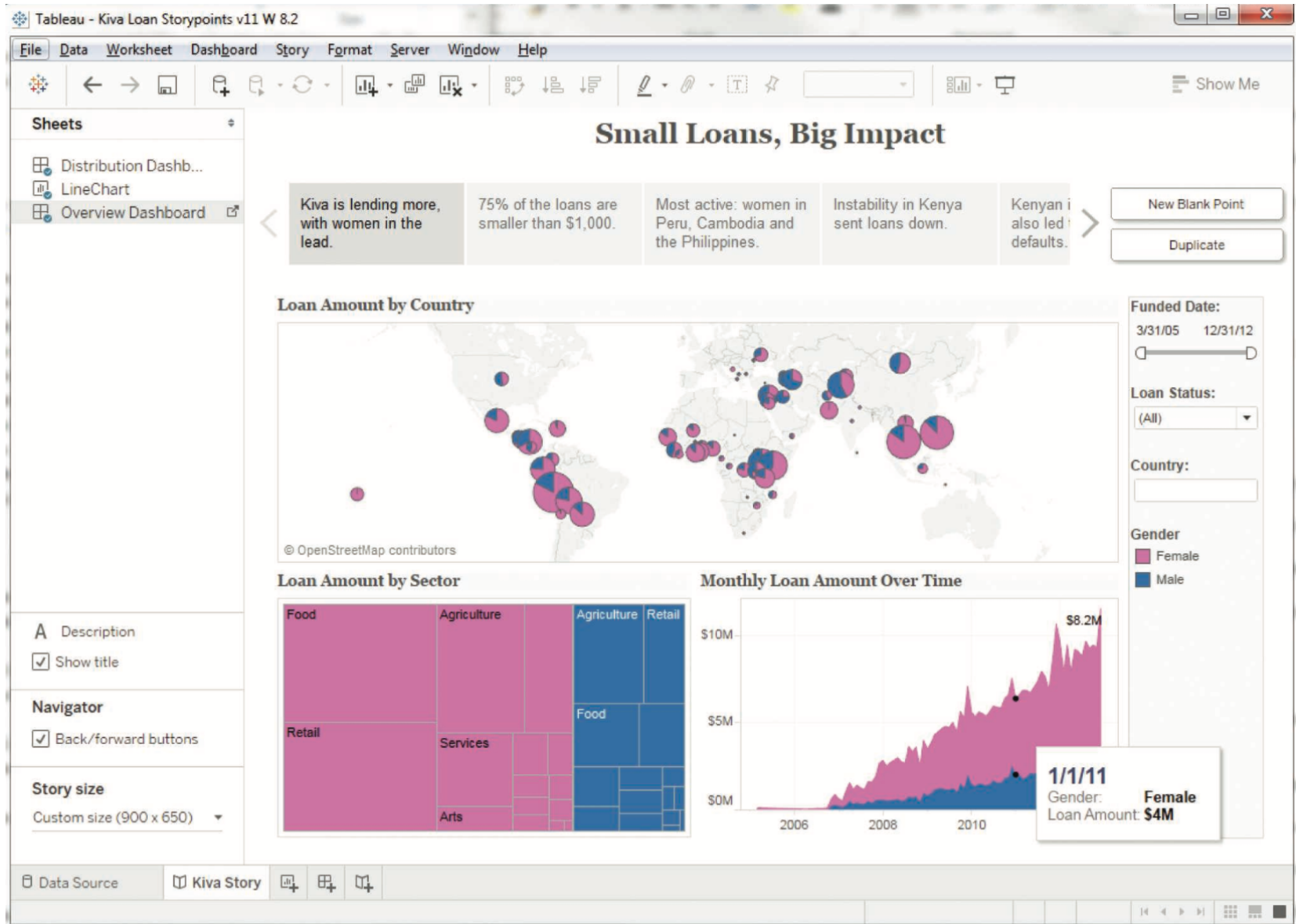


# Magic Quadrant for Business Intelligence and Analytics Platforms



Source: <https://www.tableau.com/reports/gartner>

# A Storyline Visualization in Tableau Software



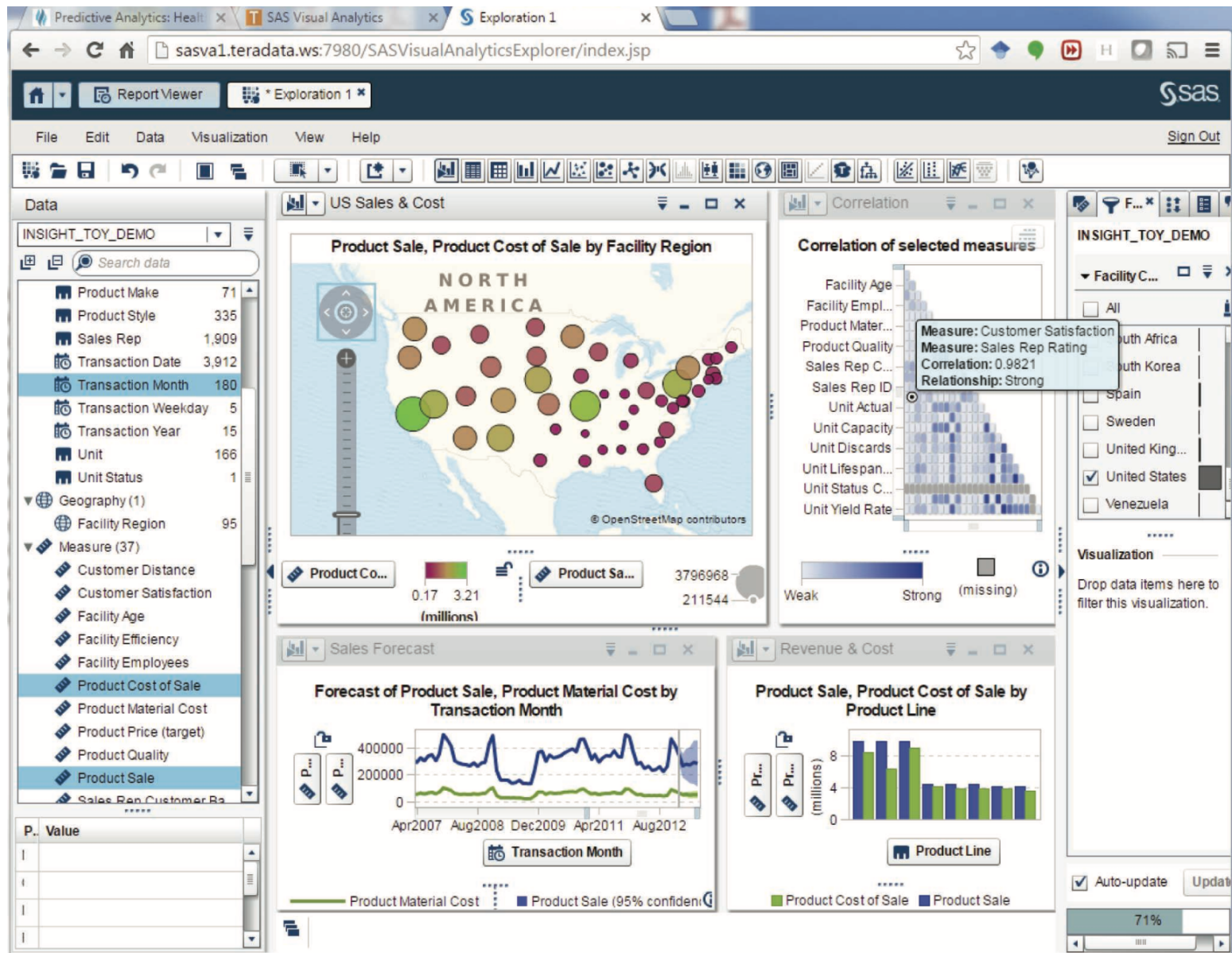


# An Overview of SAS Visual Analytics Architecture

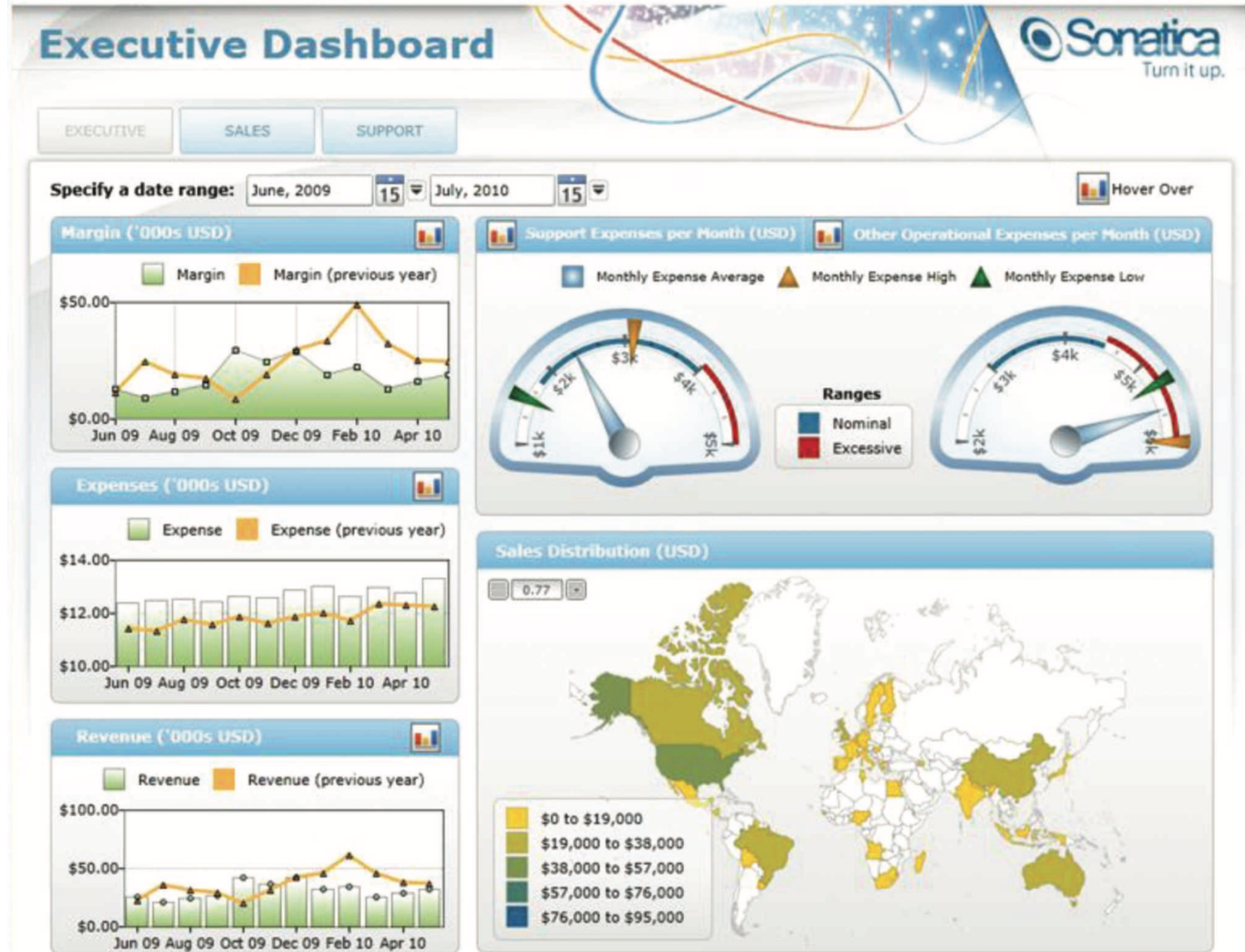




# A Screenshot from SAS Visual Analytics



# A Sample Executive Dashboard



# Big Data



**Mobile  
Sensors**



**Social  
Media**



**Video  
Surveillance**



**Video  
Rendering**



**Smart  
Grids**



**Geophysical  
Exploration**

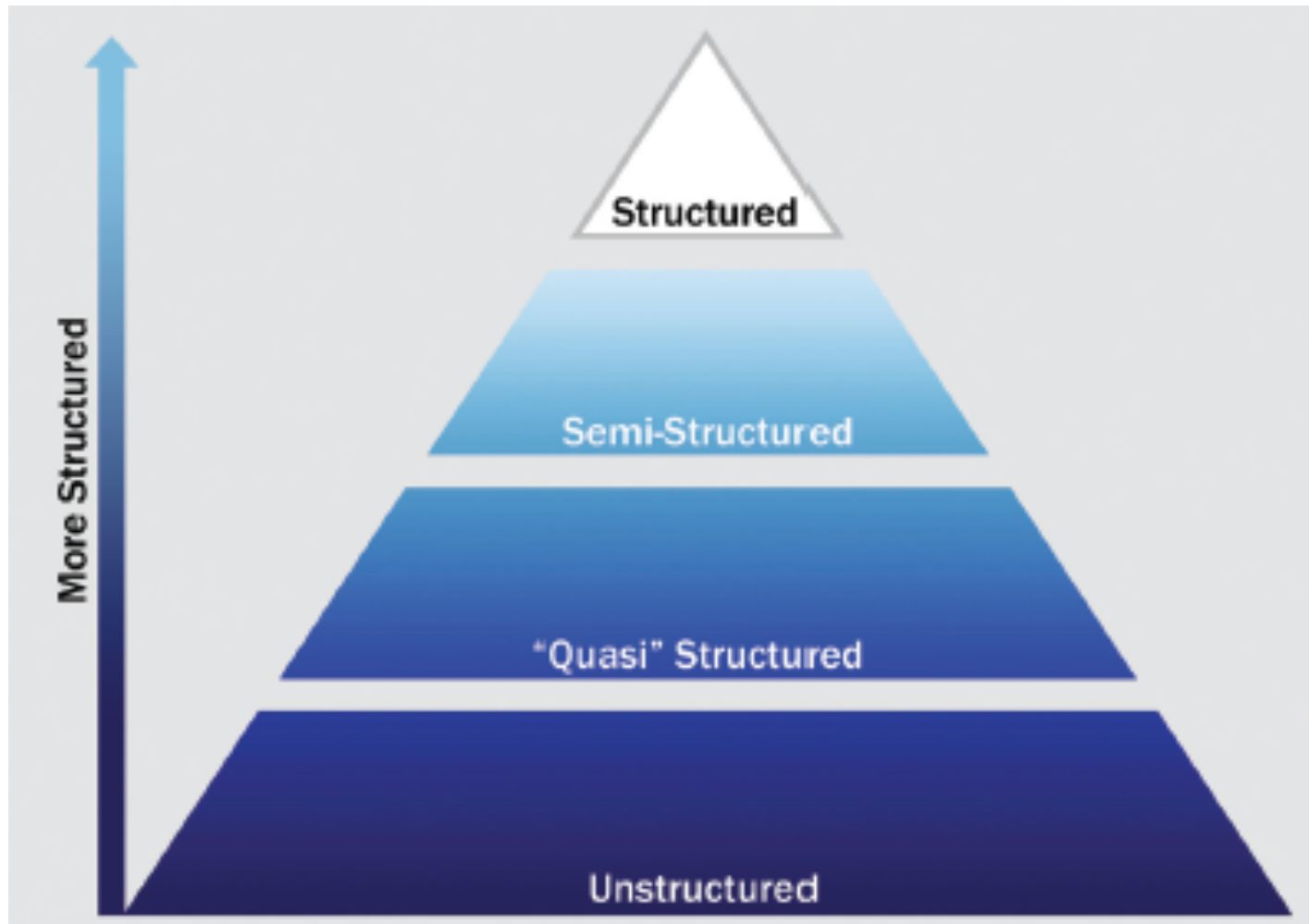


**Medical  
Imaging**

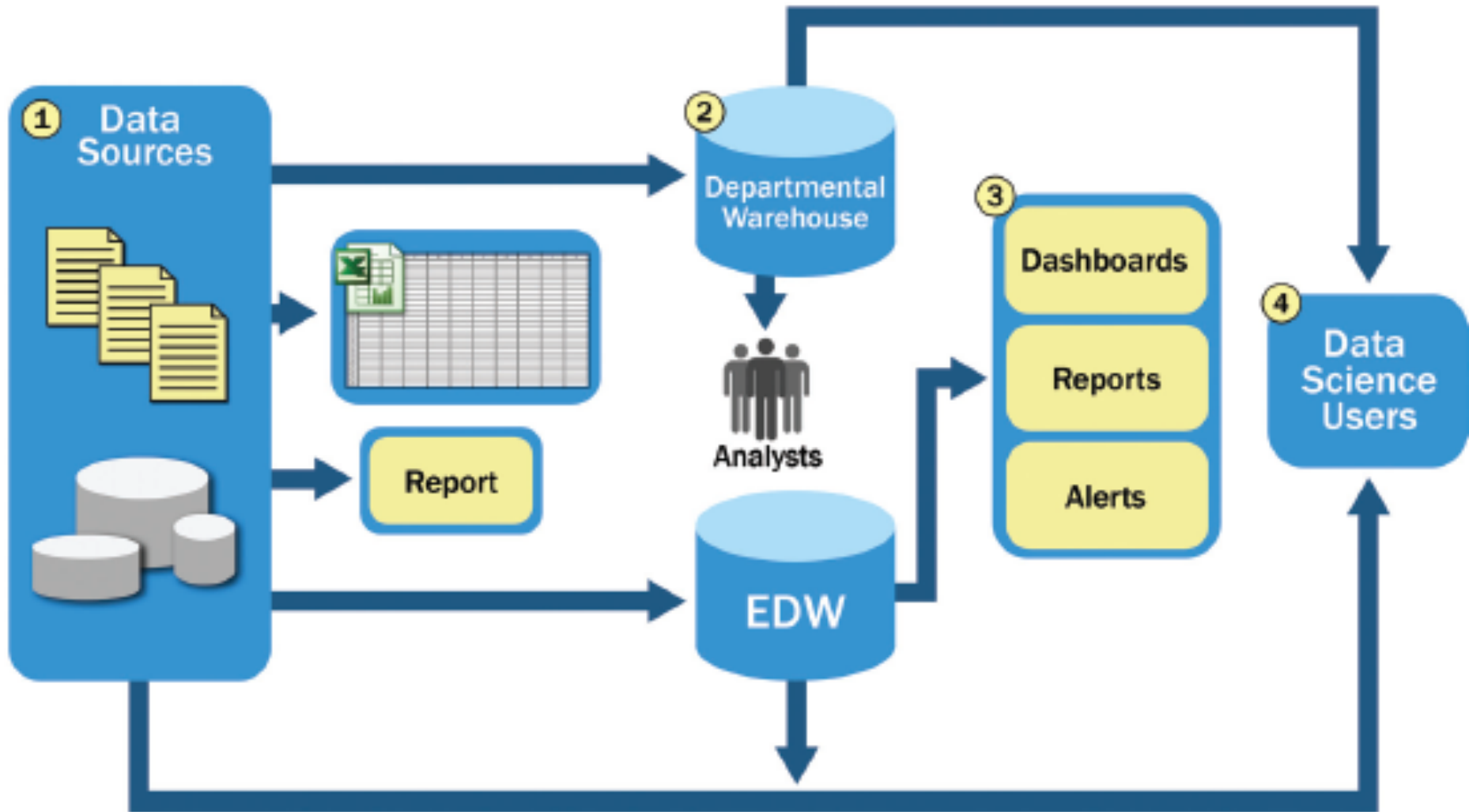


**Gene  
Sequencing**

# Big Data Growth is increasingly **unstructured**

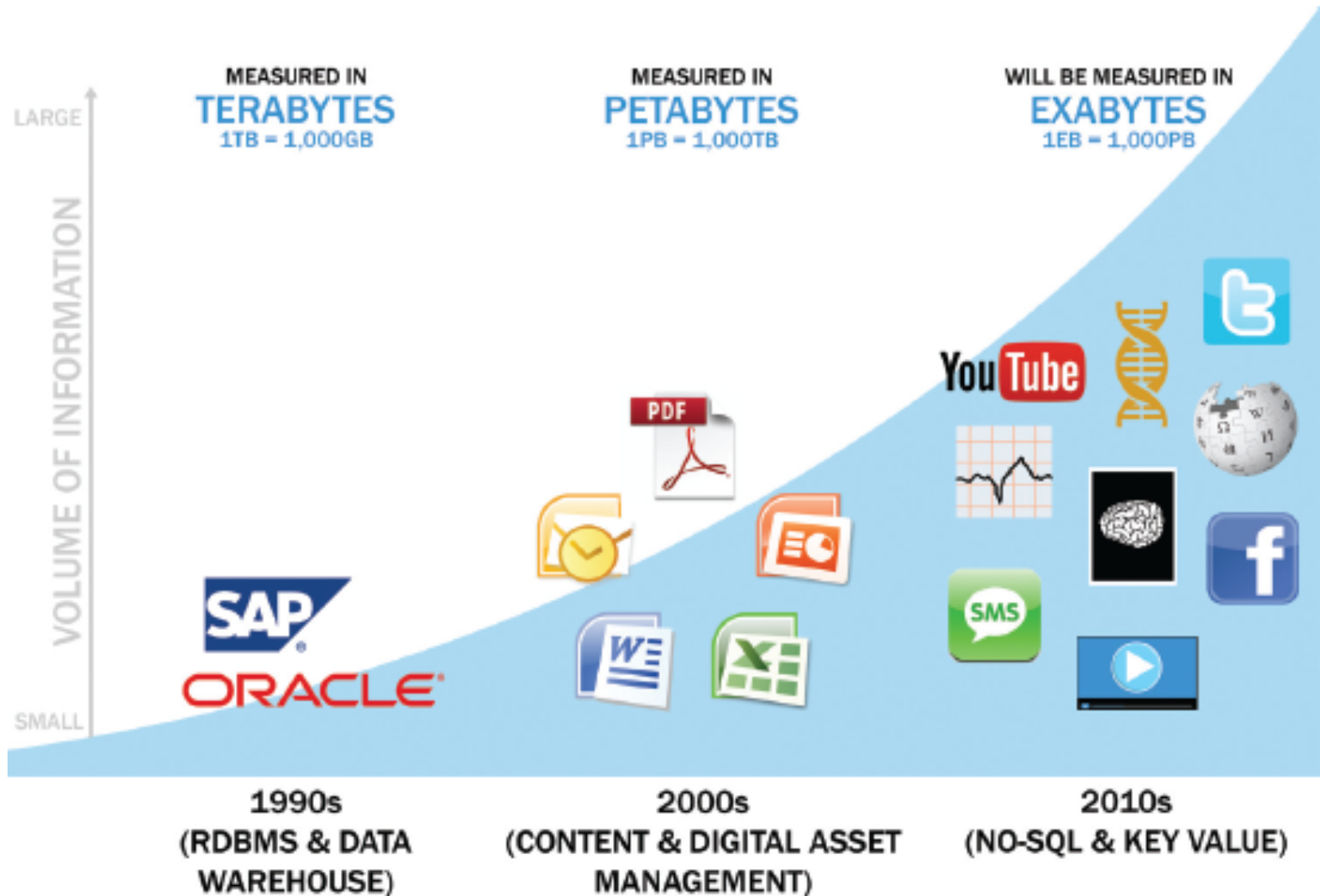


# Typical Analytic Architecture





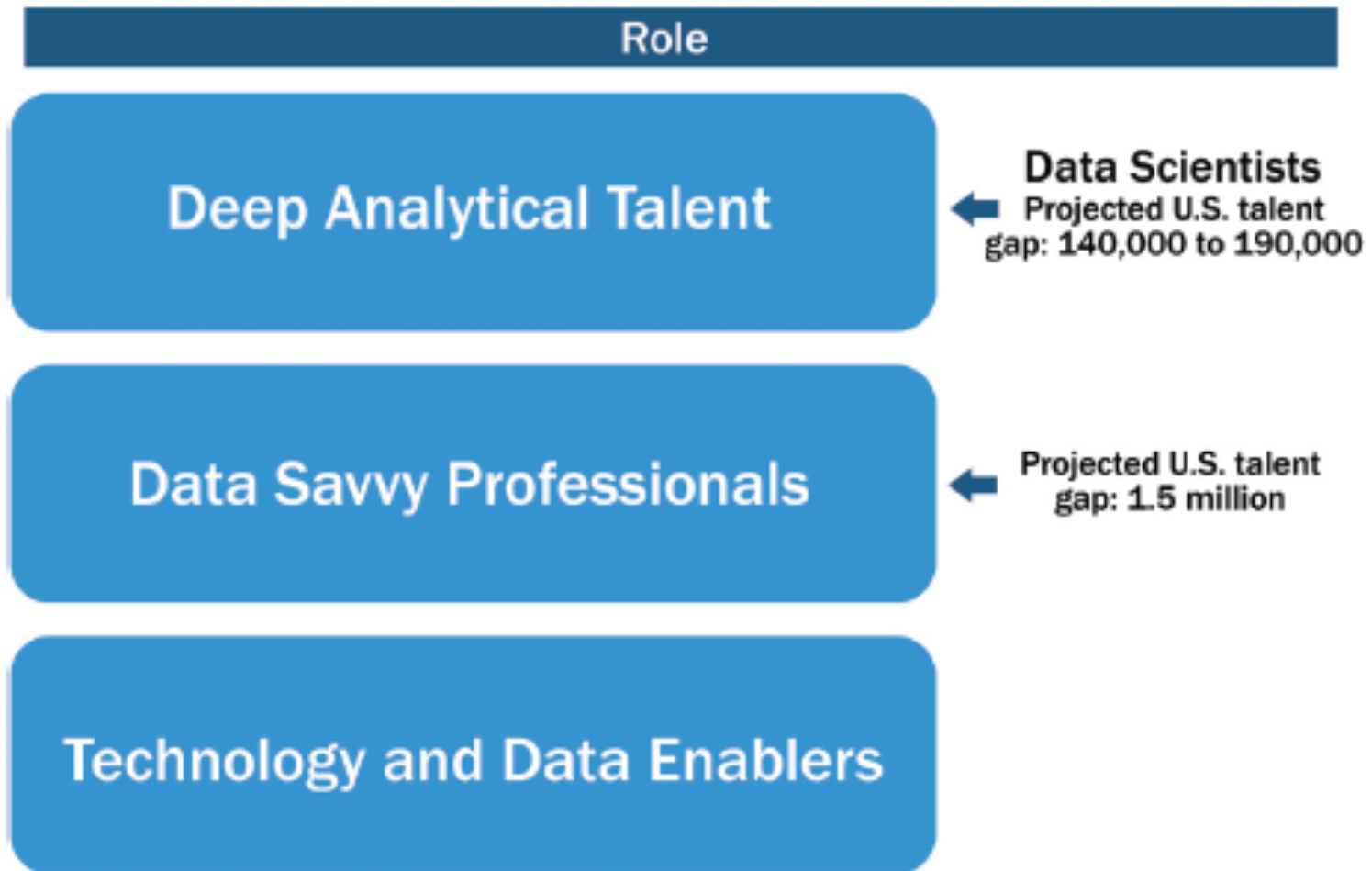
# Data Evolution and the Rise of Big Data Sources



# Emerging Big Data Ecosystem



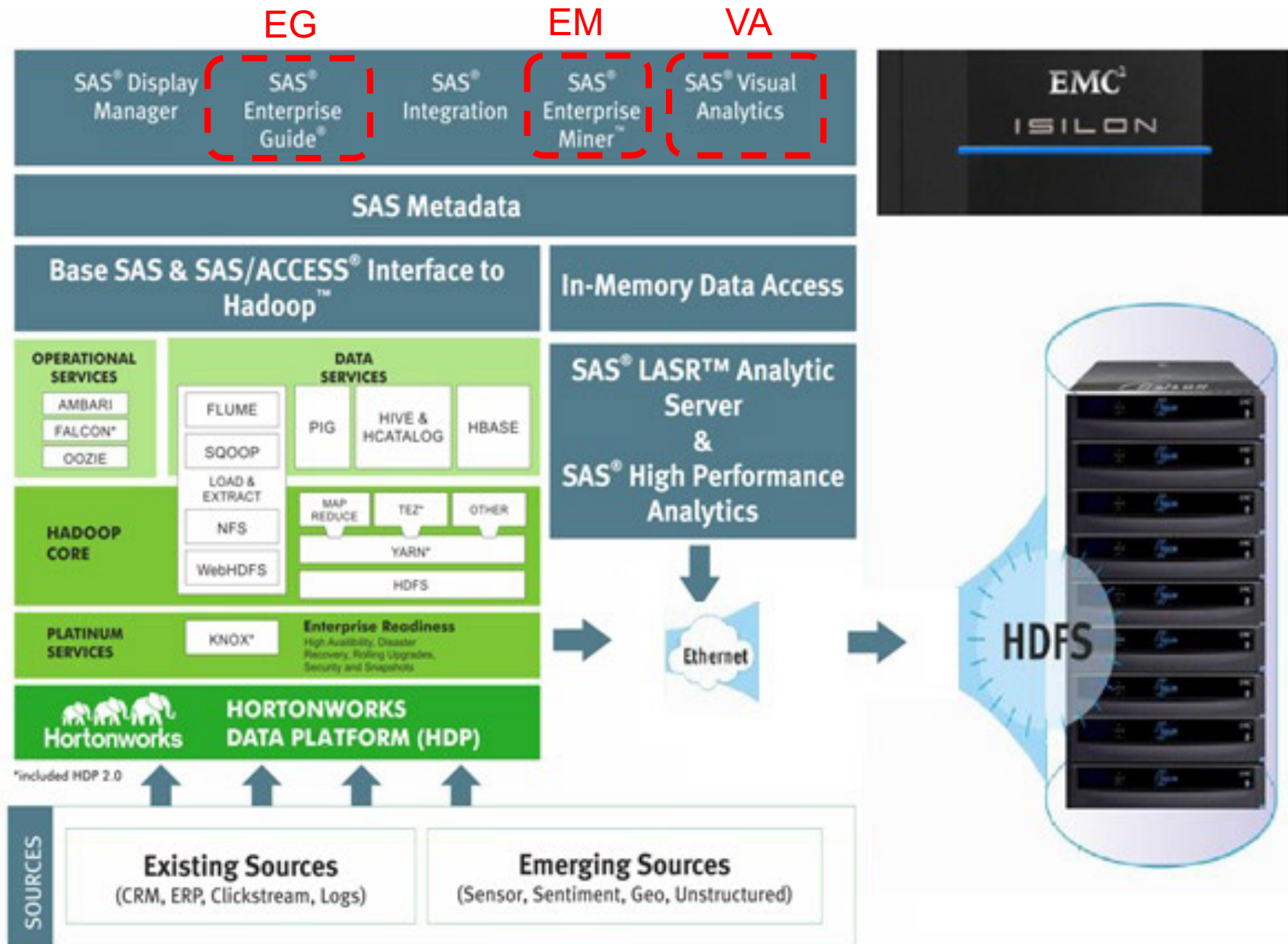
# Key Roles for the New Big Data Ecosystem



Note: Figures above reflect a projected talent gap in US in 2018, as shown in McKinsey May 2011 article "Big Data: The Next Frontier for Innovation, Competition, and Productivity"



# Big Data Solution



National  
Security

Cyber  
security

Maritime  
security

Smarter  
Transport

...

## VISUAL ANALYTICS

DYNAMIC & INTERACTIVE

Dashboard Graph  
Map

ENHANCE

Understanding Investigation  
User Experience



## BIG ANALYTICS

QUERY & FILTER

Complex queries  
 $R^2I^2$

DETECT

Anomalies  
Communities  
Typologies

PREDICT

Trending  
Real-time  
Prediction

DECIDE

Simulation  
Optimization



## BIG DATA – Batch



## BIG DATA – Real Time

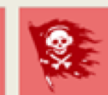


Complex by nature



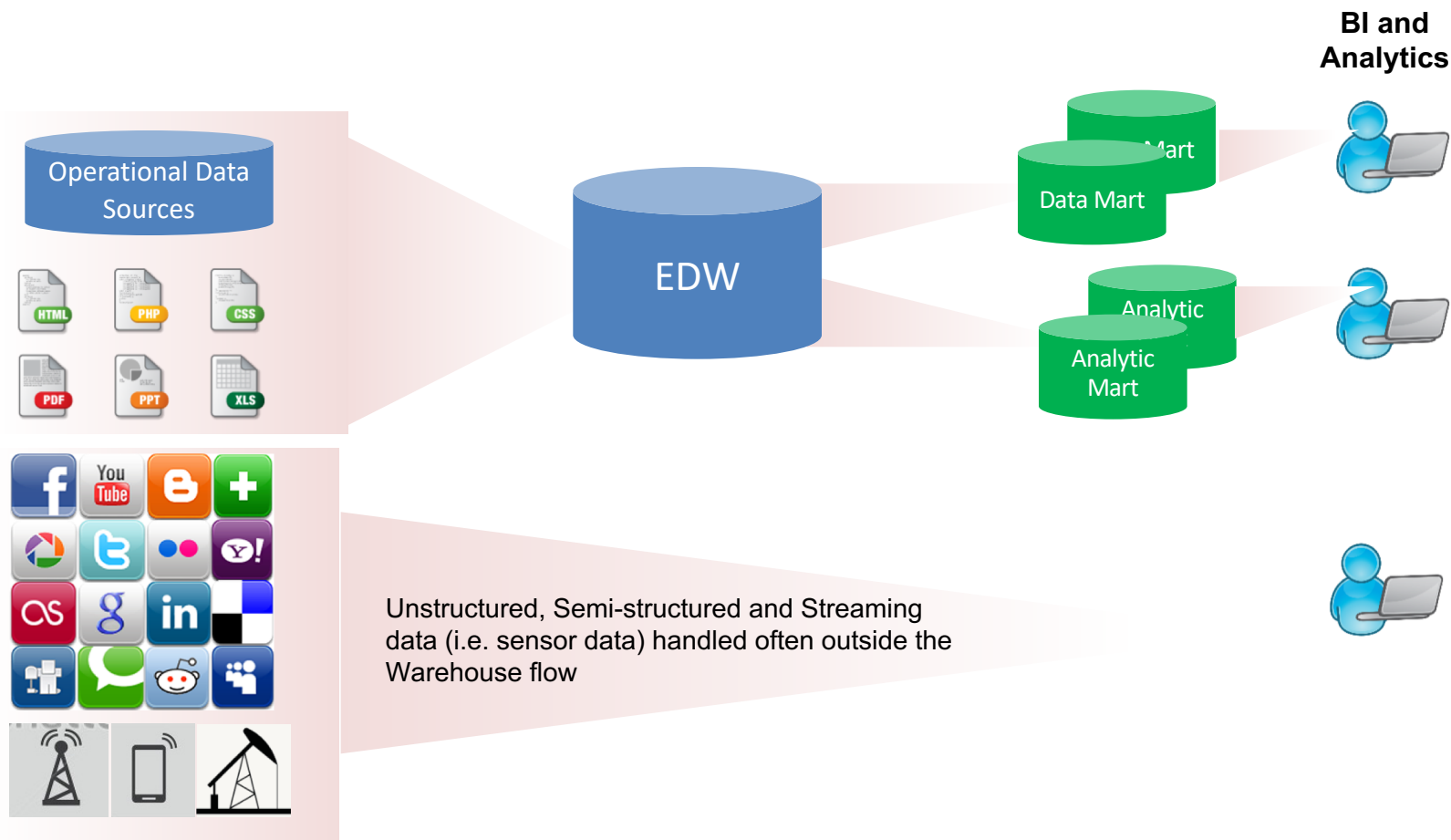
# DATA

Complex by structure

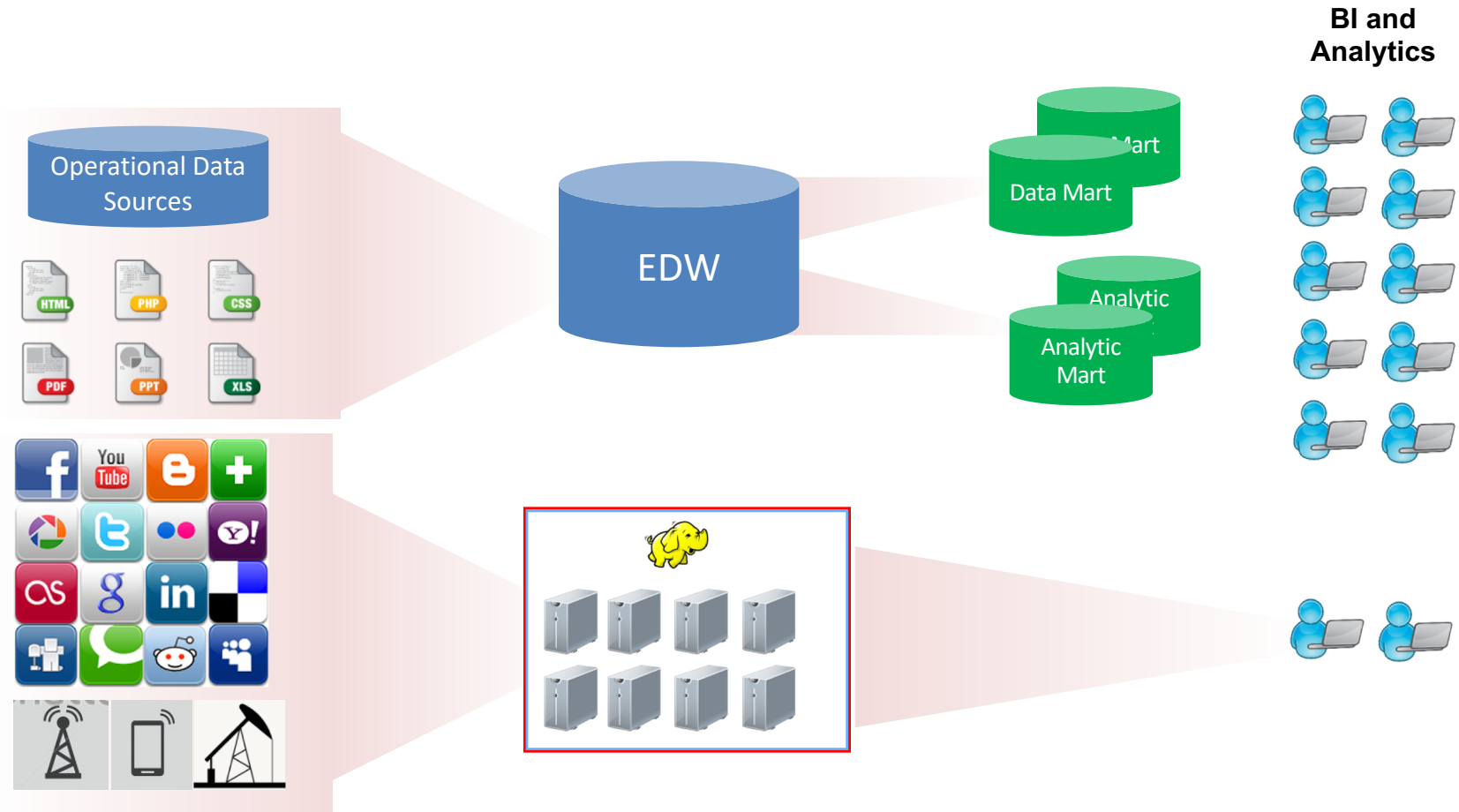


# Architectures of Big Data Analytics

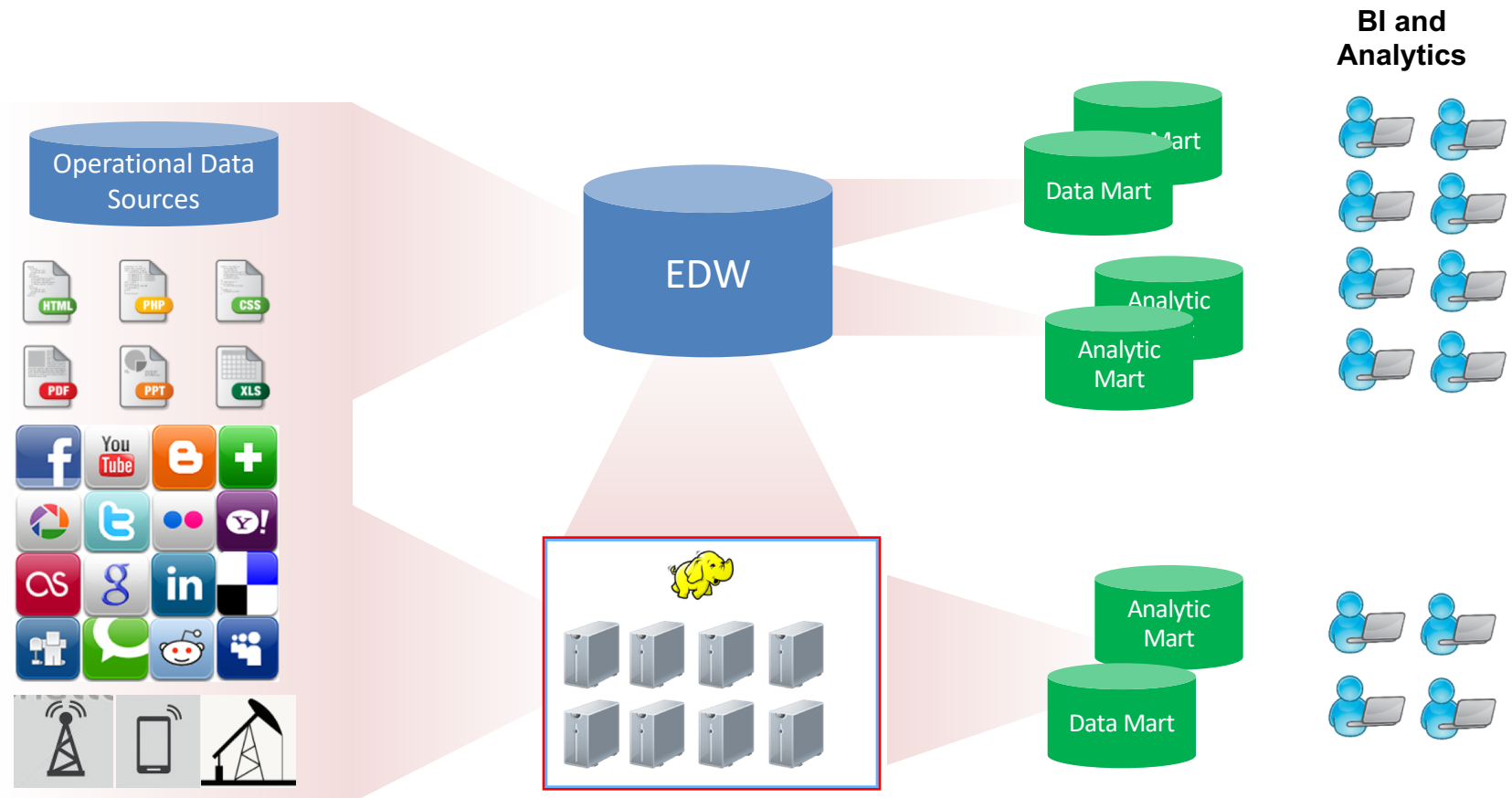
# Traditional Analytics



# Hadoop as a “new data” Store

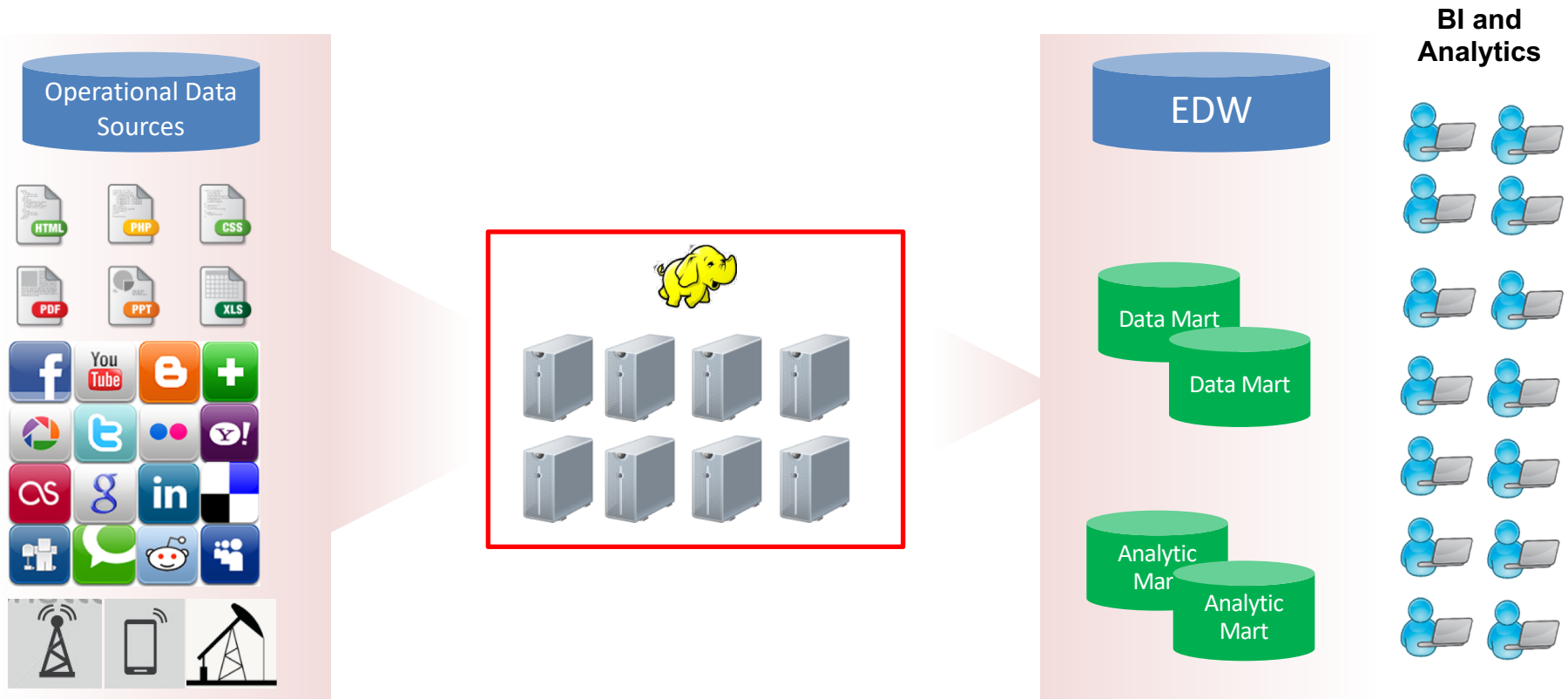


# Hadoop as an additional input to the EDW



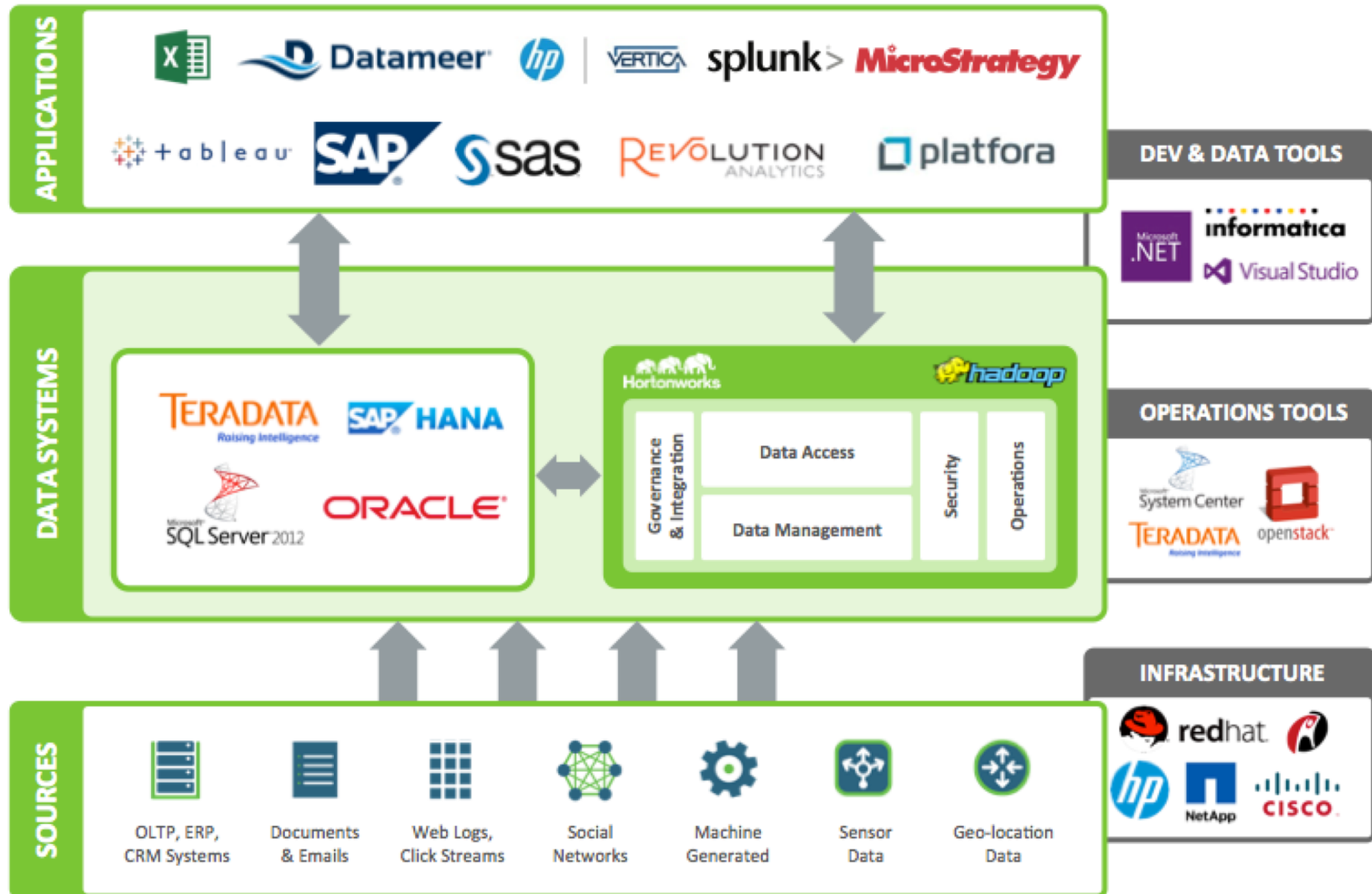
# Hadoop Data Platform As a “staging Layer” as part of a “data Lake”

– Downstream stores could be Hadoop, data appliances or an RDBMS



# SAS Big data Strategy

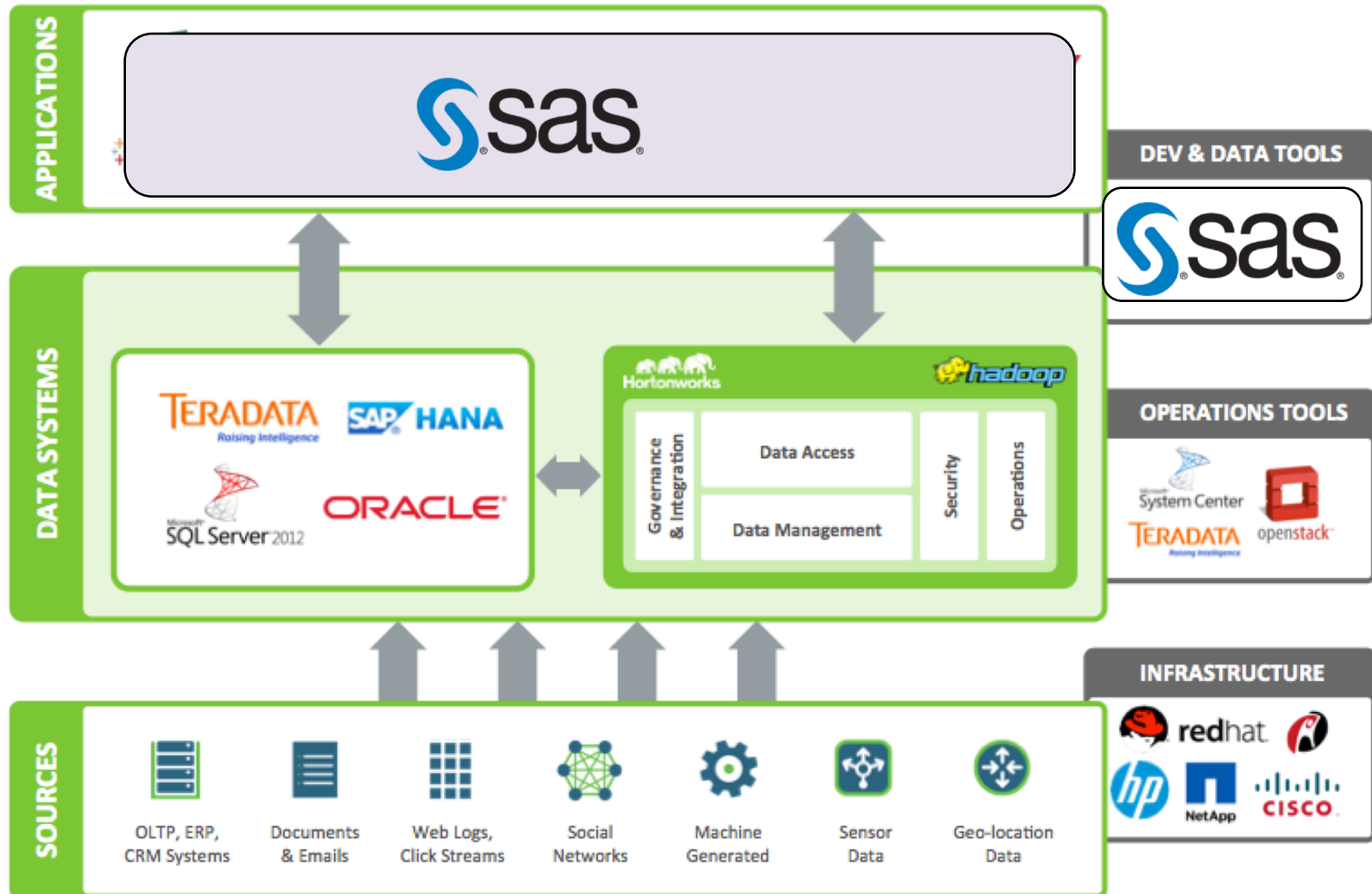
## – SAS areas



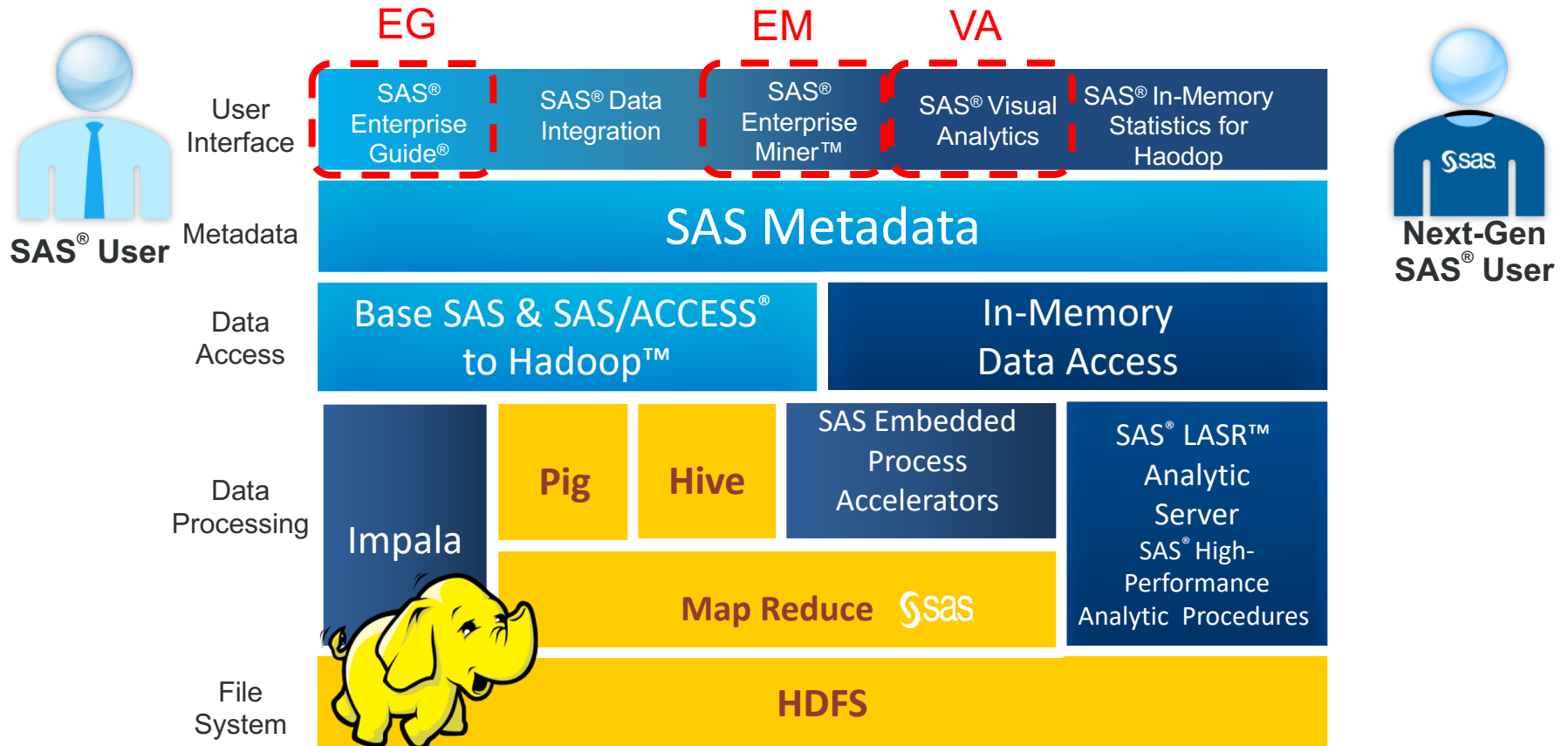


# SAS Big data Strategy

## – SAS areas

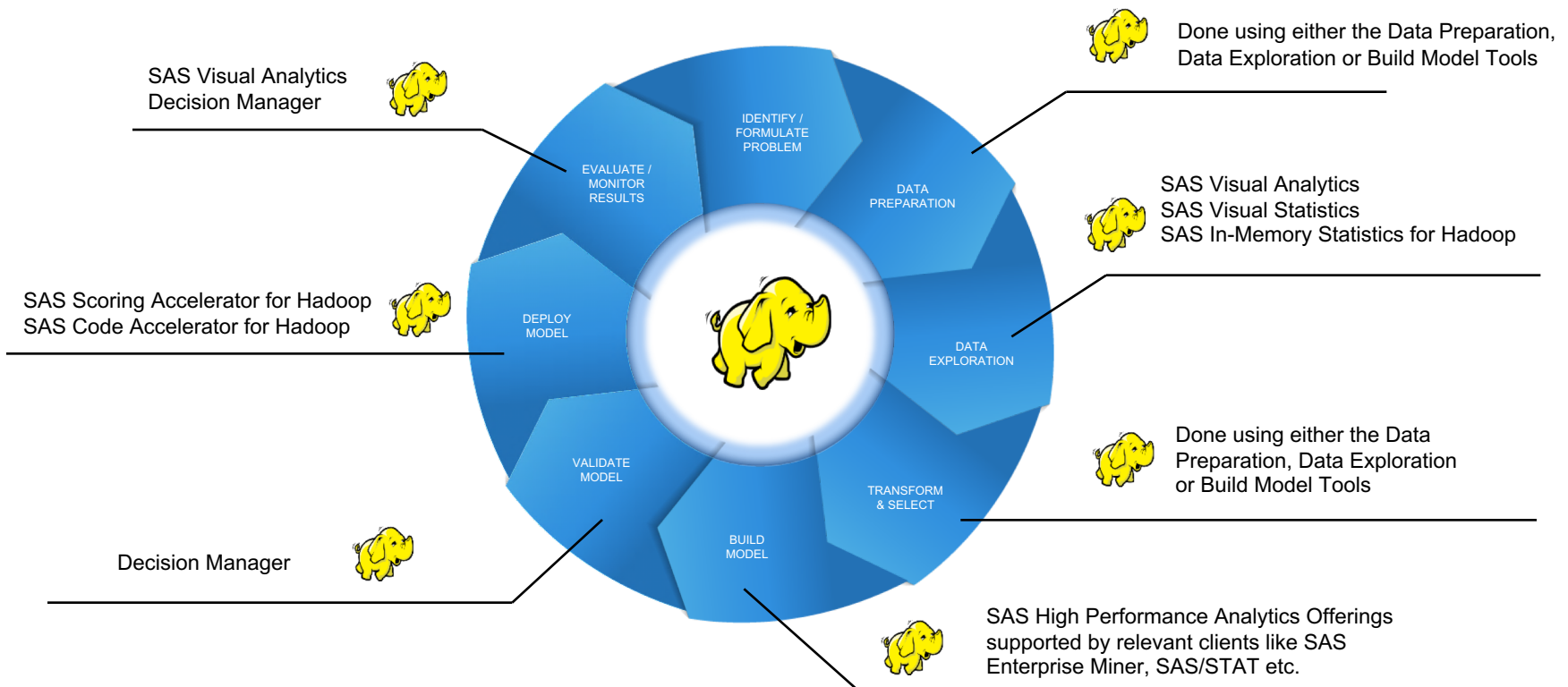


# SAS® Within the HADOOP ECOSYSTEM



# SAS enables the entire lifecycle around HADOOP

SAS enableS the entire lifecycle around HADOOP



# **SAS® VISUAL ANALYTICS**

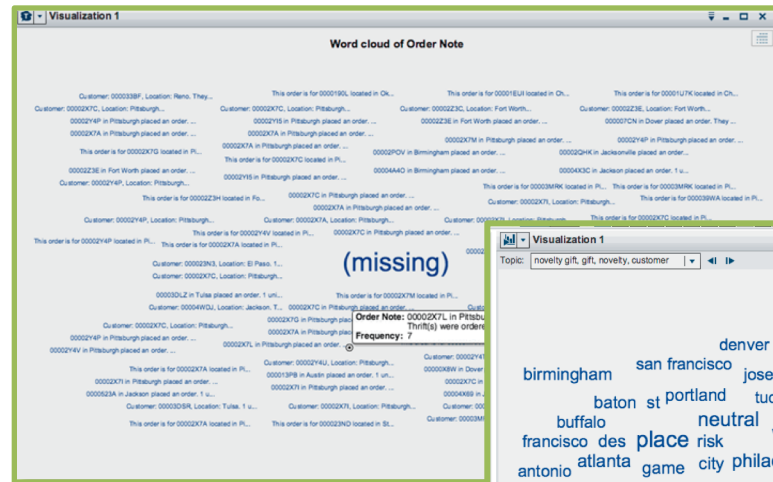
**A Single solution for  
Data Discovery,  
Visualization, analytics and  
reporting**

# SAS® VISUAL ANALYTICS

Example: text analysis gives you insight to customer experience and opinion



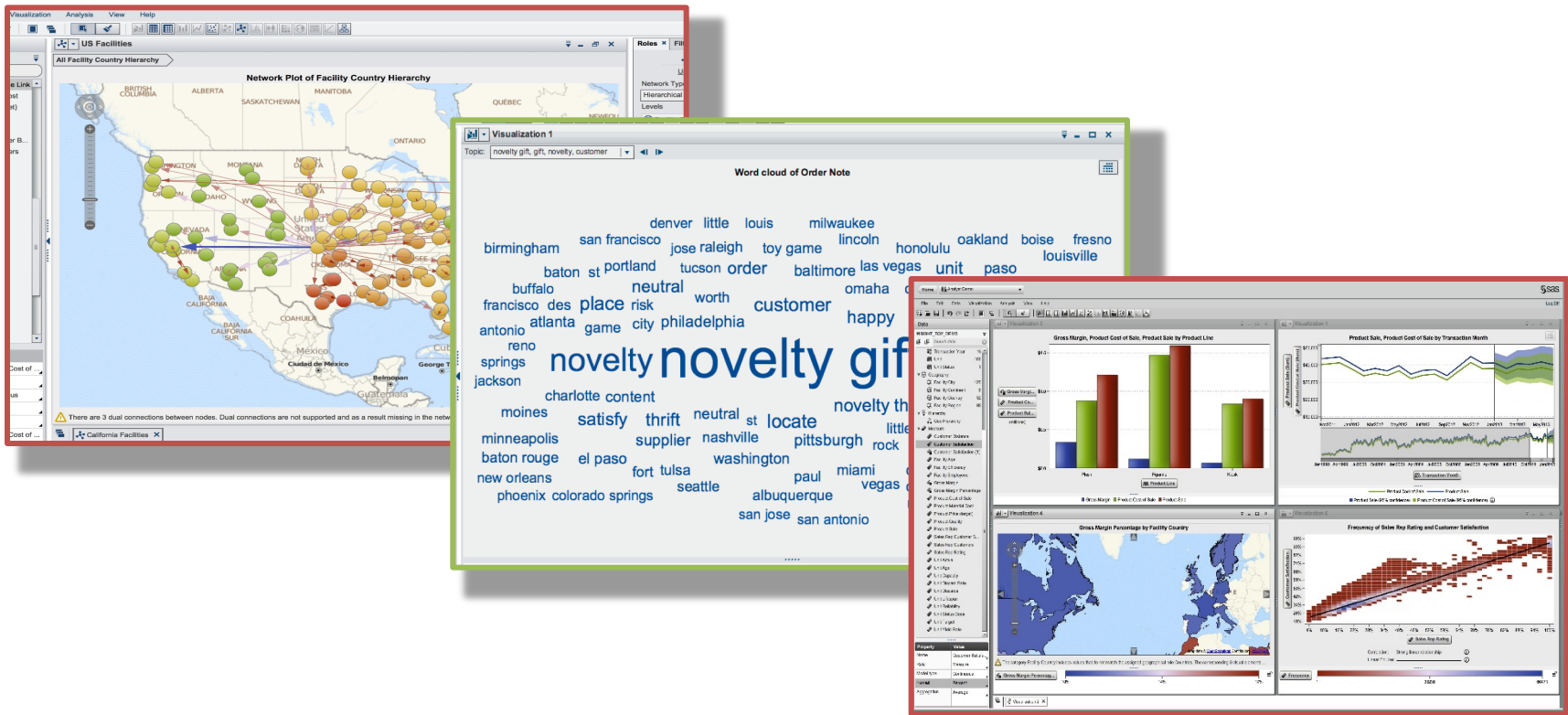
ANALYTICS POWERED



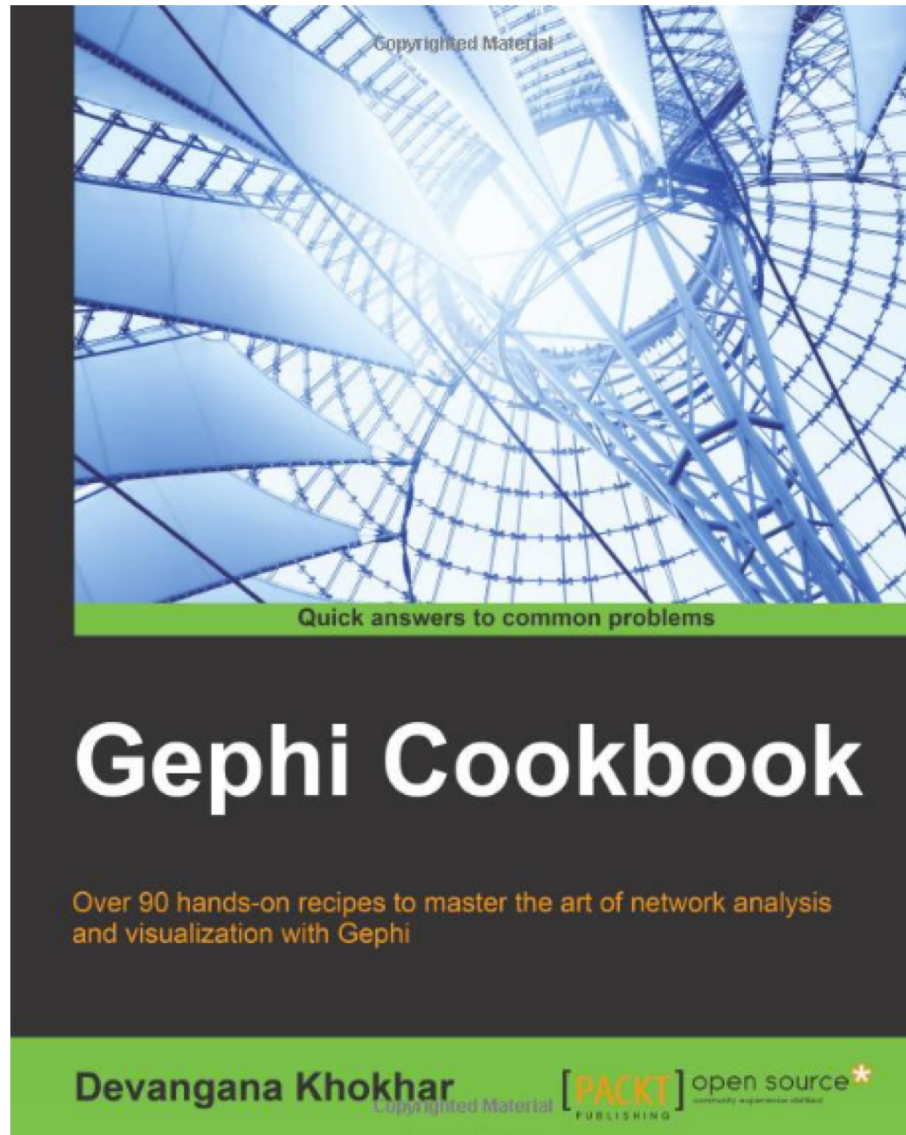
Analytics applied  
to text provides  
real MEANING



# Visualization



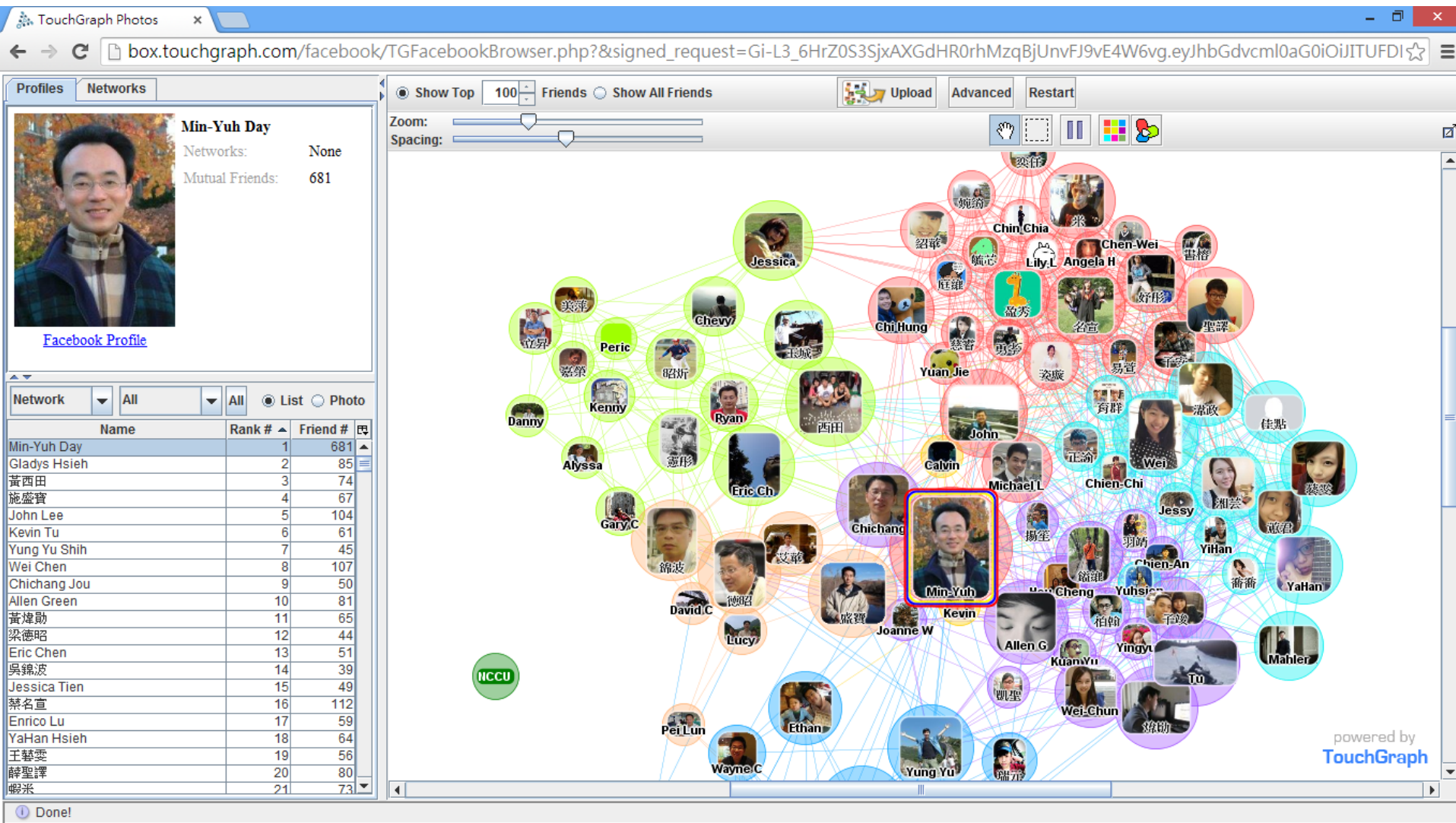
Devangana Khokhar (2015),  
**Gephi Cookbook**, Packt Publishing





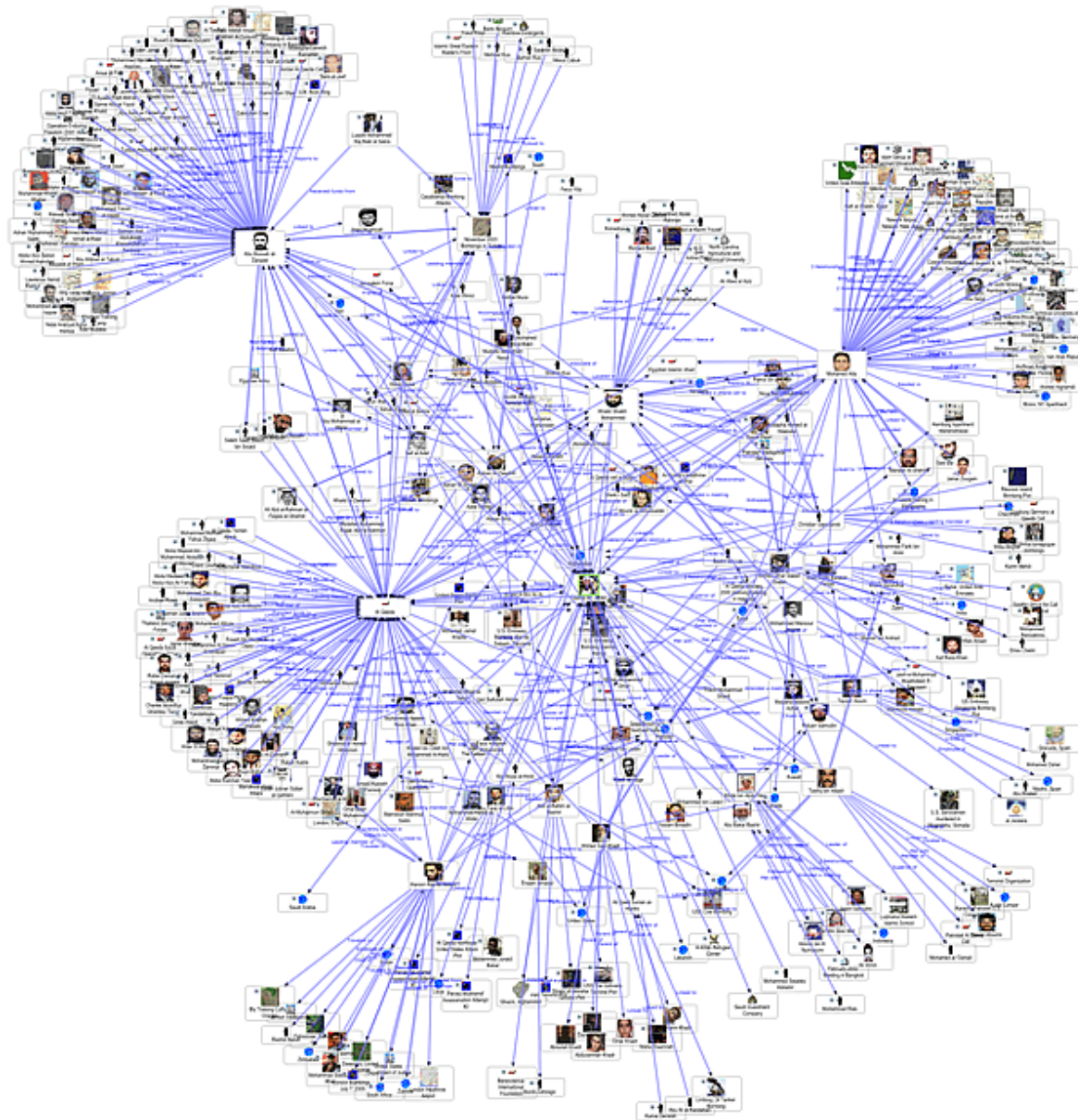
# Social Network Analysis (SNA)

## Facebook TouchGraph

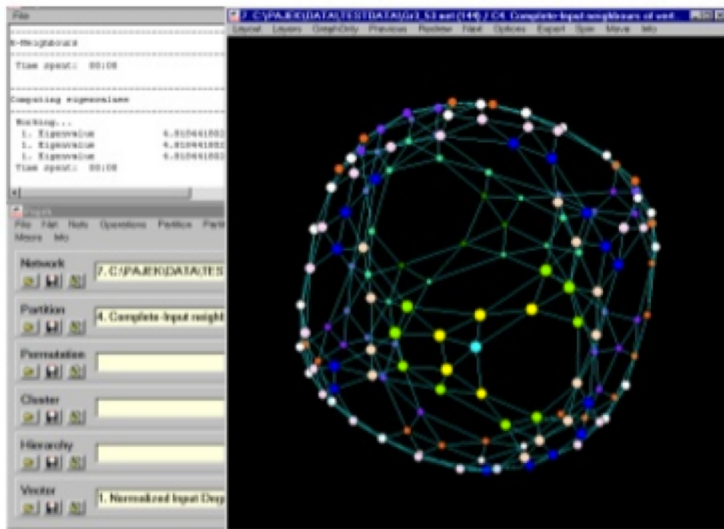




# Social Network Analysis

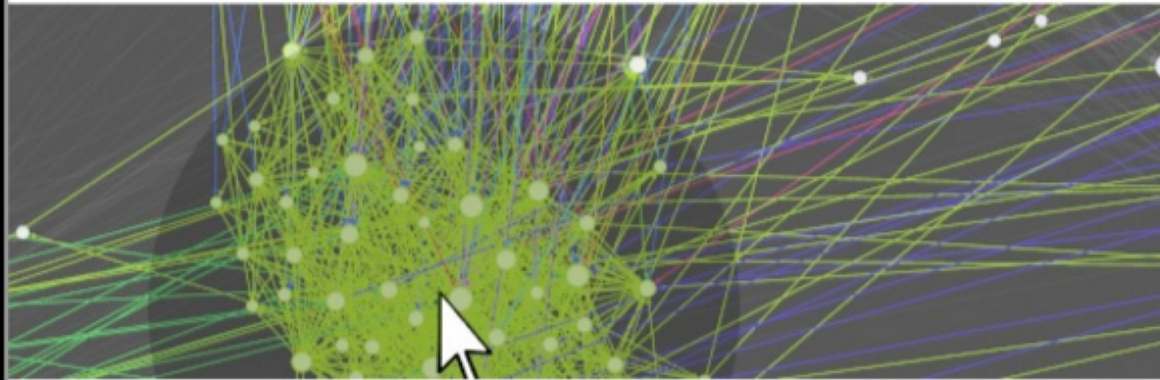


# Exploratory Network Analysis



## 1 see the network

1st graph viz tool: Pajek (1996)  
Vladimir Batagelj, Andrej Mrvar

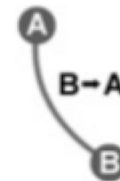
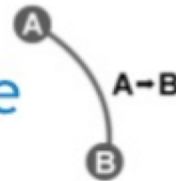


## 2 interact in real time

Gephi prototype (2008)  
group, filter, compute metrics...

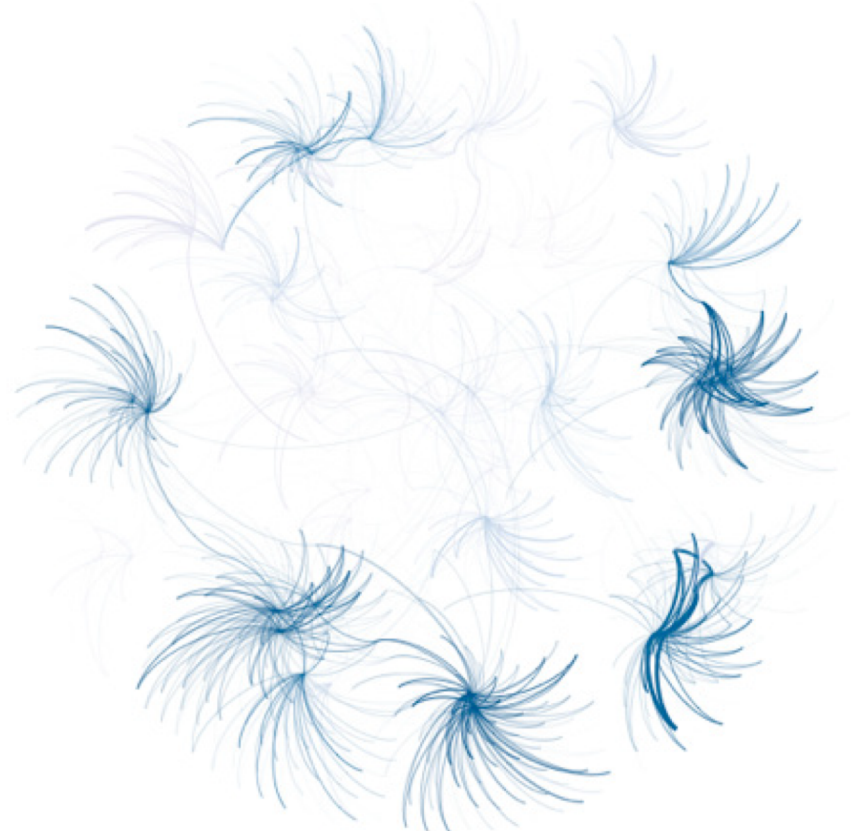
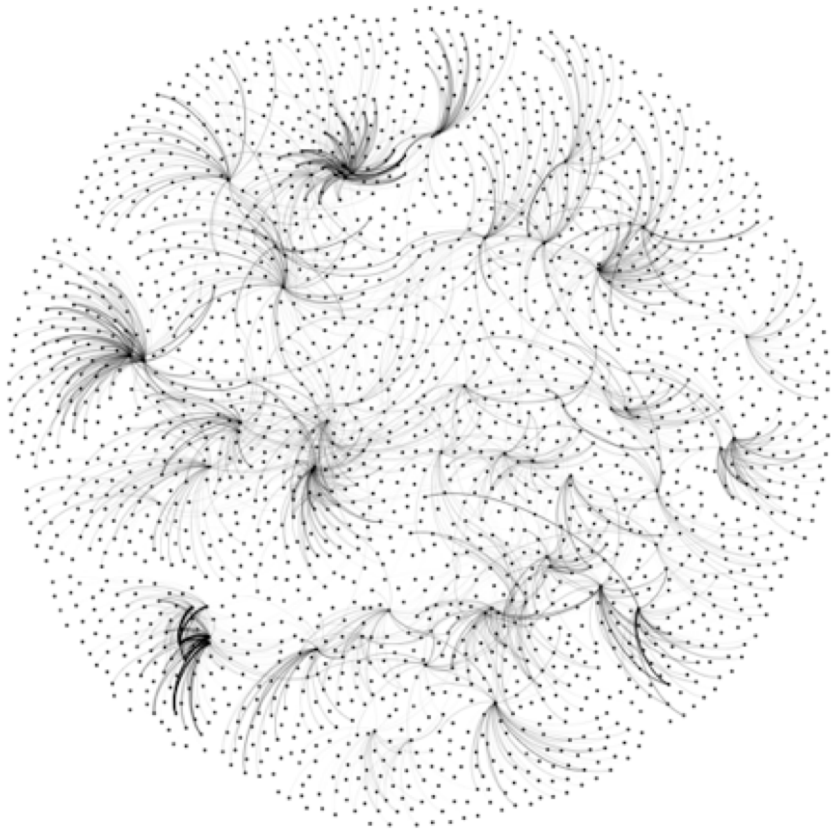
## 3 build a visual language

size by rank, color by partition,  
label, curved edges, thickness...



# Looking for a “Simple Small Truth”?

## What Data Visualization Should Do?



1. Make complex things **simple**
2. Extract **small** information from large data
3. Present **truth**, do not deceive



# igraph



Products ▾

News

On github



## igraph – The network analysis package

igraph is a collection of network analysis tools with the emphasis on **efficiency**, **portability** and ease of use. igraph is **open source** and free. igraph can be programmed in **R**, **Python** and **C/C++**.

igraph R package

python-igraph

igraph C library

R/igraph 1.0.0

Repositories at Github

R/igraph 0.7.1

C/igraph 0.7.1

R/igraph 0.7.0

python-igraph 0.7.0

C/igraph 0.7.0

R/igraph 0.6.5

## Recent news

### R/igraph 1.0.0

June 24, 2015

### Release Notes

This is a new major release, with a lot of UI changes. We tried to make it easier to use, with short and easy to remember, consistent function names. Unfortunately

<http://igraph.org/redirect.html>

# Gephi



[Download](#) [Blog](#) [Wiki](#) [Forum](#) [Support](#) [Bug tracker](#)

[Home](#) [Features](#) [Learn](#) [Develop](#) [Plugins](#) [Services](#) [Consortium](#)

## The Open Graph Viz Platform

**Gephi is the leading visualization and exploration software for all kinds of graphs and networks. Gephi is open-source and free.**

**Runs on Windows, Mac OS X and Linux.**

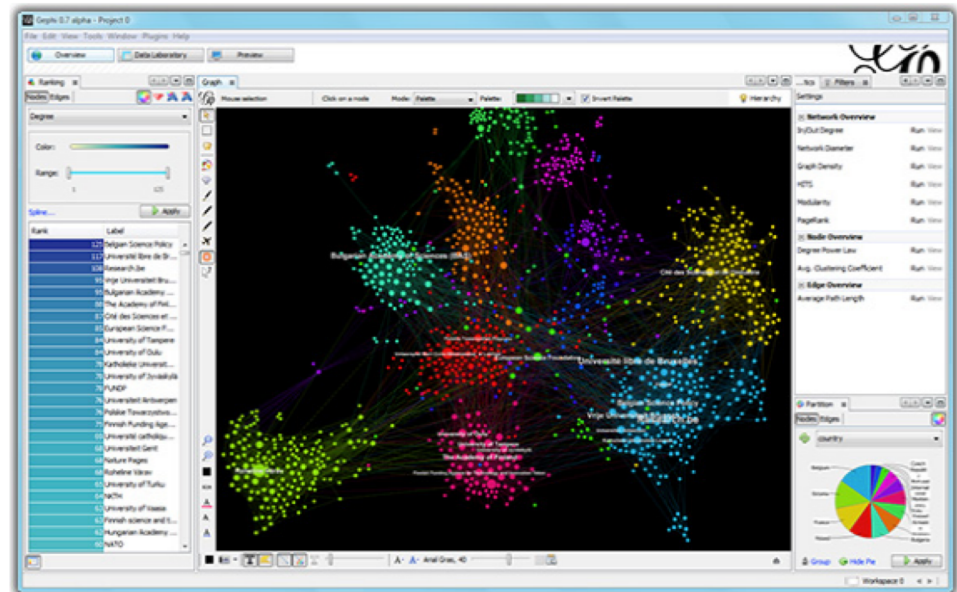
[Learn More on Gephi Platform »](#)



[Release Notes](#) | [System Requirements](#)

► **Features**  
► **Quick start**

► **Screenshots**  
► **Videos**



**Support us! We are non-profit. Help us to innovate and empower the community by donating only 8€:**

[Donate](#)



### APPLICATIONS

- ✓ **Exploratory Data Analysis:** intuition-oriented analysis by networks manipulations in real time.
- ✓ **Link Analysis:** revealing the underlying structures of associations between objects.
- ✓ **Social Network Analysis:** easy creation of social

**Like Photoshop™ for graphs.**

— the Community

### LATEST NEWS

► [Gephi updates with 0.9.1 version](#)

### PAPERS



<https://gephi.org/>

# **Discovering, Analyzing, Visualizing and Presenting Data with Python in Google Colab**

# Google Colab

The screenshot shows the Google Colaboratory interface in a web browser. The browser's address bar displays the URL <https://colab.research.google.com/notebooks/welcome.ipynb>. The page header includes the Colab logo, the text "Hello, Colaboratory", and a menu with options: File, Edit, View, Insert, Runtime, Tools, and Help. On the right side of the header, there is a "SHARE" button and a user profile icon. Below the header, a toolbar contains icons for "CODE", "TEXT", "CELL" (up and down arrows), and "COPY TO DRIVE". To the right of the toolbar are "CONNECT" and "EDITING" buttons. A left sidebar shows a "Table of contents" with links to "Getting Started", "Highlighted Features", "TensorFlow execution", "GitHub", "Visualization", "Forms", "Examples", and "Local runtime support". The main content area features a "Welcome to Colaboratory!" message with the Colab logo and a brief description. Below this is a "Getting Started" section with a list of links. Further down is a "Highlighted Features" section with a "Seedbank" subsection and a "TensorFlow execution" subsection. The TensorFlow section includes a text description and a matrix addition example.

Table of contents

- Getting Started
- Highlighted Features
  - TensorFlow execution
- GitHub
- Visualization
- Forms
- Examples
- Local runtime support

SECTION

## Welcome to Colaboratory!

Colaboratory is a free Jupyter notebook environment that requires no setup and runs entirely in the cloud. See our [FAQ](#) for more info.

### Getting Started

- [Overview of Colaboratory](#)
- [Loading and saving data: Local files, Drive, Sheets, Google Cloud Storage](#)
- [Importing libraries and installing dependencies](#)
- [Using Google Cloud BigQuery](#)
- [Forms, Charts, Markdown, & Widgets](#)
- [TensorFlow with GPU](#)
- [Machine Learning Crash Course: Intro to Pandas & First Steps with TensorFlow](#)

### Highlighted Features

#### Seedbank

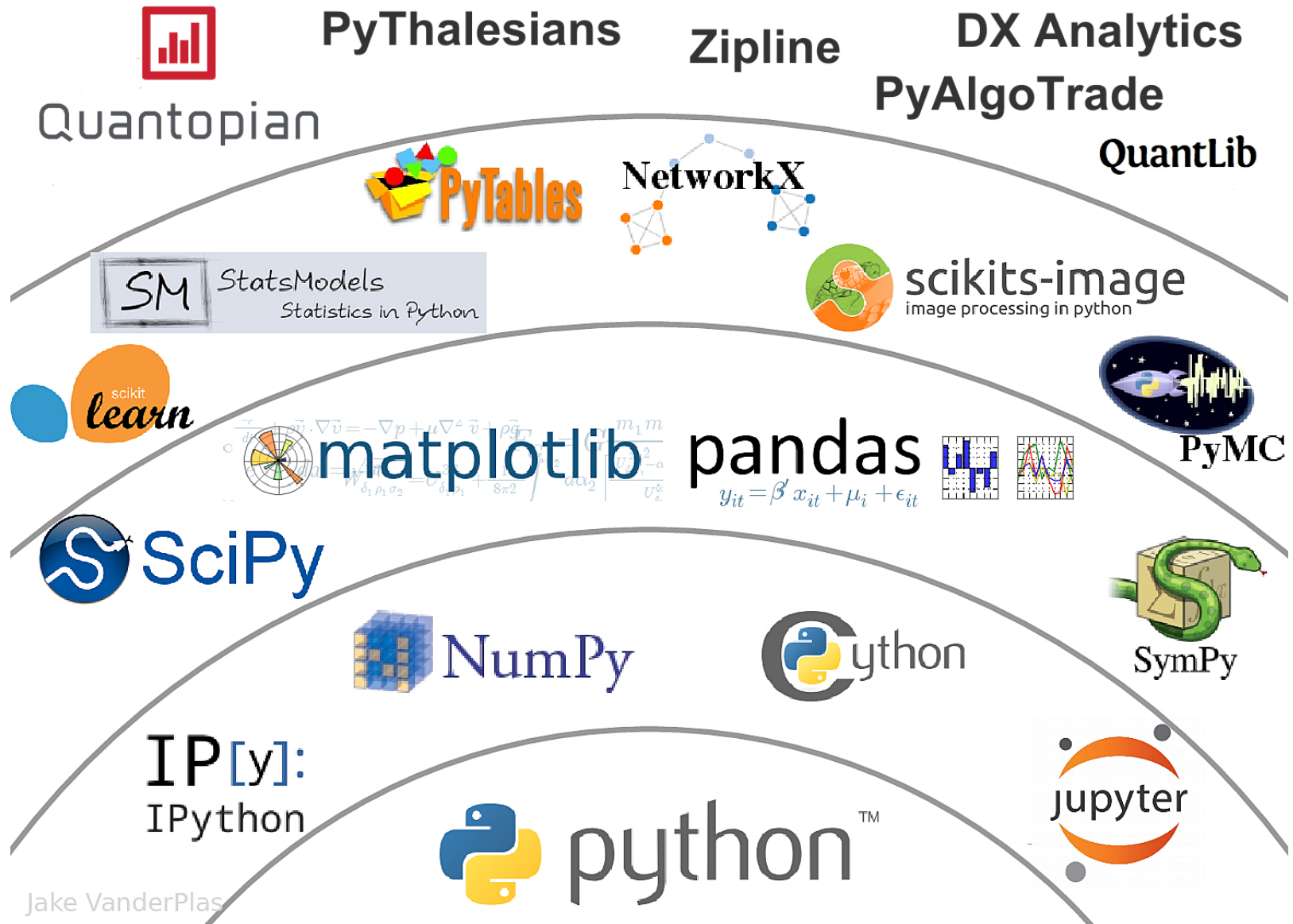
Looking for Colab notebooks to learn from? Check out [Seedbank](#), a place to discover interactive machine learning examples.

#### TensorFlow execution

Colaboratory allows you to execute TensorFlow code in your browser with a single click. The example below adds two matrices.

$$\begin{bmatrix} 1. & 1. & 1. \end{bmatrix} + \begin{bmatrix} 1. & 2. & 3. \end{bmatrix} = \begin{bmatrix} 2. & 3. & 4. \end{bmatrix}$$

# The Quant Finance PyData Stack





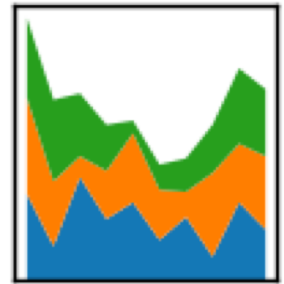
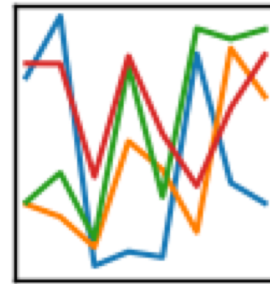
**Python**  
**matplotlib**  
*matplotlib*

# Python

# Pandas

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



# Iris flower data set

**setosa**



**versicolor**



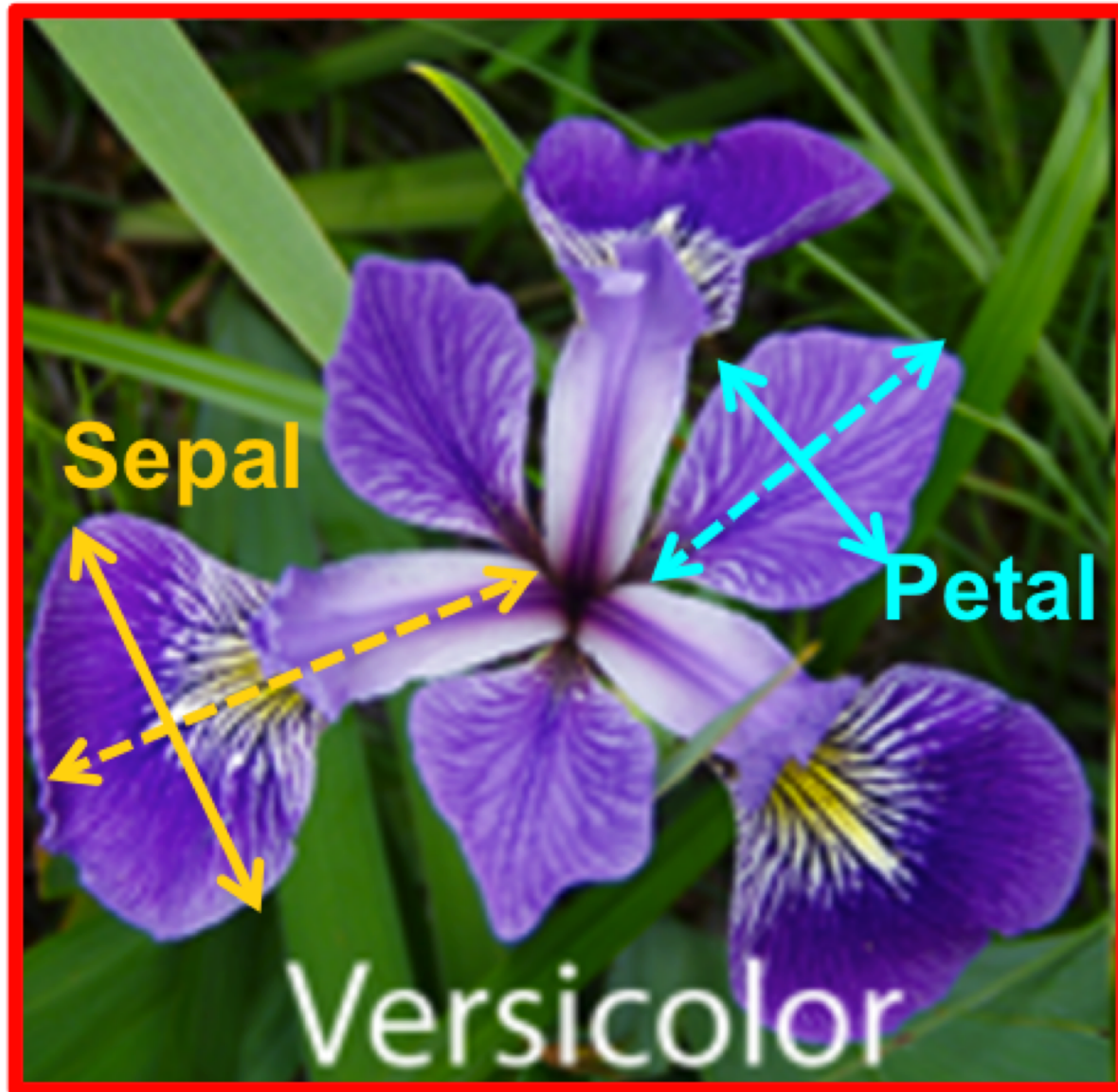
**virginica**



Source: [https://en.wikipedia.org/wiki/Iris\\_flower\\_data\\_set](https://en.wikipedia.org/wiki/Iris_flower_data_set)

Source: <http://suruchifialoke.com/2016-10-13-machine-learning-tutorial-iris-classification/>

# Iris Classification





# iris.data

<https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data>

5.1,3.5,1.4,0.2,Iris-setosa  
4.9,3.0,1.4,0.2,Iris-setosa  
4.7,3.2,1.3,0.2,Iris-setosa  
4.6,3.1,1.5,0.2,Iris-setosa  
5.0,3.6,1.4,0.2,Iris-setosa  
5.4,3.9,1.7,0.4,Iris-setosa  
4.6,3.4,1.4,0.3,Iris-setosa  
5.0,3.4,1.5,0.2,Iris-setosa  
4.4,2.9,1.4,0.2,Iris-setosa  
4.9,3.1,1.5,0.1,Iris-setosa  
5.4,3.7,1.5,0.2,Iris-setosa  
4.8,3.4,1.6,0.2,Iris-setosa  
4.8,3.0,1.4,0.1,Iris-setosa  
4.3,3.0,1.1,0.1,Iris-setosa  
5.8,4.0,1.2,0.2,Iris-setosa  
5.7,4.4,1.5,0.4,Iris-setosa  
5.4,3.9,1.3,0.4,Iris-setosa  
5.1,3.5,1.4,0.3,Iris-setosa  
5.7,3.8,1.7,0.3,Iris-setosa  
5.1,3.8,1.5,0.3,Iris-setosa  
5.4,3.4,1.7,0.2,Iris-setosa  
5.1,3.7,1.5,0.4,Iris-setosa  
4.6,3.6,1.0,0.2,Iris-setosa  
5.1,3.3,1.7,0.5,Iris-setosa  
4.8,3.4,1.9,0.2,Iris-setosa  
5.0,3.0,1.6,0.2,Iris-setosa  
5.0,3.4,1.6,0.4,Iris-setosa

**setosa**



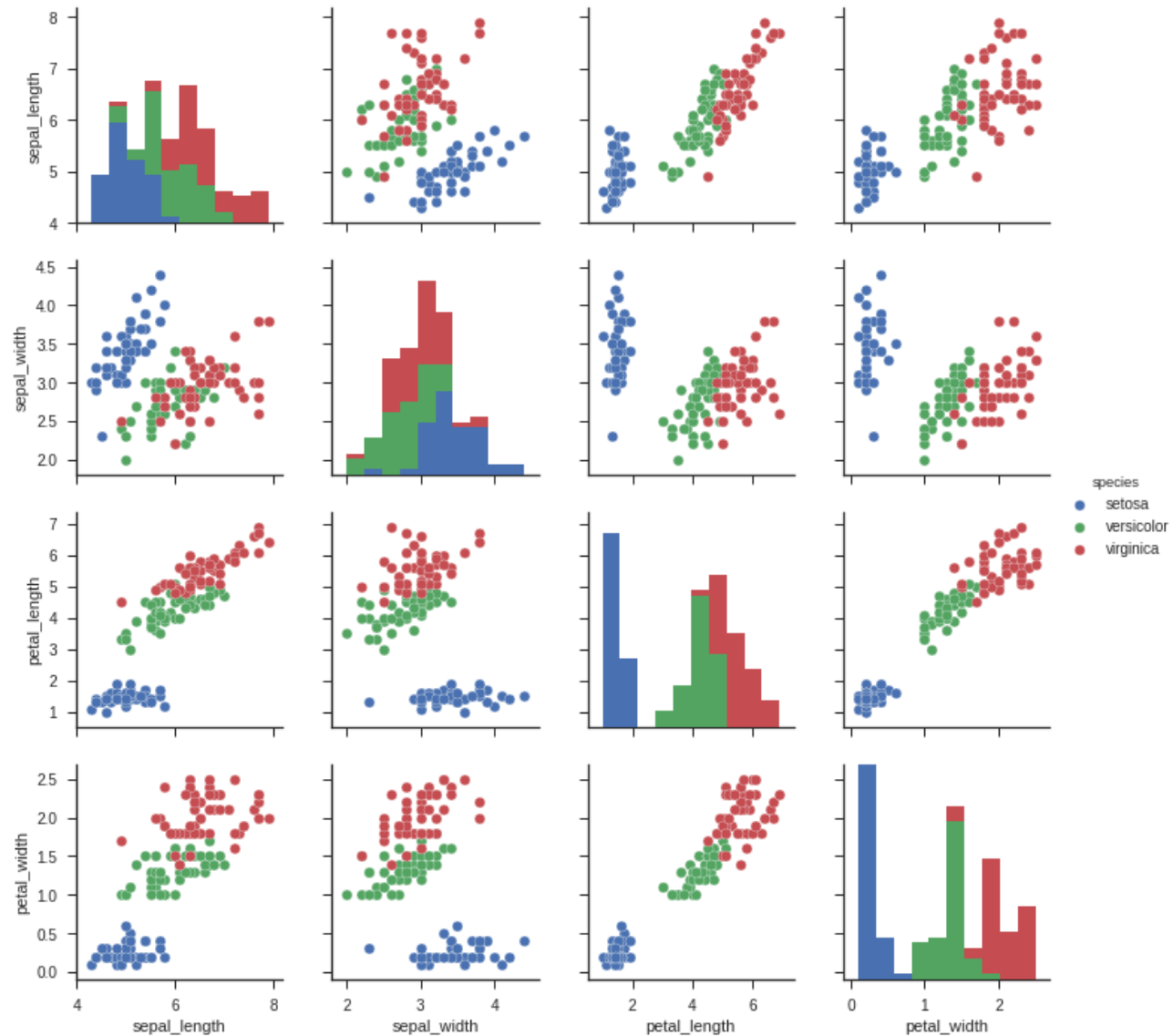
**virginica**



**versicolor**



# Iris Data Visualization



# Connect Google Colab in Google Drive

The screenshot shows the Google Drive web interface. The browser address bar displays `https://drive.google.com/drive/u/2/my-drive`. The left sidebar contains navigation options: New, My Drive, Computers, Shared with me, Recent, Starred, Trash, Backups, and Storage. The 'New' button is highlighted with a red dashed border. A dropdown menu is open from 'New', showing options: New folder..., Upload files..., Upload folder..., Google Docs, Google Sheets, Google Slides, and More. The 'More' option is also highlighted with a red dashed border. A second dropdown menu is open from 'More', listing Google Forms, Google Drawings, Google My Maps, Google Sites, and a red dashed box around the '+ Connect more apps' link. The main content area shows a 'Quick Access' section and a 'Files' section with a table of file management options.

My Drive - Google Drive

Search Drive

My Drive

Quick Access

New

My Drive

Computers

Shared with me

Recent

Starred

Trash

Backups

Storage

0 bytes of 15 GB used

UPGRADE STORAGE

Get Backup and Sync for Mac

New folder...

Upload files...

Upload folder...

Google Docs

Google Sheets

Google Slides

More

Google Forms

Google Drawings

Google My Maps

Google Sites

+ Connect more apps

Name	↑
Store safely	Sync seamlessly
Access anywhere	Share easily

# Google Colab

My Drive - Google Drive x +











https://drive.google.com/drive/u/2/my-drive

Drive

Search Drive

Connect apps to Drive

All ▾ colab x

 <b>ZIP Extractor</b> Extract ZIP files to Google Drive Extraction complete. <a href="#">View extracted files</a> <a href="#">Share</a> <a href="#">Extract another</a>  <b>Test.zip</b> ZIP Extractor 307,585 users	 <b>LUMIN PDF</b> The fast and simple PDF Viewer    Lumin PDF - Beautiful PDF Editor 289,310 users	 <b>cloudconvert</b> CloudConvert 373,161 users
 <b>Sejda</b> Merge PDF - Split PDF - Sejda.com ★★★★★ (1106)	 <b>DocHub</b> Edit, Send & Sign PDFs DocHub - Edit and Sign PDF Docu... 2,131,600 users	 <b>Google Forms</b> Google Forms 4,803,614 users

Get Backup and Sync for Mac

Access anywhere  
Every file is always accessible.

Share easily  
Give others access to any file or folder.



# Google Colab

My Drive - Google Drive x +

https://drive.google.com/drive/u/2/my-drive

Drive

Search Drive

New

My Drive

Computers

Shared with me

Recent

Starred

Trash

Backups

Storage

0 bytes of 15 GB used

[UPGRADE STORAGE](#)


Get Backup and Sync for Mac

Access anywhere

Share easily

## Connect apps to Drive

All colab



**Colaboratory**  
offered by <https://colab.research.google.com>  
A data analysis tool that combines code, output, and descriptive text into one collaborative document.

[+ CONNECT](#)

Productivity

★★★★★ (195)

Name ↑

# Connect Colaboratory to Google Drive

My Drive - Google Drive

https://drive.google.com/drive/u/2/my-drive

Drive

Search Drive

New

My Drive

Computers

Shared with me

Recent

Starred

Trash

Backups

Storage

0 bytes of 15 GB used

[UPGRADE STORAGE](#)

Get Backup and Sync for Mac


Access anywhere

Share easily

Connect apps to Drive

All

colab

 **Colaboratory** was connected to Google Drive.

☒ Make **Colaboratory** the default app for files it can open

OK

RATE IT

Productivity

★★★★★ (195)

Name ↑

# Google Colab

My Drive - Google Drive

https://drive.google.com/drive/u/2/my-drive

Drive

Search Drive

My Drive

Quick Access

New

My Drive

Computers

Shared with me

Recent

Starred

Trash

Backups

Storage

0 bytes of 15 GB used

UPGRADE STORAGE

Get Backup and Sync for Mac

New folder...

Upload files...

Upload folder...

Google Docs

Google Sheets

Google Slides

More

Google Forms

Google Drawings

Google My Maps

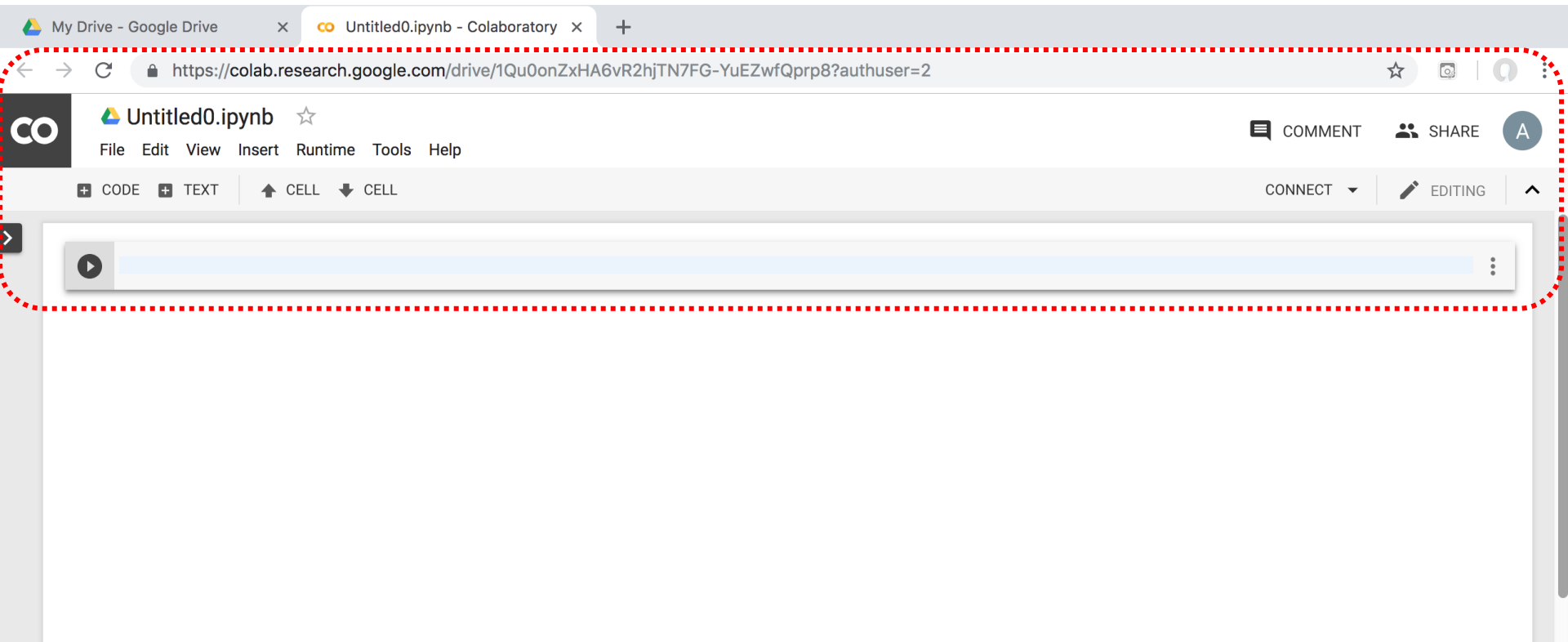
Google Sites

Colaboratory

Connect more apps

Name ↑

# Google Colab



# Google Colab

The screenshot shows the Google Colab web interface. At the top, there's a browser tab for 'Untitled0.ipynb - Colaboratory' and a URL bar with the address 'https://colab.research.google.com/drive/1Qu0onZxHA6vR2hjTN7FG-YuEZwfQprp8?authuser=2'. Below the browser, the Colab logo is on the left, followed by the document title 'Untitled0.ipynb' and a star icon. A menu bar contains 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. The 'Runtime' menu is open, displaying a list of options: 'Run all' (⌘/Ctrl+F9), 'Run before' (⌘/Ctrl+F8), 'Run the focused cell' (⌘/Ctrl+Enter), 'Run selection' (⌘/Ctrl+Shift+Enter), 'Run after' (⌘/Ctrl+F10), 'Interrupt execution' (⌘/Ctrl+M I), 'Restart runtime...' (⌘/Ctrl+M .), 'Restart and run all...', 'Reset all runtimes...', 'Change runtime type' (highlighted with a red dashed box), and 'Manage sessions'. On the right side of the interface, there are buttons for 'COMMENT', 'SHARE', and a user profile icon 'A'. Below these are 'CONNECT' and 'EDITING' buttons. The main workspace area is visible in the background, showing a code editor with a blue line of code.

# Run Jupyter Notebook Python3 GPU Google Colab

The screenshot shows the Google Colab web interface. At the top, the browser tab is titled 'Untitled0.ipynb - Colaboratory'. The address bar shows the URL: <https://colab.research.google.com/drive/1Qu0onZxHA6vR2hjTN7FG-YuEZwfQprp8?authuser=2>. The Colab logo is on the left, and the menu bar includes File, Edit, View, Insert, Runtime, Tools, and Help. On the right, there are buttons for COMMENT, SHARE, and a user profile icon. Below the menu bar, there are tabs for CODE, TEXT, and CELL. The main area is a Jupyter Notebook editor. A 'Notebook settings' dialog box is open in the center. It has a title 'Notebook settings' and two dropdown menus: 'Runtime type' set to 'Python 3' and 'Hardware accelerator' set to 'GPU'. A red dashed box highlights both dropdown menus. Below the dropdowns is a checkbox labeled 'Omit code cell output when saving this notebook' which is currently unchecked. At the bottom right of the dialog are 'CANCEL' and 'SAVE' buttons.

My Drive - Google Drive x Untitled0.ipynb - Colaboratory x +

← → ↻ <https://colab.research.google.com/drive/1Qu0onZxHA6vR2hjTN7FG-YuEZwfQprp8?authuser=2> ☆ 📷 🔊

co Untitled0.ipynb ☆

File Edit View Insert Runtime Tools Help

+ CODE + TEXT ↑ CELL ↓ CELL

CONNECT ▾ EDITING ^

>

**Notebook settings**

Runtime type  
Python 3 ▾

Hardware accelerator  
GPU ▾ ?

☐ Omit code cell output when saving this notebook

CANCEL SAVE

# Google Colab Python Hello World

```
print('Hello World')
```

The screenshot displays the Google Colaboratory web interface. At the top, the browser tab is labeled 'Untitled0.ipynb - Colaboratory'. The address bar shows the URL: <https://colab.research.google.com/drive/1Qu0onZxHA6vR2hjTN7FG-YuEZwfQprp8?authuser=2#scrollTo=6s-m3sER8G1u>. The Colab logo is on the left, and the document title 'Untitled0.ipynb' is in the center. A menu bar includes 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. On the right, there are buttons for 'COMMENT', 'SHARE', and a user profile icon. Below the menu bar, a toolbar shows '+ CODE', '+ TEXT', '↑ CELL', and '↓ CELL'. The status bar indicates 'CONNECTED' with a green checkmark and 'EDITING' with a pencil icon. The main workspace contains a single code cell with a play button icon on the left. The code inside the cell is `print('Hello World')`. Below the code, the output is displayed as 'Hello World' with a copy icon to its left.

Untitled0.ipynb - Colaboratory

https://colab.research.google.com/drive/1Qu0onZxHA6vR2hjTN7FG-YuEZwfQprp8?authuser=2#scrollTo=6s-m3sER8G1u

Untitled0.ipynb

File Edit View Insert Runtime Tools Help

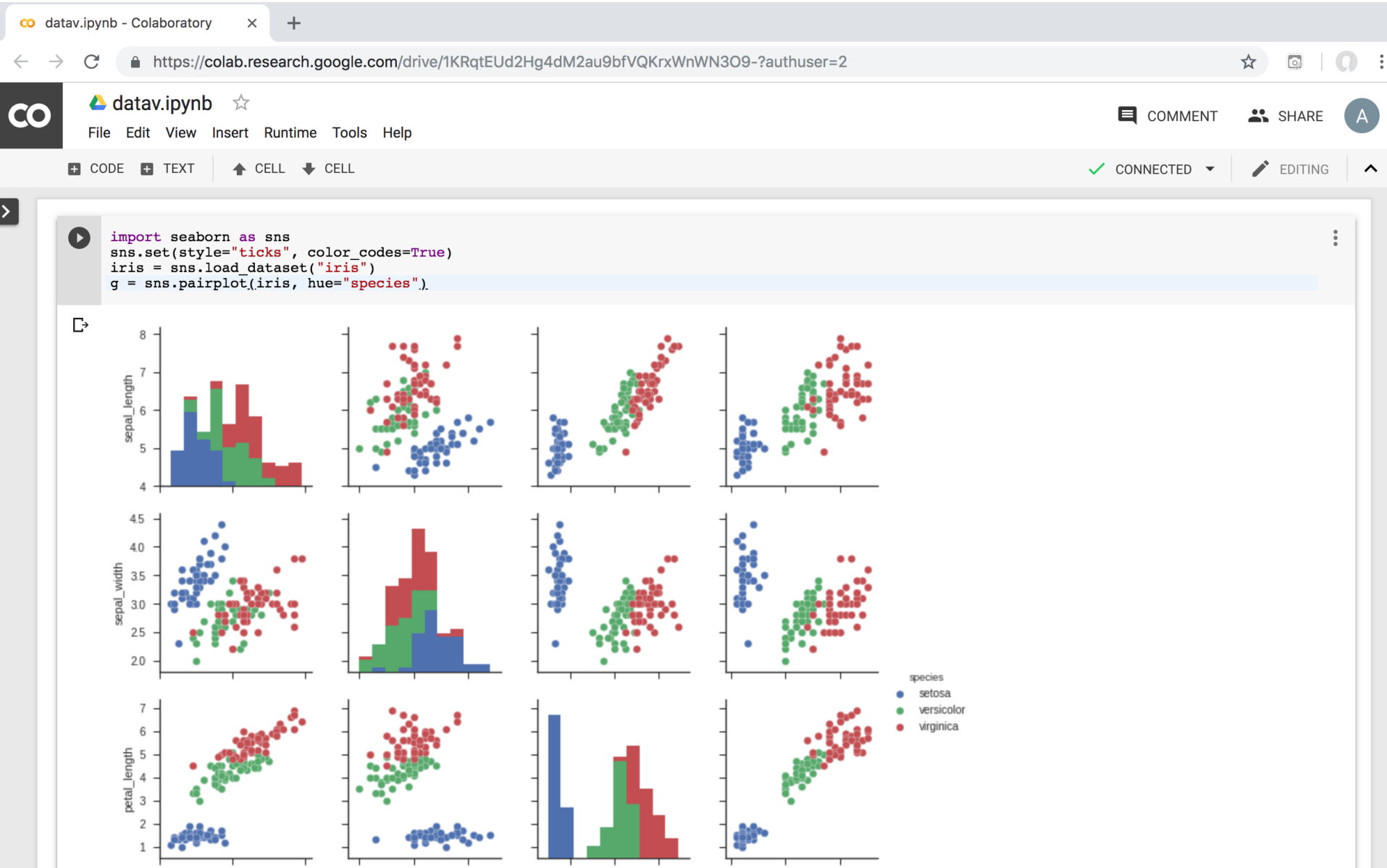
+ CODE + TEXT ↑ CELL ↓ CELL

CONNECTED EDITING

```
print('Hello World')
```

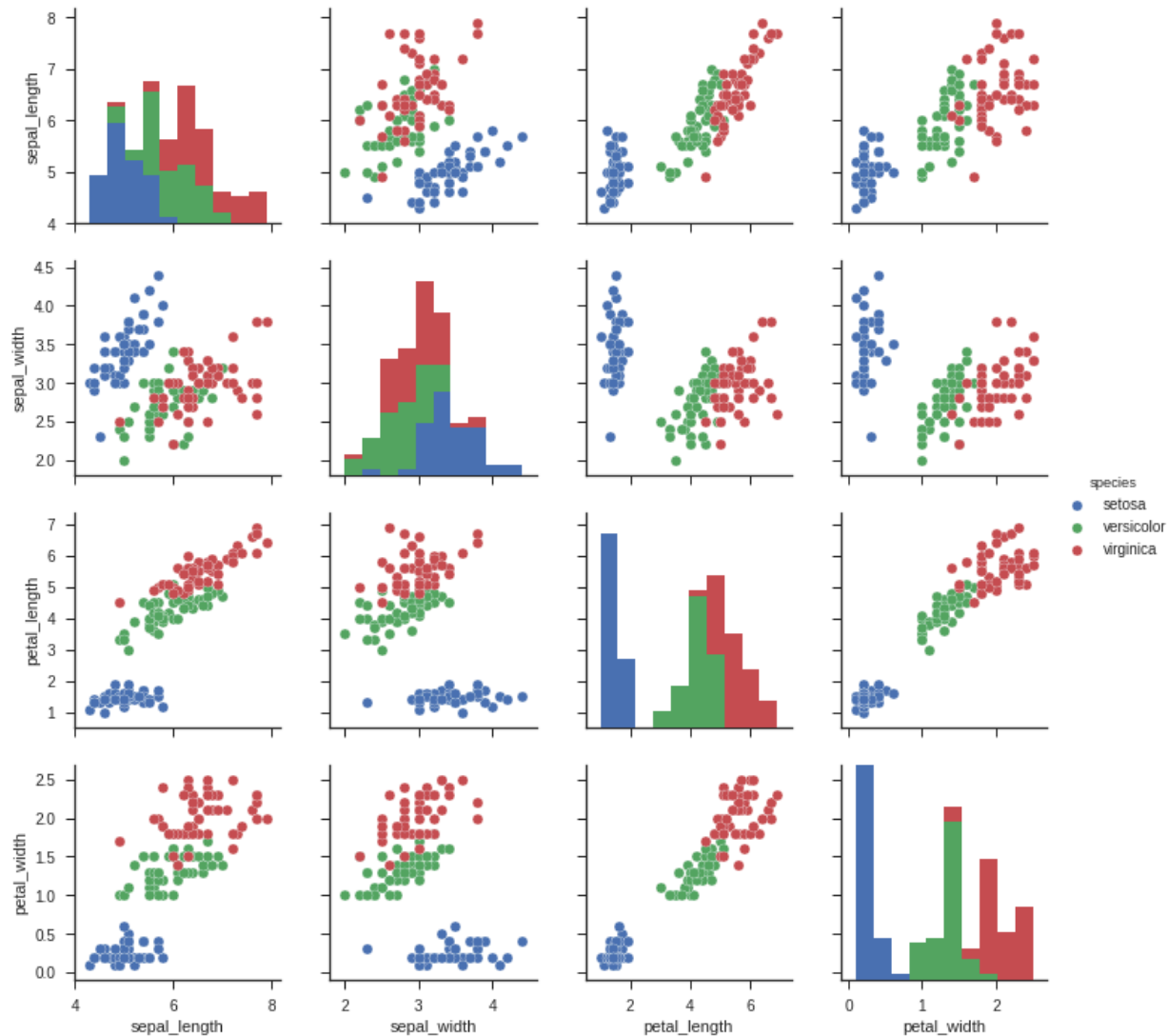
Hello World

# Data Visualization in Google Colab





```
import seaborn as sns
sns.set(style="ticks", color_codes=True)
iris = sns.load_dataset("iris")
g = sns.pairplot(iris, hue="species")
```



```
import numpy as np
import pandas as pd
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns
from pandas.plotting import scatter_matrix

# Load dataset
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"
names = ['sepal-length', 'sepal-width', 'petal-length', 'petal-width', 'class']
df = pd.read_csv(url, names=names)

print(df.head(10))
print(df.tail(10))
print(df.describe())
print(df.info())
print(df.shape)
print(df.groupby('class').size())

plt.rcParams["figure.figsize"] = (10,8)
df.plot(kind='box', subplots=True, layout=(2,2), sharex=False, sharey=False)
plt.show()

df.hist()
plt.show()

scatter_matrix(df)
plt.show()

sns.pairplot(df, hue="class", size=2)
```

```
import numpy as np
import pandas as pd
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns
from pandas.plotting import scatter_matrix
```

```
# Import Libraries
import numpy as np
import pandas as pd
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns
from pandas.plotting import scatter_matrix
print('imported')
```

imported

```
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"
names = ['sepal-length', 'sepal-width', 'petal-length', 'petal-width', 'class']
df = pd.read_csv(url, names=names)
print(df.head(10))
```

```
# Load dataset
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"
names = ['sepal-length', 'sepal-width', 'petal-length', 'petal-width', 'class']
df = pd.read_csv(url, names=names)
print(df.head(10)).
```

	sepal-length	sepal-width	petal-length	petal-width	class
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
5	5.4	3.9	1.7	0.4	Iris-setosa
6	4.6	3.4	1.4	0.3	Iris-setosa
7	5.0	3.4	1.5	0.2	Iris-setosa
8	4.4	2.9	1.4	0.2	Iris-setosa
9	4.9	3.1	1.5	0.1	Iris-setosa

# df.tail(10)

```
print(df.tail(10)).
```

	sepal-length	sepal-width	petal-length	petal-width	class
140	6.7	3.1	5.6	2.4	Iris-virginica
141	6.9	3.1	5.1	2.3	Iris-virginica
142	5.8	2.7	5.1	1.9	Iris-virginica
143	6.8	3.2	5.9	2.3	Iris-virginica
144	6.7	3.3	5.7	2.5	Iris-virginica
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

# df.describe()

```
print(df.describe())
```

	sepal-length	sepal-width	petal-length	petal-width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.054000	3.758667	1.198667
std	0.828066	0.433594	1.764420	0.763161
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

```
print(df.info())  
print(df.shape)
```

```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 150 entries, 0 to 149  
Data columns (total 5 columns):  
sepal-length      150 non-null float64  
sepal-width       150 non-null float64  
petal-length      150 non-null float64  
petal-width       150 non-null float64  
class             150 non-null object  
dtypes: float64(4), object(1)  
memory usage: 5.9+ KB  
None
```

```
print(df.shape)
```

```
(150, 5)
```

```
df.groupby( 'class' ).size()
```

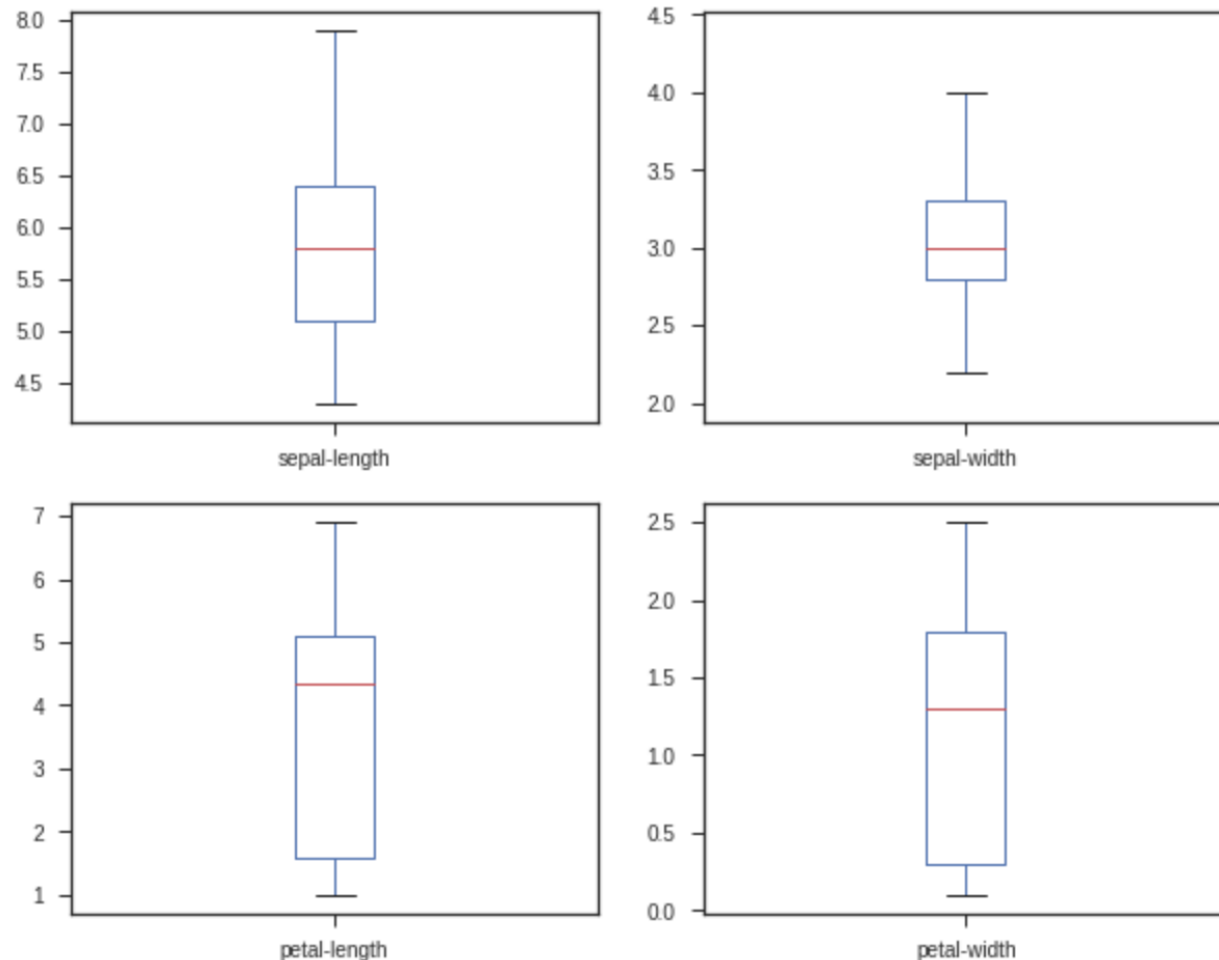
```
print(df.groupby( 'class' ).size())
```

```
class
Iris-setosa      50
Iris-versicolor  50
Iris-virginica   50
dtype: int64
```



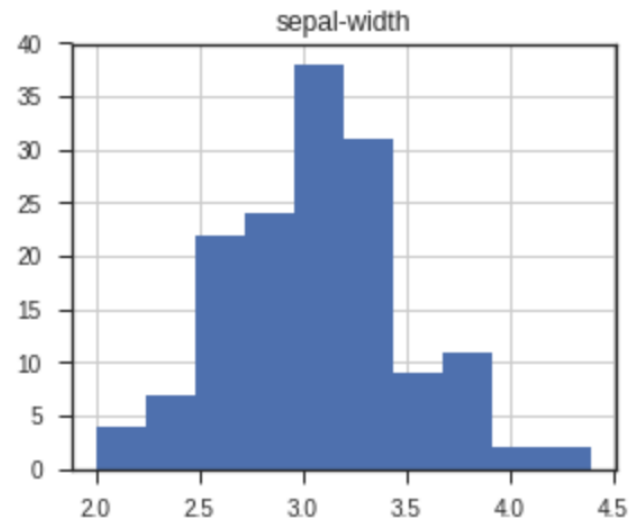
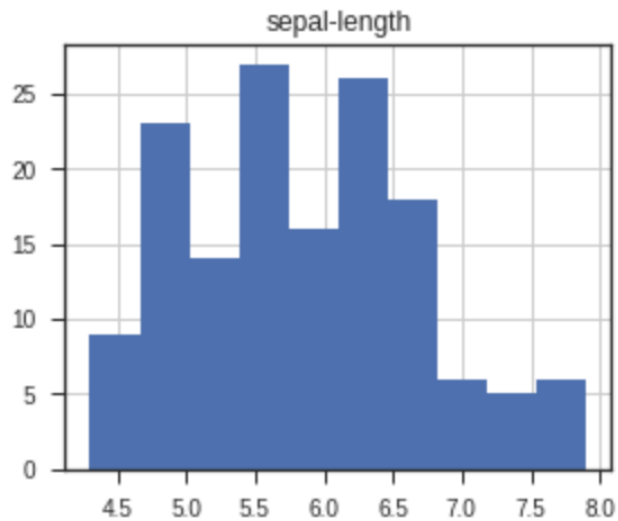
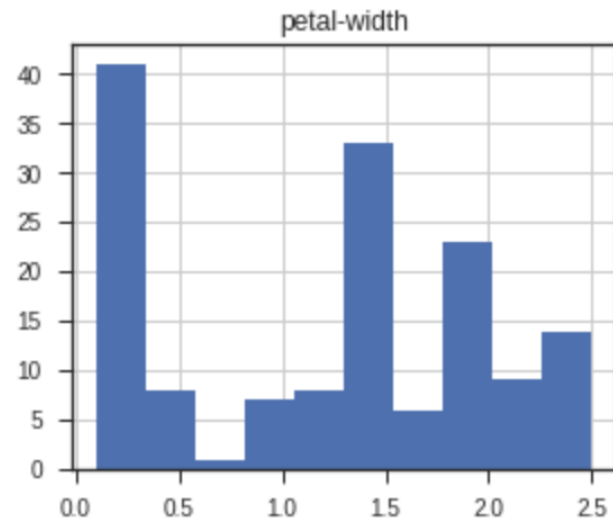
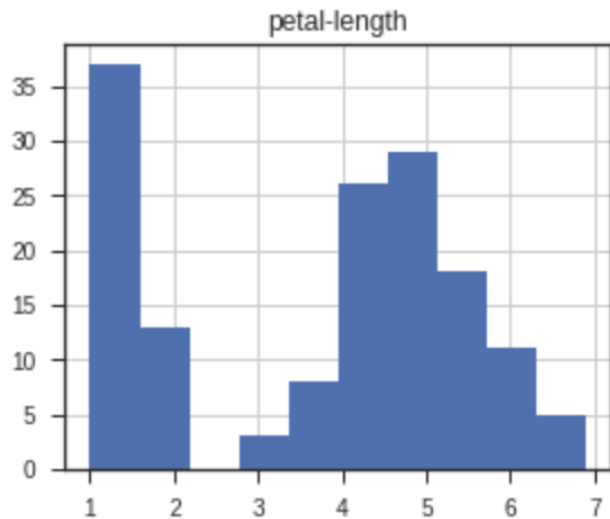
```
plt.rcParams["figure.figsize"] = (10,8)
df.plot(kind='box', subplots=True, layout=(2,2), sharex=False, sharey=False)
plt.show()
```

```
plt.rcParams["figure.figsize"] = (10,8)
df.plot(kind='box', subplots=True, layout=(2,2), sharex=False, sharey=False)
plt.show()
```



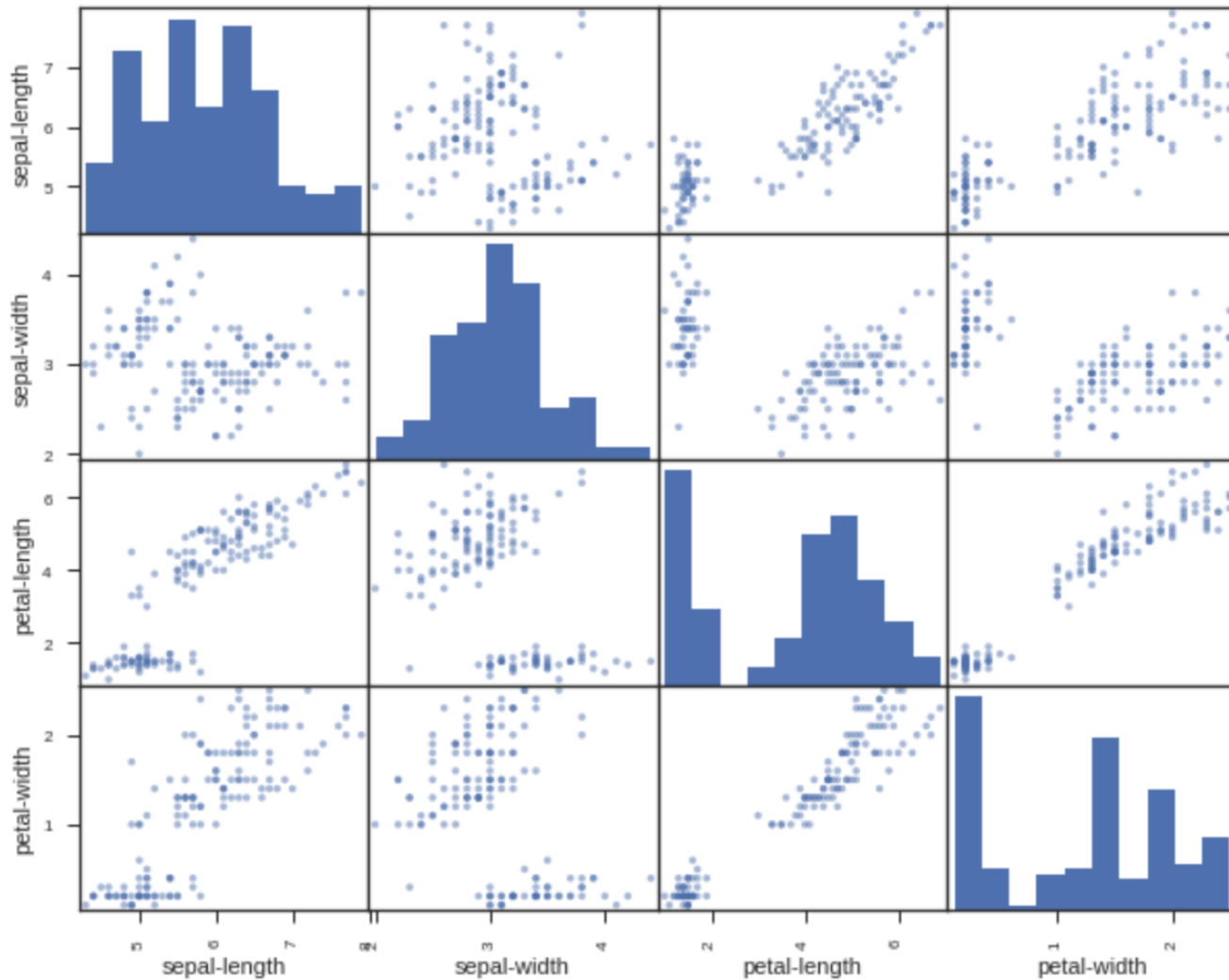
```
df.hist()  
plt.show()
```

```
df.hist()  
plt.show()
```



```
scatter_matrix(df)  
plt.show()
```

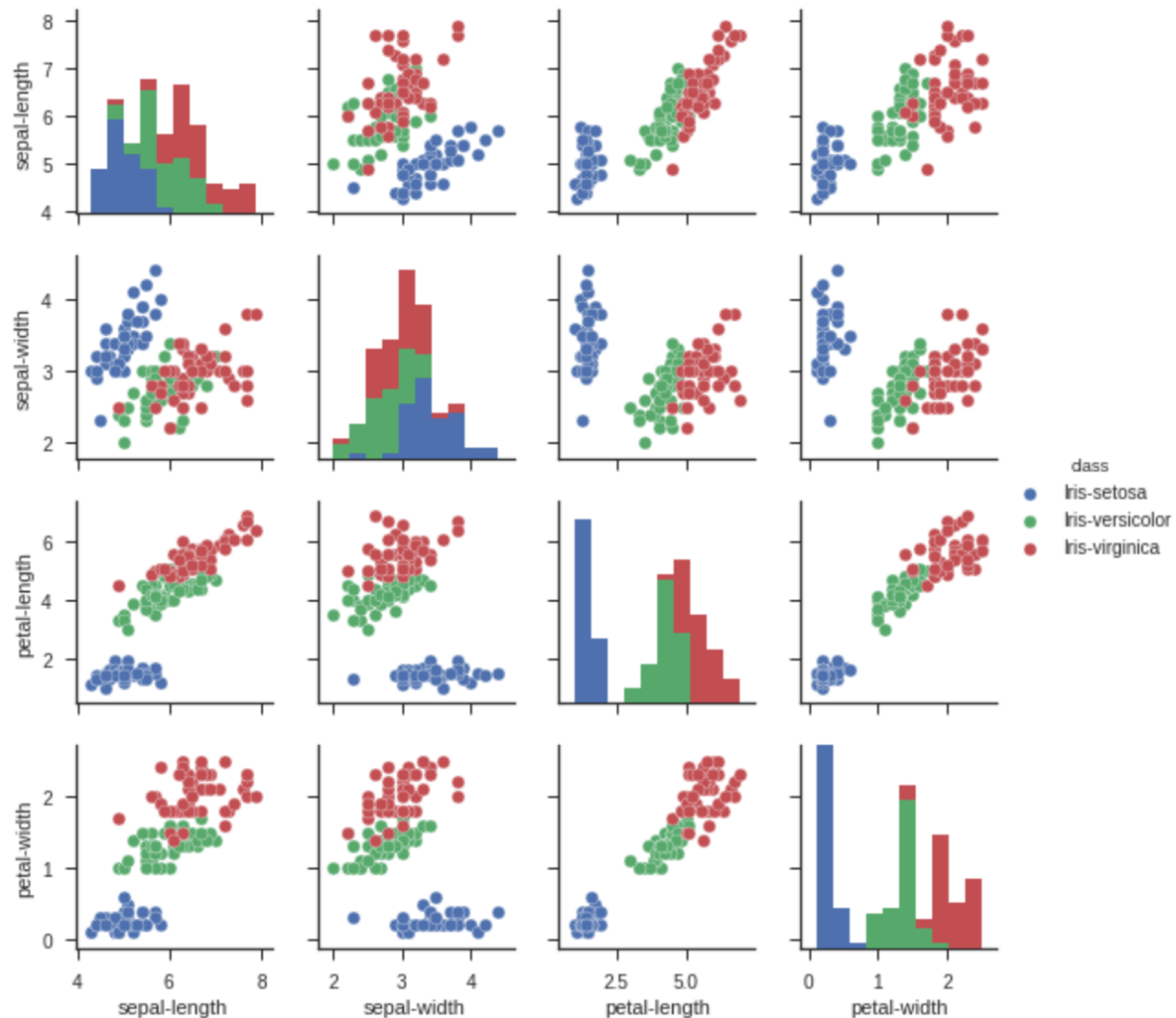
```
scatter_matrix(df)  
plt.show(.)
```



# `sns.pairplot(df, hue="class", size=2)`

```
sns.pairplot(df, hue="class", size=2)
```

<seaborn.axisgrid.PairGrid at 0x7f1d21267390>



# References

- Ramesh Sharda, Dursun Delen, and Efraim Turban (2017), Business Intelligence, Analytics, and Data Science: A Managerial Perspective, 4th Edition, Pearson.
- EMC Education Services (2015), Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, Wiley
- SAS Modernization architectures - Big Data Analytics, <http://www.slideshare.net/deepakramanathan/sas-modernization-architectures-big-data-analytics>