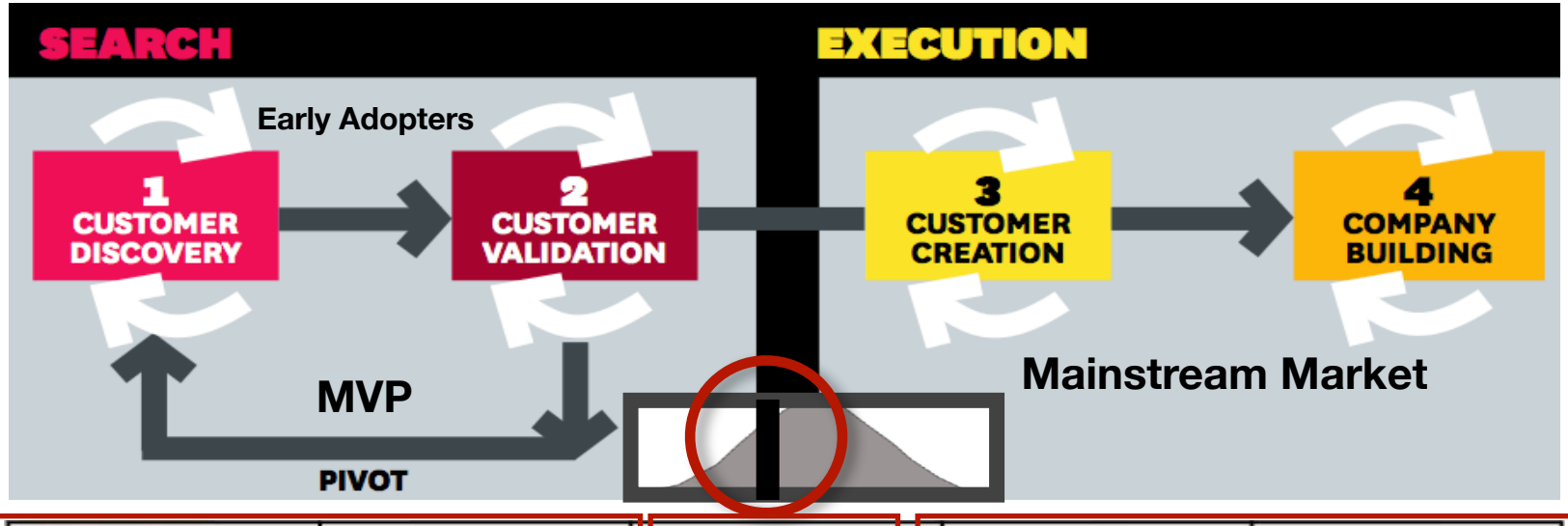
COM5940: NEW MEDIA BUSINESS MODEL & INNOVATION
OVERVIEW OF MACHINE LEARNING AND PREDICTIVE ANALYTICS

CUHK BUSINESS SCHOOL

Center for Entrepreneurship

Bernard Suen
Center for Entrepreneurship
Chinese University of Hong Kong

Quick Recap on Stack Strategy, Data Preparation & Exploratory Data Analysis



PROBLEM

SOLUTION

KEY METRICS

How

UNIQUE VALUE PROPOSITION

What

UNFAIR ADVANTAGE

CUSTOMER SEGMENTS

Why

CHANNELS

Who

COST STRUCTURE

REVENUE STREAMS

How much

最小化可行产品 MVP(Minimum Viable Product)

最小化可行架构 MVA(Minimum Viable Architecture

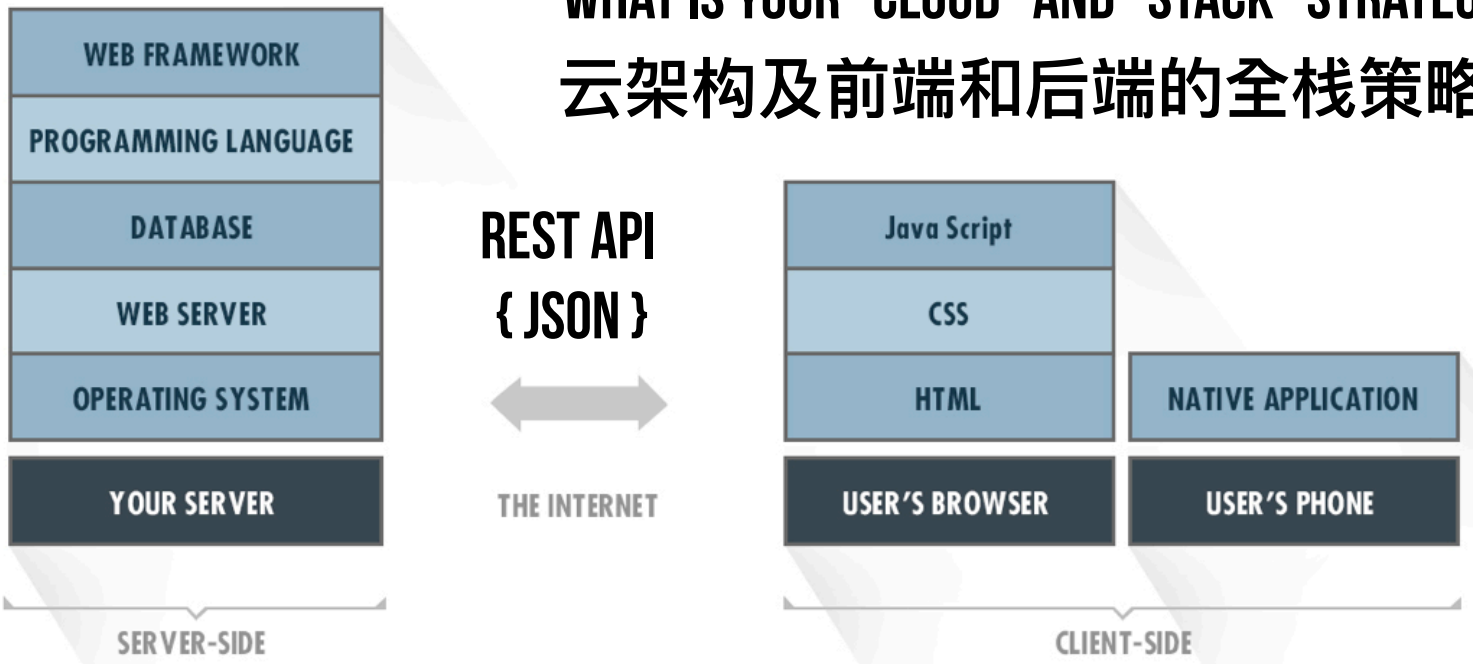
PLATFORM (平台)

DEVELOPMENT
开发平台

DEPLOYMENT
发放营运平台

兼容和整合考虑

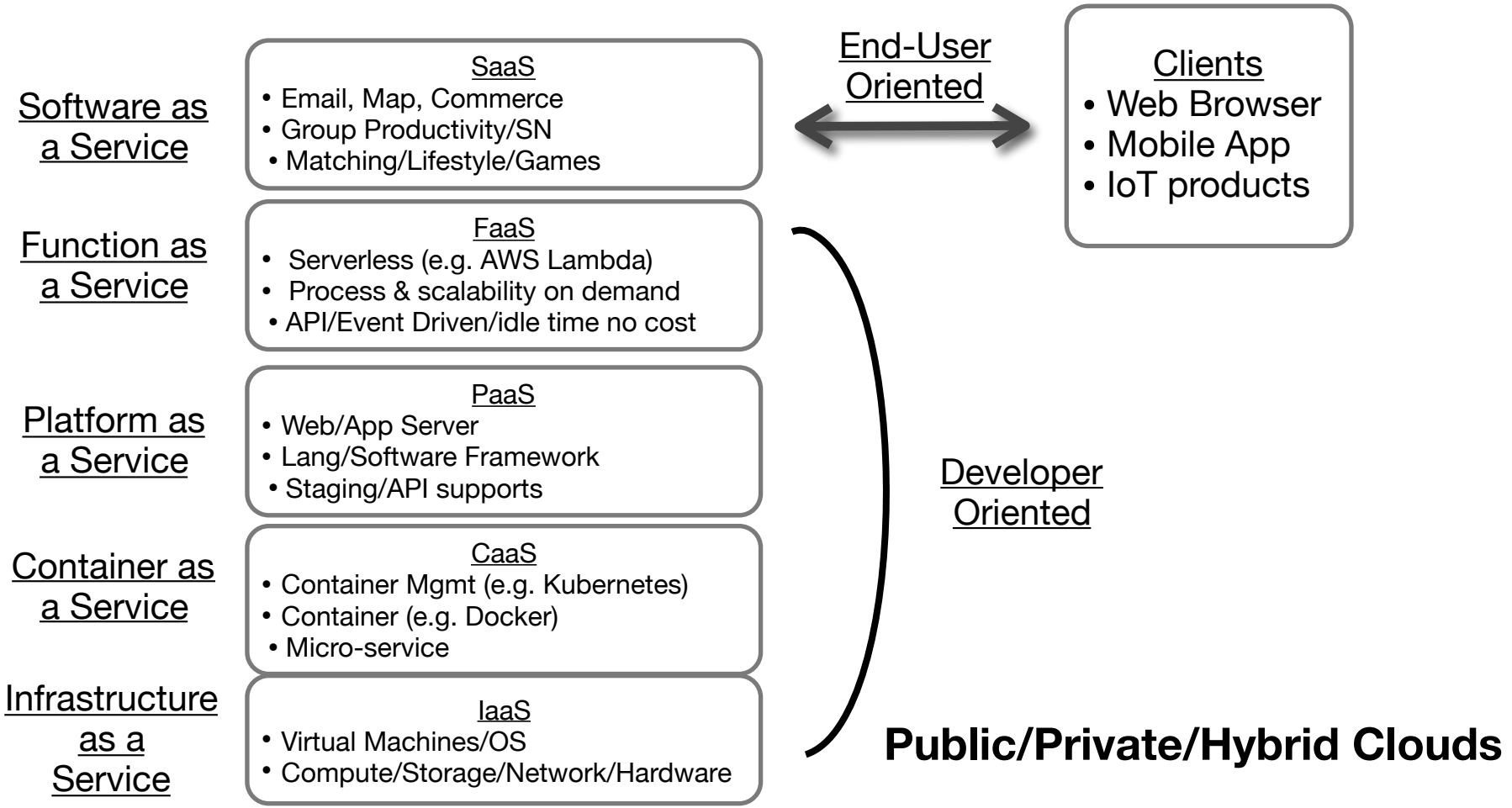
WHAT IS YOUR “CLOUD” AND “STACK” STRATEGY?
云架构及前端和后端的全栈策略



BACK-END TECHNOLOGY
后台

FRONT-END TECHNOLOGY
前台

云计算服务架构重组


Software as a Service
SaaS
• Email, Map, Commerce
• Group Productivity/SN
• Matching/Lifestyle/Games
End-User Oriented
Clients
• Web Browser
• Mobile App
• IoT products
Function as a Service
FaaS
• Serverless (e.g. AWS Lambda)
• Process & scalability on demand
• API/Event Driven/idle time no cost
Platform as a Service
PaaS
• Web/App Server
• Lang/Software Framework
• Staging/API supports
Developer Oriented
Container as a Service
CaaS
• Container Mgmt (e.g. Kubernetes)
• Container (e.g. Docker)
• Micro-service
Infrastructure as a Service
IaaS
• Virtual Machines/OS
• Compute/Storage/Network/Hardware
Public/Private/Hybrid Clouds

MVA Strategy —> Web Stack Strategy

FRONT-END 前端

Which Cross-Platform Framework (e.g. React/React Native, Flutter, Xamarin, PhoneGap/HTML/CSS/JQuery)?

Pure Native Support (e.g. Swift, Java, Objective C)?

Your MVA

BACK-END 後端

Which database (e.g. MySQL, MongoDb, DynamoDB) and App Service Framework (e.g. Flask, Java Spring)?

Function as a Service (FAAS)服务

Which serverless service (e.g. AWS Lambda, Microsoft Function)?

Platform as a Service (PAAS)服务

Which platform (e.g. AWS Elastic, Beanstalk, PythonAnywhere)?

Container as a Service (CAAS)服务

Which container support (e.g. AWS ECS, Microsoft AKS)?

Infrastructure as a Service(IAAS)服务

Which VM Service (e.g. AWS EC2, Microsoft VM)?

MVA Strategy —> Web Stack Strategy

FRONT-END 前端

Which Cross-Platform Framework (e.g. React/React Native, Flutter, Xamarin, PhoneGap/HTML/CSS/JQuery)?

Pure Native Support (e.g. Swift, Java, Objective C)?

Your Whole Product

BACK-END 後端

Which database (e.g. MySQL, MongoDb, DynamoDB) and App Service Framework (e.g. Flask, Java Spring)?

Function as a Service (FAAS)服务

Which serverless service (e.g. AWS Lambda, Microsoft Function)?

Platform as a Service (PAAS)服务

Which platform (e.g. AWS Elastic, Beanstalk, PythonAnywhere)?

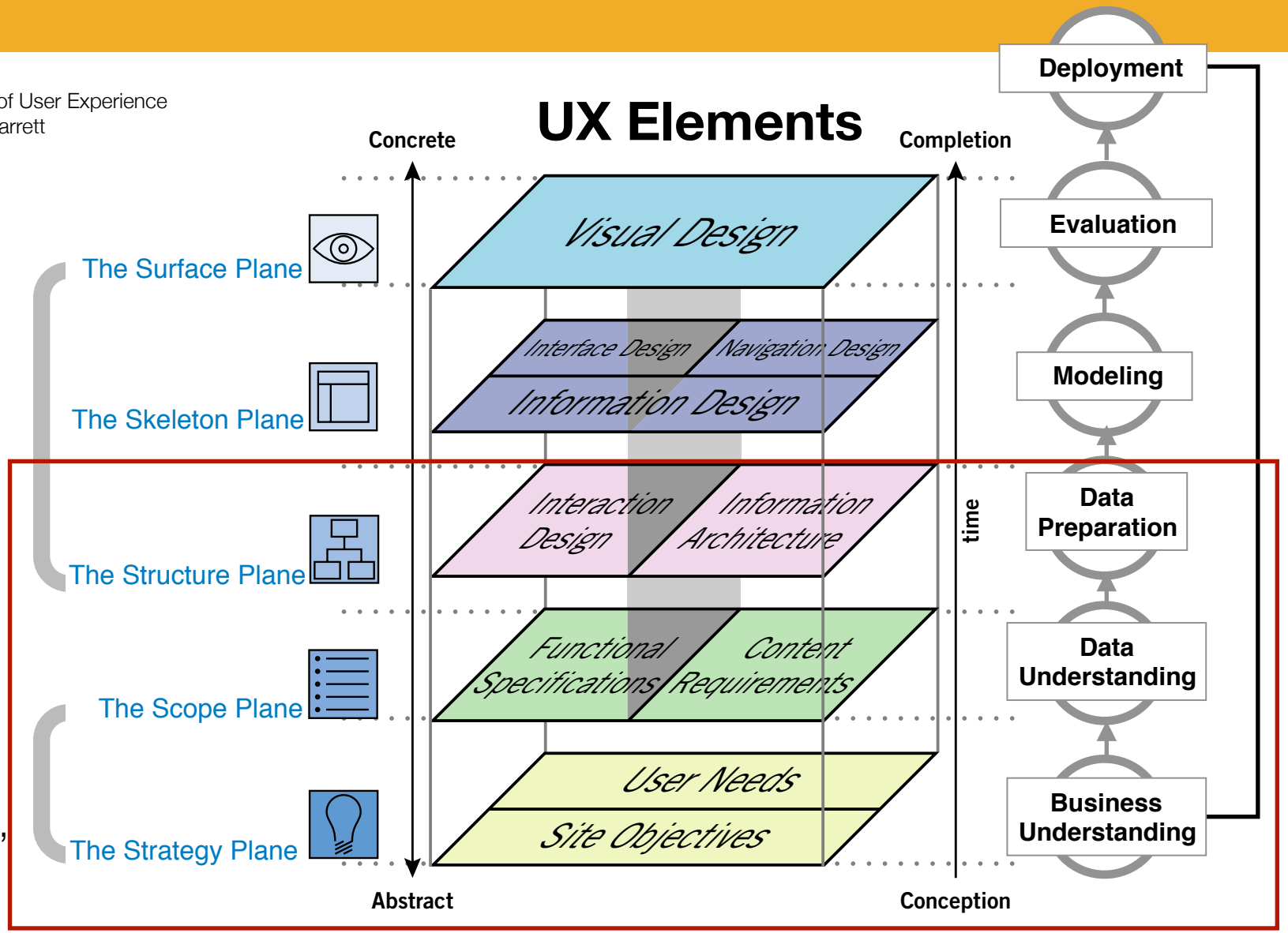
Container as a Service (CAAS)服务

Which container support (e.g. AWS ECS, Microsoft AKS)?

Infrastructure as a Service(IAAS)服务

Which VM Service (e.g. AWS EC2, Microsoft VM)?

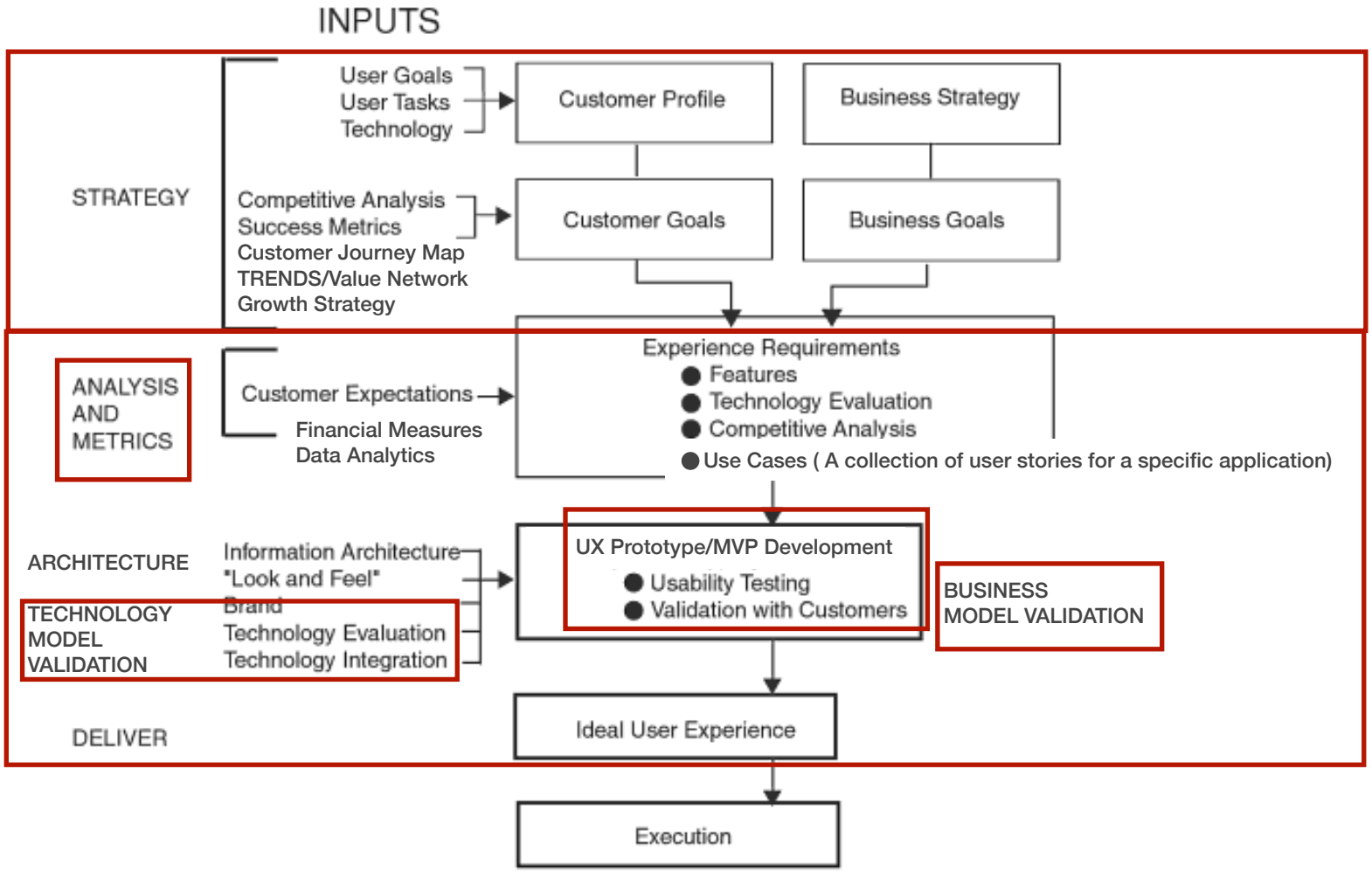
Source: Elements of User Experience
by Jesse James Garrett



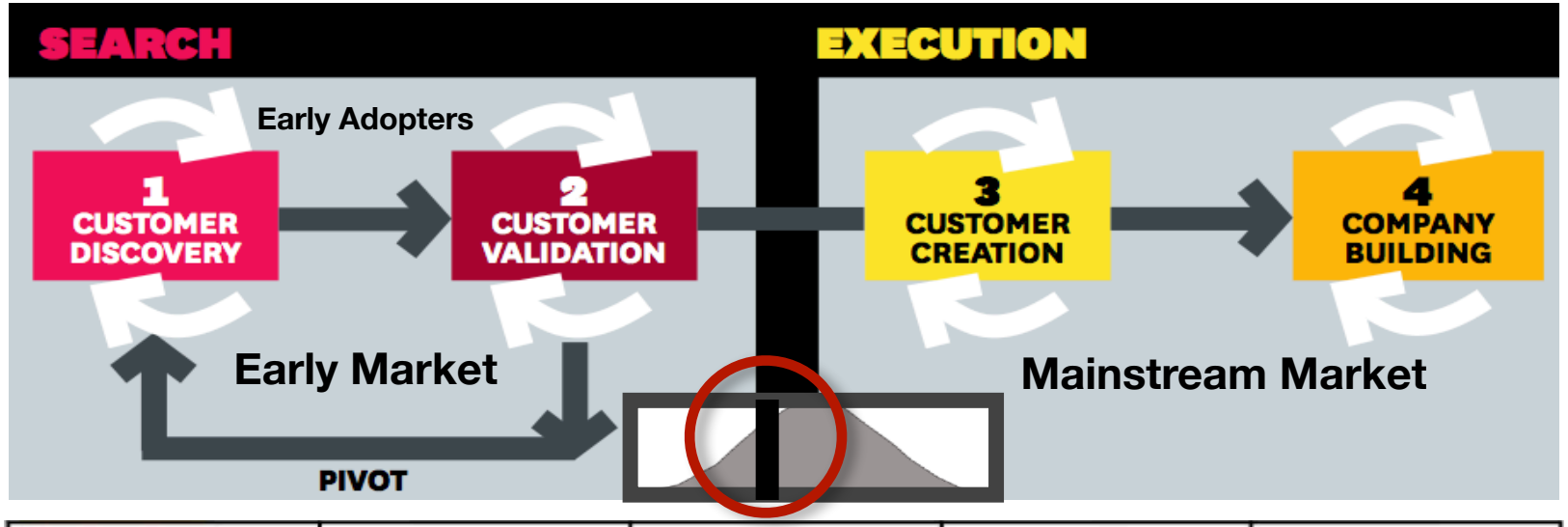
Solution Space
how and how much

Problem Space
who, what, and why

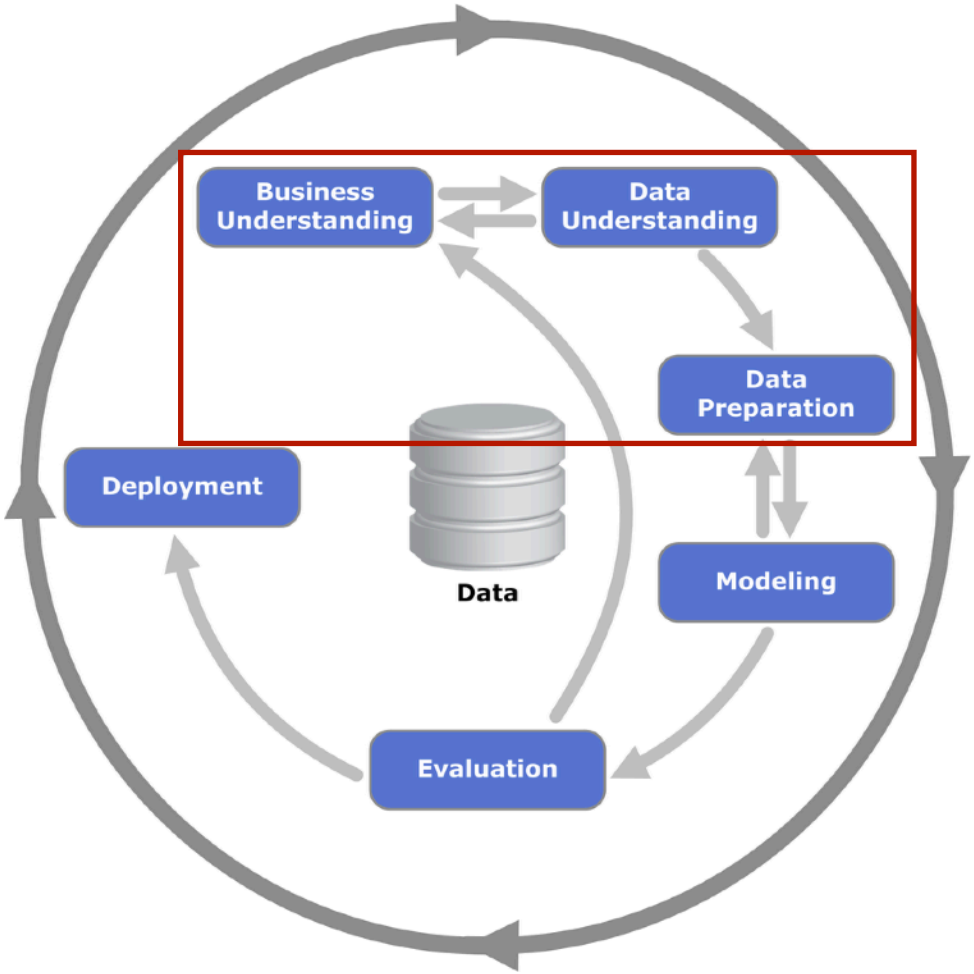
Stack, Analytic Model and API



Adapted from: Karen Donoghue, “Built for Use:Driving Profitability Through the User Experience”



PROBLEM	SOLUTION	UNIQUE VALUE PROPOSITION	UNFAIR ADVANTAGE	CUSTOMER SEGMENTS
	KEY METRICS e.g. 2A3R, burn rate, break-even point.		CHANNELS	
COST STRUCTURE			REVENUE STREAMS	



1. Trends, market, competitor, segmentation and positioning analysis
2. Persona, Journey and story mapping
3. Problem definition (functional and content requirement)
4. Data and content acquisition and production (including scraping)
5. Data cleaning and exploration

CRISP-DM Framework

Source: Wikipedia

Comparison Between ParseHub and BeautiSoup

	ParseHub	Scrapy/BeautifulSoup
Ease of Use	X	
Crawling	X	X
Parsing	X	X
Speed		X
Control		X

Comparison Between OpenRefine and Pandas

	Open Refine	Pandas
Ease of Use	X	
Pre-built Features	X	X
Grouping	X	X
Speed		X
Control		X

Using Python to Scrap, Transform, Visualize and Explore Data to Find Meaningful Insights

Scraping Hong Kong Economic Journal (http://startupbeat.hkej.com)

```
1  import requests
2  import csv
3  import pandas as pd
4  from bs4 import BeautifulSoup
5
6  # quote_page = requests.get('http://startupbeat.hkej.com/?tag=fintech&paged=1')
7  # soup = BeautifulSoup(quote_page.text,'html.parser')
8
9  header = ['page #','title','url','details','post date']
10 data = []
11 # Display and store away 2 pages of scrapped data from startupbeat.hkej.com
12 for i in range(1,4):
13     quote_page = requests.get('http://startupbeat.hkej.com/?tag=fintech&paged='+str(i))
14     print("\n***** Page " + str(i) +" in action *****")
15     soup = BeautifulSoup(quote_page.content,'html.parser')
16
17     for article in soup.find_all("div", attrs={"class":"archive-text"}):
18     # for article in soup.find_all('div',class_='archive-text'):
19         page_no = str(i)
20         title = article.a.text.encode('utf-8').strip()
21         decoded_title = title.decode('utf-8')
22         url = article.a.get('href')
23         details = article.p.text.encode('utf-8').strip()
24         decoded_details = details.decode('utf-8')
25         post_date = article.div.ul.li.text
26         print(decoded_title)
27         print(url)
28         print(decoded_details)
```

	page #	title	url	details	post date
0	1	無現金支付進一步突圍 (方保僑)	http://startupbeat.hkej.com/?p=85359	抗疫過程中，很多生活細節亦隨之改變，例如以往我經常會帶備小量現金傍身，但其實鈔票布滿細菌，所...	Posted March 23, 2020
1	1	金融科技協會憂肺疫礙融資	http://startupbeat.hkej.com/?p=85349	香港金融科技協會（FTAHK）創會兼董事會成員董欣怡受訪時指出，「我們估計（疫情）第一波影響...	Posted March 23, 2020
2	1	比特幣疫市瀉 避險價值淪喪 投資平台eToro：目前太投機 未來仍看好	http://startupbeat.hkej.com/?p=85333	比特幣（Bitcoin）一直為極具爭議的新類型「資產」隨着近期新冠肺炎疫情在全球爆發，股、債...	Posted March 23, 2020
3	1	黑客松大賽避疫Hack From Home	http://startupbeat.hkej.com/?p=85099	康宏（01019）及數碼港合力舉辦首屆「FINSPIRE FinTech Online Ha...	Posted March 16, 2020
4	1	港初創Oriente籌1.56億攻菲國FinTech	http://startupbeat.hkej.com/?p=84744	本港金融科技初創Oriente周三（4日）宣布，已從資產投資公司Silverhorn Gro...	Posted March 6, 2020
5	1	AI 平台配對融資服務 助港中小企疫市撲水	http://startupbeat.hkej.com/?p=84639	運用AI科技為企業配對融資服務的金融科技平台FinMonster，其創辦人鄭文耀及陳健明受訪...	Posted March 5, 2020
6	1	FinTech初創去年共吸2644億 逾億美元融資83宗 勁飆八成	http://startupbeat.hkej.com/?p=84225	科技媒體TechCrunch報道指出，全球金融科技的初創企業，去年共錄得1912宗融資交易，...	Posted February 25, 2020
7	1	雷蛇數碼金融瞄準年輕人 藉玩家商戶網絡 拓展東南亞	http://startupbeat.hkej.com/?p=84176	以開發電競及電子遊戲設備成名的雷蛇，近年積極拓展金融科技業務。公司早前宣布，已向新加坡申請數...	Posted February 24, 2020
8	1	OpenRice撐食肆 外賣自取免收佣	http://startupbeat.hkej.com/?p=84158	OpenRice《開飯喇》宣布，由2月21日起至4月30日將全額豁免「外賣自取」的服務佣金，...	Posted February 22, 2020
9	1	AI財策師助散戶穩健增值 度身訂造投資組合 一萬元入場	http://startupbeat.hkej.com/?p=84078	由多名前銀行家共同開發的智能投資平台Kristal.AI，利用人工智能科技分析風險承受力、目...	Posted February 20, 2020
10	2	本地加密幣交易商Amber估值7.8億	http://startupbeat.hkej.com/?p=83984	美國《福布斯》報道，本港加密貨幣交易科企Amber近日獲得A輪融資2800萬美元（約2.18...	Posted February 18, 2020
11	2	超級App正在形成 (老占)	http://startupbeat.hkej.com/?p=83640	馬雲曾經講過，如果銀行不改變，我們去改變銀行。二○一八年六月，螞蟻金服完成140億美元的C輪...	Posted February 8, 2020

Basic Descriptive Statistic Functions in Python (Self-Computed vs. Pre-programmed)

In [25]:

```
# Compare Numpy and Scipy statistical functions and my own functions
import numpy as np
from scipy import stats
from matplotlib import pyplot as plt

def round(x):
    pos = str(x).find(".")+1
    val = str(x)[pos:pos+1]
    if (int(val) >= 5):
        return(str(x+1)[0:pos-1])
    else:
        return(str(x)[0:pos-1])

def average_calc(list):
    total = 0
    counter = 0
    for i in list:
        total = total + i
        # print(counter,i)
        counter = counter + 1
    # print ("total:",total)
    # print("no. of items:",counter)
    return total/counter

def median_calc(list):
```

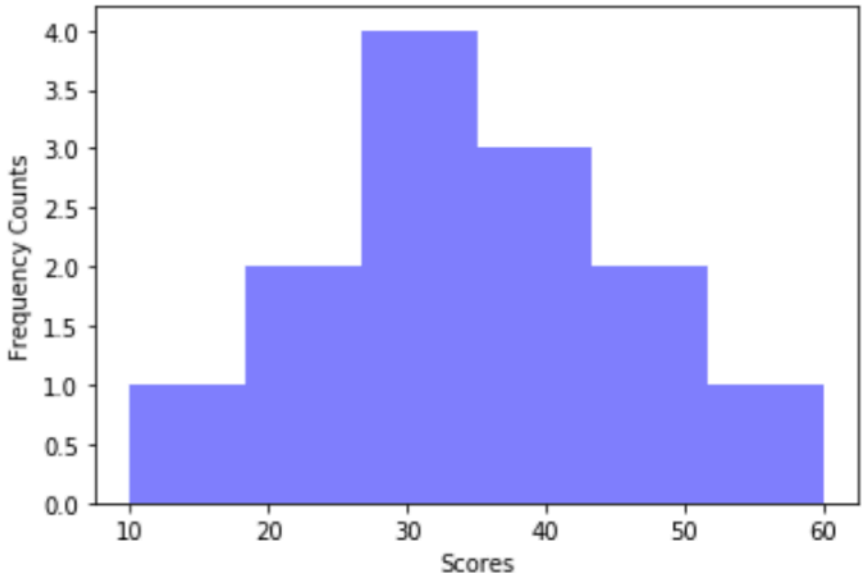
```
26 def median_calc(list):
27     sorted_list = sorted(list)
28     # print("sorted list:",sorted_list)
29     result = 0
30     position = int(round(len(sorted_list) / 2))
31     if len(sorted_list) % 2 == 0:
32         # print("even")
33         result = (sorted_list[position-1] + sorted_list[position]) / 2
34     else:
35         # print("odd")
36         result = sorted_list[position-1]
37     return(result)
38
39 def mode_calc(list):
40     list = sorted(list)
41     list_count = []
42     mode_result = []
43     highest_count = 0
44     for i in list:
45         if list.count(i) >= highest_count:
46                 list_count = i
47                 highest_count = list.count(i)
48     mode_result.append(list_count)
49     mode_result.append(highest_count)
50     return(mode_result)
51
52 def variance_calc(list):
53   average = average_calc(list)
54   variance = 0
55   for item in list:
56     # print("variance=",variance)
57     # print("average=",average)
58     # print("item=",item)
59     variance = variance + (average - item)**2
60   return variance/len(list)
61
62 def std_deviation_calc(variance):
63   return variance ** 0.5
64
```

```
average = 0
media = 0
mode = 0
list = [60,30,20,30,40,40,50,20,40,50,10,30,30,]

average = average_calc(list)
print("My computed mean:",average)
print("numpy mean:",np.mean(list))
median = median_calc(list)
print("My computed median",median)
print("numpy median:",np.median(list))
mode = mode_calc(list)
print("My computed mode=",mode[0],",count=",mode[1])
print("scipy mode:",stats.mode(list))
variance = variance_calc(list)
print("My computed variance:",variance)
print("numpy variance:",np.var(list))
std_deviation = std_deviation_calc(variance)
print("My computed standard deviation:",std_deviation)
print("numpy standard deviation:",np.std(list))
num_bins = 6
# Display histogram showing the distribution of
n, bins, patches = plt.hist(list, num_bins, facecolor='blue', alpha=0.5)
plt.xlabel('Scores')
plt.ylabel('Frequency Counts')
plt.show()
```

```
My computed mean: 34.61538461538461
numpy mean: 34.61538461538461
My computed median 30
numpy median: 30.0
My computed mode= 30 ,count= 4
scipy mode: ModeResult(mode=array([30]), count=array([4]))
My computed variance: 178.69822485207098
numpy variance: 178.69822485207098
My computed standard deviation: 13.367805536140589
numpy standard deviation: 13.367805536140589
```

```
My computed mean: 34.61538461538461
numpy mean: 34.61538461538461
My computed median 30
numpy median: 30.0
My computed mode= 30 ,count= 4
scipy mode: ModeResult(mode=array([30]), count=array([4]))
My computed variance: 178.69822485207098
numpy variance: 178.69822485207098
My computed standard deviation: 13.367805536140589
numpy standard deviation: 13.367805536140589
```



Histogram

China Air Pollution: A Case Demonstration

China Weather

- January
- February
- March
- April
- May
- June
- July
- August
- September
- October
- November
- December
- Climate Change

China Air Pollution

As one of the most popular tourist destinations in the world, China has abundant tourism resources, the vast majority of which are of great renown. However, the number of tourists visiting China declined from 2013, mostly due to its increasingly serious air pollution. Many cities have been adversely affected by hazy weather since the beginning of 2013. This is especially true of central and eastern China's cities such as Tangshan, Jinan, Zhengzhou, and Xi'an for example.

It is known to all that the rapid development of China's economy in the past 30 years resulted in environmental degradation. Nevertheless, the hazy weather records of China in 2013 attracted global attention. It is a pity that as a consequence some foreign residents choose to give up their jobs and life in China.

https://www.travelchinaguide.com/climate/air-pollution.htm

Beijing Day Tours $79
Great Wall · Forbidden City
Tiananmen · Hutong

Xian Day Tours $69
Terracotta Army · City Wall
Pagoda · Muslim Bazaar

Shanghai Day Tours $119
Zhujiajiao Town · Yu Garden
Old Street · the Bund

Search & Ticket Booking

Train Flight

From

To

AQI and Health Implications

AQI	Air Quality	Health Implications
0–50	Excellent	No air pollution.
51–100	Good	Few hypersensitive individuals should reduce the time for outdoor activities.
101–150	Lightly Polluted	Slight irritations may occur, children, and those who with breathing or heart problems should reduce outdoor exercise.
151–200	Moderately Polluted	Irritations may occur, and it may have an impact on healthy people's heart and / or respiratory system, so all people should reduce the time for outdoor exercise.
201–300	Heavily Polluted	Healthy people will be noticeably affected. People with breathing or heart problems will lack exercise tolerance. Those patients, children and elders should remain indoors.
300+	Severely Polluted	Even healthy people will lack endurance during activities. There may be strong irritations and symptoms. So all people should avoid outdoor activities.

Air Quality Rankings of Chinese Major Tourist Cities in 2020

Rank	City	Province belongs to	AQI	Air Quality Level	PM 2.5	PM 10
1	Sanya	Hainan	22	Excellent	10	22
2	Lijiang	Yunnan	24	Excellent	12	14
3	Dali	Yunnan	24	Excellent	15	18
4	Changsha	Hunan	25	Excellent	17	10
5	Haikou	Hainan	28	Excellent	15	28
6	Qingdao	Shandong	29	Excellent	19	29
7	Kunming	Yunnan	33	Excellent	23	30

Search & Ticket Booking

Train Flight

From

To

Date

Search

Train service fee low to $5.49 per ticket

Air Quality Rankings of Chinese Major Tourist Cities in 2020

Rank	City	Province belongs to	AQI	Air Quality Level	PM 2.5	PM 10
1	Sanya	Hainan	22	Excellent	10	22
2	Lijiang	Yunnan	24	Excellent	12	14
3	Dali	Yunnan	24	Excellent	15	18
4	Changsha	Hunan	25	Excellent	17	10
5	Haikou	Hainan	28	Excellent	15	28
6	Qingdao	Shandong	29	Excellent	19	29
7	Kunming	Yunnan	33	Excellent	23	30
8	Lhasa	Tibet	34	Excellent	15	34
9	Dalian	Liaoning	43	Excellent	20	43
10	Guangzhou	Guangdong	44	Excellent	30	44
11	Guiyang	Guizhou	45	Excellent	31	44
12	Zhangjiakou	Hebei	49	Excellent	31	49
13	Shenzhen	Guangdong	55	Good	39	57
14	Zhangjiajie	Hunan	63	Good	45	51
15	Fuzhou	Fujian	65	Good	47	51
16	Nanning	Guangxi	68	Good	49	57
17	Guilin	Guangxi	69	Good	50	55
18	Beijing	Beijing	75	Good	55	66
19	Xining	Qinghai	75	Good	55	74
20	Chongqing	Chongqing	80	Good	59	80
21	Lanzhou	Gansu	82	Good	60	95
22	Yangzhou	Jiangsu	84	Good	62	78

300	Polluted	exercise tolerance. Those patients, children and elders should remain indoors.
300+	Severely Polluted	Even healthy people will lack endurance during activities. There may be strong irritations and symptoms. So all people should avoid outdoor activities.

Air Quality Rankings of Chinese Major Tourist Cities in 2020

Rank	City	Province belongs to	AQI	Air Quality Level	PM 2.5	PM 10
1	Sanya	Hainan	22	Excellent	10	22
2	Lijiang	Yunnan	24	Excellent	12	14
3	Dali	Yunnan	24	Excellent	15	18
4	Changsha	Hunan	25	Excellent	17	10
5	Haikou	Hainan	28	Excellent	15	28

Search & Ticket Booking
Train Flight
From

TOP

Elements Console Sources Network Performance Memory Application Security Audits

<table class="c_tableX">
<tbody>
<tr>
<th style="text-align: center;">Rank</th> == $0
<th style="text-align: center;">City</th>
<th style="text-align: center;">Province belongs to</th>
<th style="text-align: center;">AQI</th>
<th style="text-align: center;">Air Quality Level</th>
<th style="text-align: center;">PM 2.5</th>
<th style="text-align: center;">PM 10</th>

body #main div div #zoom #artFlag div table.c_tableX tbody tr th

Styles Computed Event Listeners DOM Breakpoints »
Filter :hov .cls +
element.style {
text-align: center;
}
.table1 th, .table1 td, .c_tableX th, .c_tableX td { air-pollution.htm:1945
text-align: center;
}
.c_tableX th { air-pollution.htm:359
background: #efefef !important;
font-weight: normal;

Search & Ticket Booking

Train | Flight

From

300	Polluted	exercise tolerance. Those patients, children and elders should remain indoors.
300+	Severely Polluted	Even healthy people will lack endurance during activities. There may be strong irritations and symptoms. So all people should avoid outdoor activities.

Air Quality Rankings of Chinese Major Tourist Cities in 2020

Rank	City	Province belongs to	AQI	Air Quality Level	PM 2.5	PM 10
1	Sanya	Hainan	22	Excellent	10	22
2	Lijiang	Yunnan	24	Excellent	12	14
3	Dali	Yunnan	24	Excellent	15	18
4	Changsha	Hunan	25	Excellent	17	10
5	Haikou	Hainan	28	Excellent	15	28

TOP

Elements | Console | Sources | Network | Performance | Memory | Application | Security | Audits

```
<th style="text-align: center;">PM 2.5</th>
<th style="text-align: center;">PM 10</th>
</tr>
▼<tr>
<td style="text-align: center;">1</td>
<td style="text-align: center;">Sanya</td>
<td style="text-align: center;">Hainan</td>
<td style="text-align: center;">22</td>
<td style="text-align: center;">Excellent</td>
<td style="text-align: center;">10</td>
<td style="text-align: center;">22</td>
</tr>
```

body #main div div #zoom #artFlag div table.c_tableX tbody tr th

Styles | Computed | Event Listeners | DOM Breakpoints | »

Filter :hov .cls +

}

.table1 th, .table1 td, .c_tableX th, .c_tableX td { air-pollution.htm:1945
text-align: center;
}

.c_tableX th { air-pollution.htm:359
background: ▸ #efefef !important;
font-weight: normal;
}

Presenting China Pollution Figures Using Beautiful Soup, Pandas and Matplotlib

```
from bs4 import BeautifulSoup
import requests
import csv
import pandas as pd
import matplotlib.pyplot as plt

# Fetch URL
html_page = requests.get('https://www.travelchinaguide.com/climate/air-pollution.htm')
# Obtain the entire HTML page
soup = BeautifulSoup(html_page.content,'html.parser')
# Find all the HTML tables
tables = soup.find_all(class_="c_tableX")
# Access the second HTML table (i.e. tables[1] instrad of tablrs[0]) that contains Air Quality information
# with both table header and table data
table = tables[1] # assign 2nd table in the table list to variable called "table"
table_header = table.find_all('th') # extract table header based on the 'th' tag
# print table header tags and texts
# print(table_header)
header = []
data = []
for th in table.find_all('th'):
    # print header text
    # print(th.text)
    header.append(th.text)
# print list of headers
# print(header)
all_rows = table.find_all("tr") # extract all the table rows
```

```
# Enumerate all the rows to extract needed data
# Obtain both index and values using the enumerate function starting at 2nd row
for i, row in enumerate(all_rows,1):
        if (i < len(all_rows)): # i will start at 0 and stop when i equals the number of rows already processed
            # for each row find all the "td" (i.e. table column) element that holds the data
            tds = all_rows[i].find_all("td")
             # enumerate each column to extract the "td" value
            for j, td in enumerate(tds,1):
                # print the column name and value
                # print (j,td.text)
                if j==1:
                    rank = td.text
                if j==2:
                    city = td.text
                if j==3:
                    province = td.text
                if j==4:
                    aqi = td.text
                if j==5:
                    air_quality = td.text
                if j==6:
                    pm2_5 = td.text
                if j==7:
                    pm10 = td.text
            # append the all the column values to the data list
            data.append([rank,city,province,aqi,air_quality,pm2_5,pm10])
# Assign row data and column headers to dataframe
df = pd.DataFrame(data,
  columns = header
)
# Display data frame with both header and data
# df
# Save dataframe to external csv file
df.to_csv('projects/china_air_quality.csv', sep='\t', encoding='utf-8')
```

```
# open csv file and read csv data into Pandas dataframe
df = pd.read_csv("projects/china_air_quality.csv",sep='\t', encoding='utf-8')
df
# Set column headings for entire air quality table and print out the entire table
air_quality_ranking = df[['Rank','City','Province belongs to','AQI','Air Quality Level','PM 2.5','PM 10']]
# air_quality_ranking = pd.DataFrame(df.values[1:], columns=df.values[0])
print("Air Quality Ranking\n")
print(air_quality_ranking)
# Extract cities that are polluted
lightly_polluted    = df[df['Air Quality Level'] == 'Lightly Polluted']
moderately_polluted = df[df['Air Quality Level'] == 'Moderately Polluted']
heavily_polluted    = df[df['Air Quality Level'] == 'Heavely Polluted']
# Combine the cities of different pollution level into one table
selected = lightly_polluted.append(moderately_polluted).append(heavily_polluted)
pc = selected[['Rank','City','AQI','Air Quality Level','PM 2.5','PM 10']]
```

```
# ////////////////////////////////////////////////////////////////
# Display Cities with Pollution
# ////////////////////////////////////////////////////////////////
print("\nCities with Pollution\n")
print(pc)
cities = pc['City'].tolist()
aqi_lvl = pc['AQI'].tolist()
pm_2_5 = pc['PM 2.5'].tolist()
ax = plt.subplot()
plt.bar(range(len(cities)),aqi_lvl)
# Create ax object here
j = 0
ax_list = []
while j<len(cities):
    ax_list.append(j)
    j += 1
ax.set_xticks(ax_list)
plt.xlabel('Cities')
plt.ylabel('AQI Levels')
plt.title('Cities with High AQI Levels')
ax.set_xticklabels(cities, rotation=70)
plt.show()
```

Air Quality Ranking

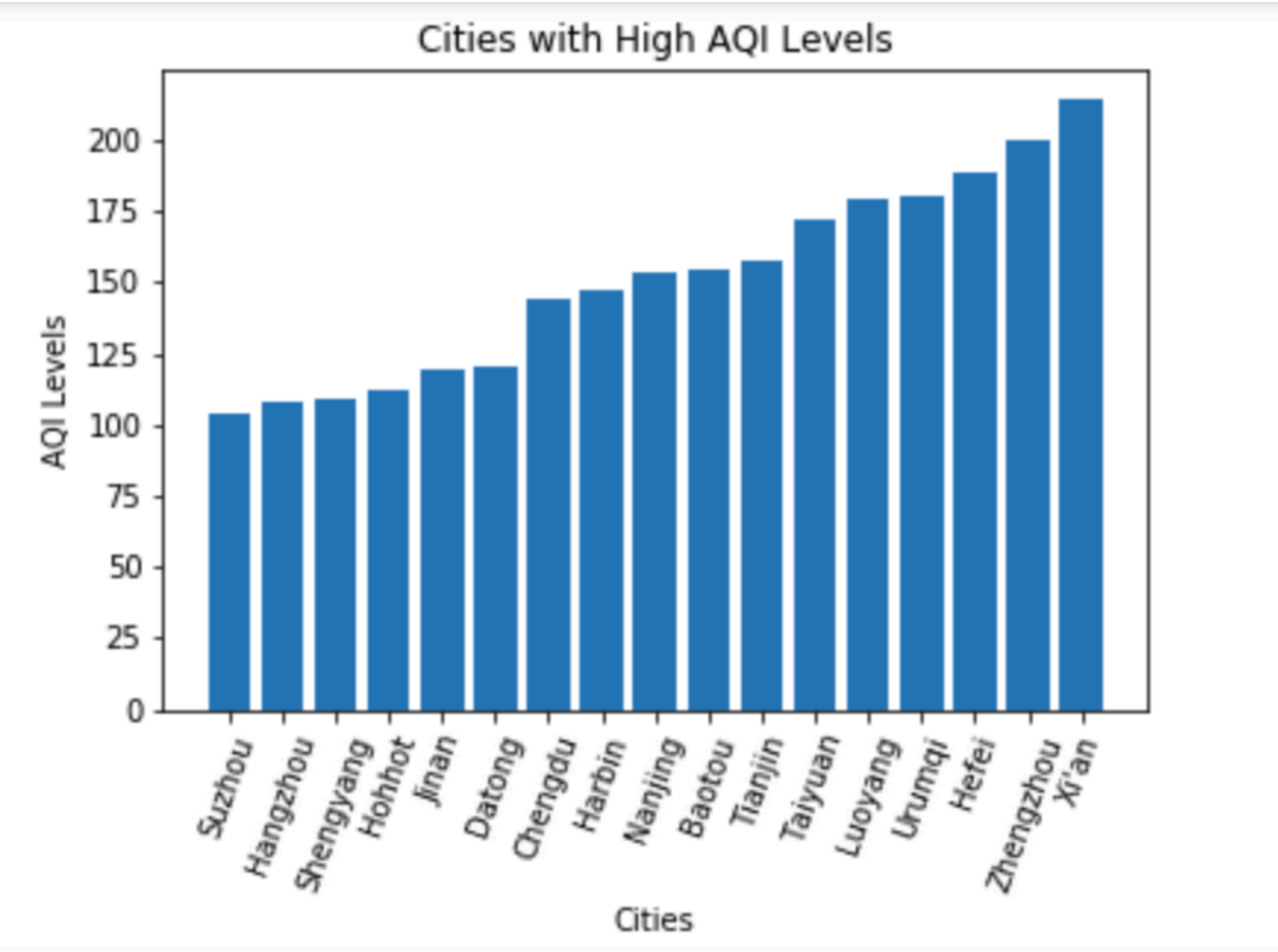
	Rank	City	Province belongs to	AQI	Air Quality Level	PM 2.5
0	1	Sanya	Hainan	22	Excellent	10
1	2	Lijiang	Yunnan	24	Excellent	12
2	3	Dali	Yunnan	24	Excellent	15
3	4	Changsha	Hunan	25	Excellent	17
4	5	Haikou	Hainan	28	Excellent	15
5	6	Qingdao	Shandong	29	Excellent	19
6	7	Kunming	Yunnan	33	Excellent	23
7	8	Lhasa	Tibet	34	Excellent	15
8	9	Dalian	Liaoning	43	Excellent	20
9	10	Guangzhou	Guangdong	44	Excellent	30
10	11	Guiyang	Guizhou	45	Excellent	31
11	12	Zhangjiakou	Hebei	49	Excellent	31
12	13	Shenzhen	Guangdong	55	Good	39
13	14	Zhangjiajie	Hunan	63	Good	45
14	15	Fuzhou	Fujian	65	Good	47
15	16	Nanning	Guangxi	68	Good	49

16	17	Guilin	Guangxi	69	Good	50
17	18	Beijing	Beijing	75	Good	55
18	19	Xining	Qinghai	75	Good	55
19	20	Chongqing	Chongqing	80	Good	59
20	21	Lanzhou	Gansu	82	Good	60
21	22	Yangzhou	Jiangsu	84	Good	62
22	23	Changchun	Jilin	85	Good	63
23	24	Yinchuan	Ningxia	89	Good	66
24	25	Xiamen	Fujian	90	Good	67
25	26	Wuhan	Hubei	97	Good	72
26	27	Shanghai	Shanghai	97	Good	72
27	28	Shijiazhuang	Hebei	98	Good	73
28	29	Suzhou	Jiangsu	104	Lightly Polluted	78
29	30	Hangzhou	Zhejiang	108	Lightly Polluted	81
30	31	Shengyang	Liaoning	109	Lightly Polluted	89
31	32	Hohhot	Inner Mongolia	112	Lightly Polluted	84
32	33	Jinan	Shandong	119	Lightly Polluted	90
33	34	Datong	Shanxi	120	Lightly Polluted	91
34	35	Chengdu	Sichuan	144	Lightly Polluted	110
35	36	Harbin	Heilongjiang	147	Lightly Polluted	112

Cities with Pollution

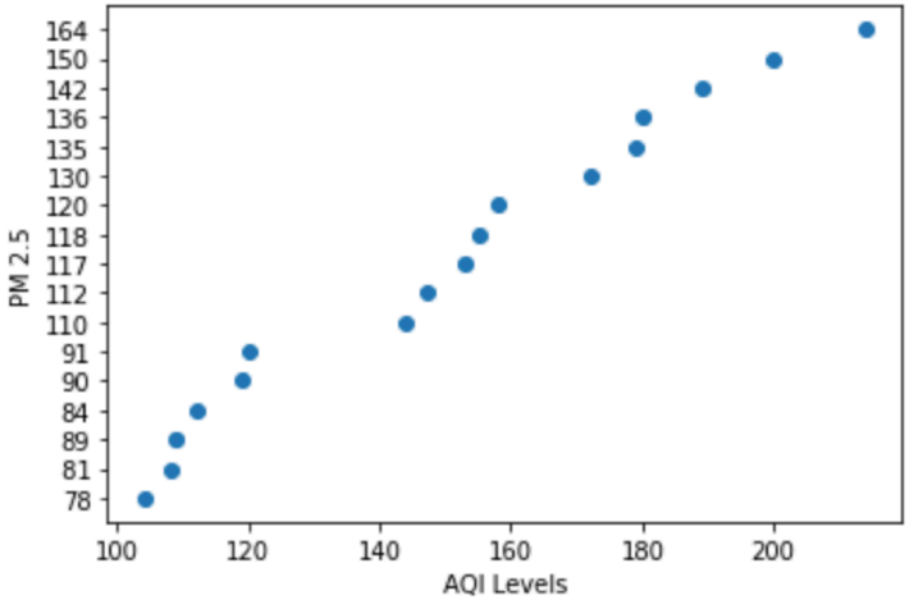
	Rank	City	AQI	Air Quality Level	PM 2.5	PM 10
28	29	Suzhou	104	Lightly Polluted	78	88
29	30	Hangzhou	108	Lightly Polluted	81	116
30	31	Shengyang	109	Lightly Polluted	89	112
31	32	Hohhot	112	Lightly Polluted	84	93
32	33	Jinan	119	Lightly Polluted	90	125
33	34	Datong	120	Lightly Polluted	91	172
34	35	Chengdu	144	Lightly Polluted	110	125
35	36	Harbin	147	Lightly Polluted	112	97
36	37	Nanjing	153	Moderately Polluted	117	144
37	38	Baotou	155	Moderately Polluted	118	146
38	39	Tianjin	158	Moderately Polluted	120	125
39	40	Taiyuan	172	Moderately Polluted	130	182
40	41	Luoyang	179	Moderately Polluted	135	166
41	42	Urumqi	180	Moderately Polluted	136	138
42	43	Hefei	189	Moderately Polluted	142	100
43	44	Zhengzhou	200	Moderately Polluted	150	184

26	27	Shanghai	Shanghai	97	Good	72
27	28	Shijiazhuang	Hebei	98	Good	73
28	29	Suzhou	Jiangsu	104	Lightly Polluted	78
29	30	Hangzhou	Zhejiang	108	Lightly Polluted	81
30	31	Shengyang	Liaoning	109	Lightly Polluted	89
31	32	Hohhot	Inner Mongolia	112	Lightly Polluted	84
32	33	Jinan	Shandong	119	Lightly Polluted	90
33	34	Datong	Shanxi	120	Lightly Polluted	91
34	35	Chengdu	Sichuan	144	Lightly Polluted	110
35	36	Harbin	Heilongjiang	147	Lightly Polluted	112
36	37	Nanjing	Jiangsu	153	Moderately Polluted	117
37	38	Baotou	Inner Mongolia	155	Moderately Polluted	118
38	39	Tianjin	Tianjin	158	Moderately Polluted	120
39	40	Taiyuan	Shanxi	172	Moderately Polluted	130
40	41	Luoyang	Henan	179	Moderately Polluted	135
41	42	Urumqi	Xinjiang	180	Moderately Polluted	136
42	43	Hefei	Anhui	189	Moderately Polluted	142
43	44	Zhengzhou	Henan	200	Moderately Polluted	150
44	45	Xi'an	Shaanxi	214	Heavely Polluted	164

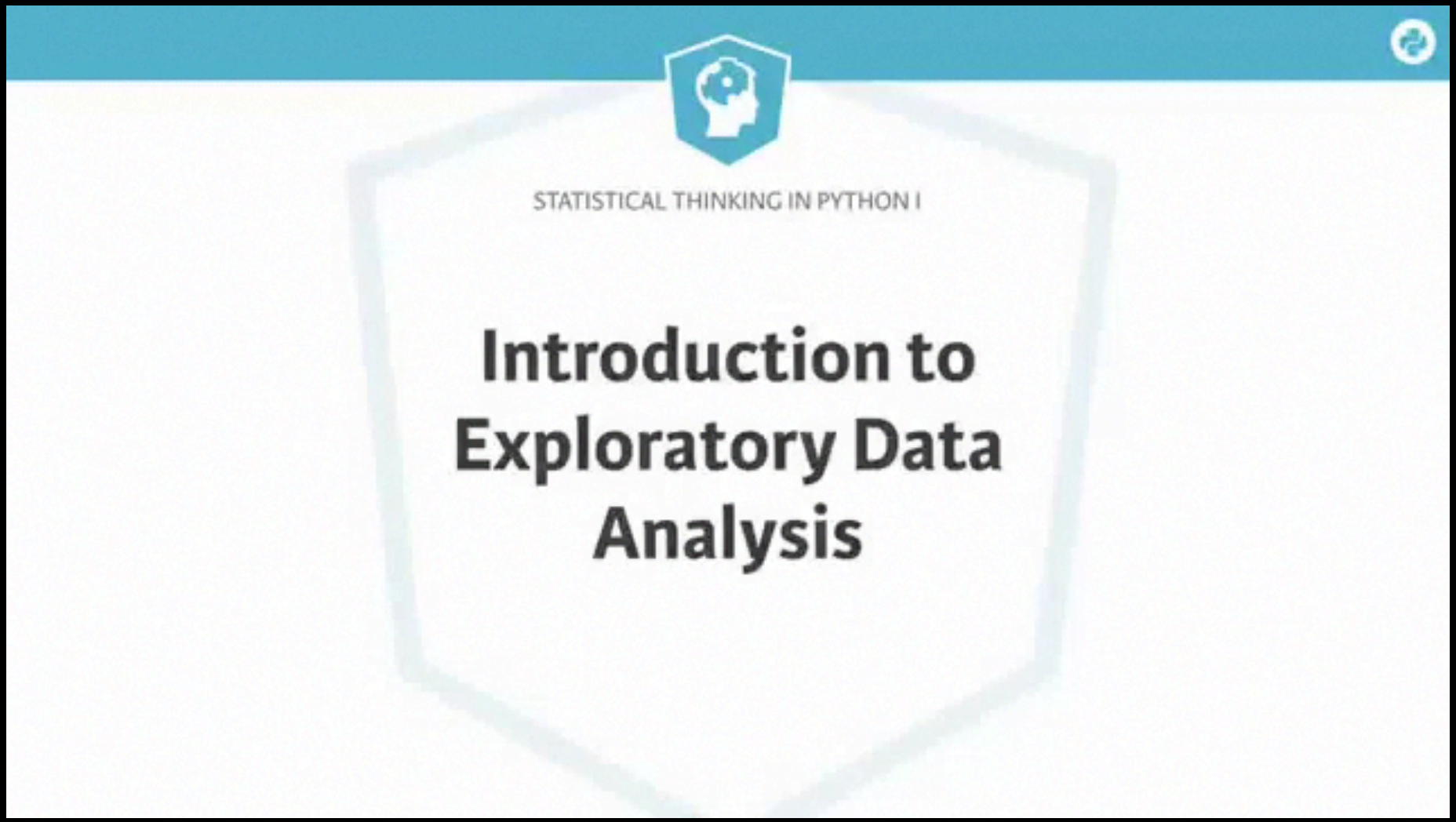

Cities with High AQI Levels
AQI Levels
0
25
50
75
100
125
150
175
200
Suzhou
Hangzhou
Shengyang
Hohhot
Jinan
Datong
Chengdu
Harbin
Nanjing
Baotou
Tianjin
Taiyuan
Luoyang
Urumqi
Hefei
Zhengzhou
Xi'an
Cities

Bar chart

```
1  from matplotlib import pyplot as plt
2  # aqi_lvl = pc['AQI'].tolist()
3  # pm_2_5 = pc['PM 2.5'].tolist()
4  ax = plt.subplot()
5  plt.plot(aqi_lvl, pm_2_5, 'o')
6  plt.ylabel('PM 2.5')
7  plt.xlabel('AQI Levels')
8  plt.show()
```



Scatter Plots



https://www.youtube.com/watch?v=fPkIFa9uZ9A

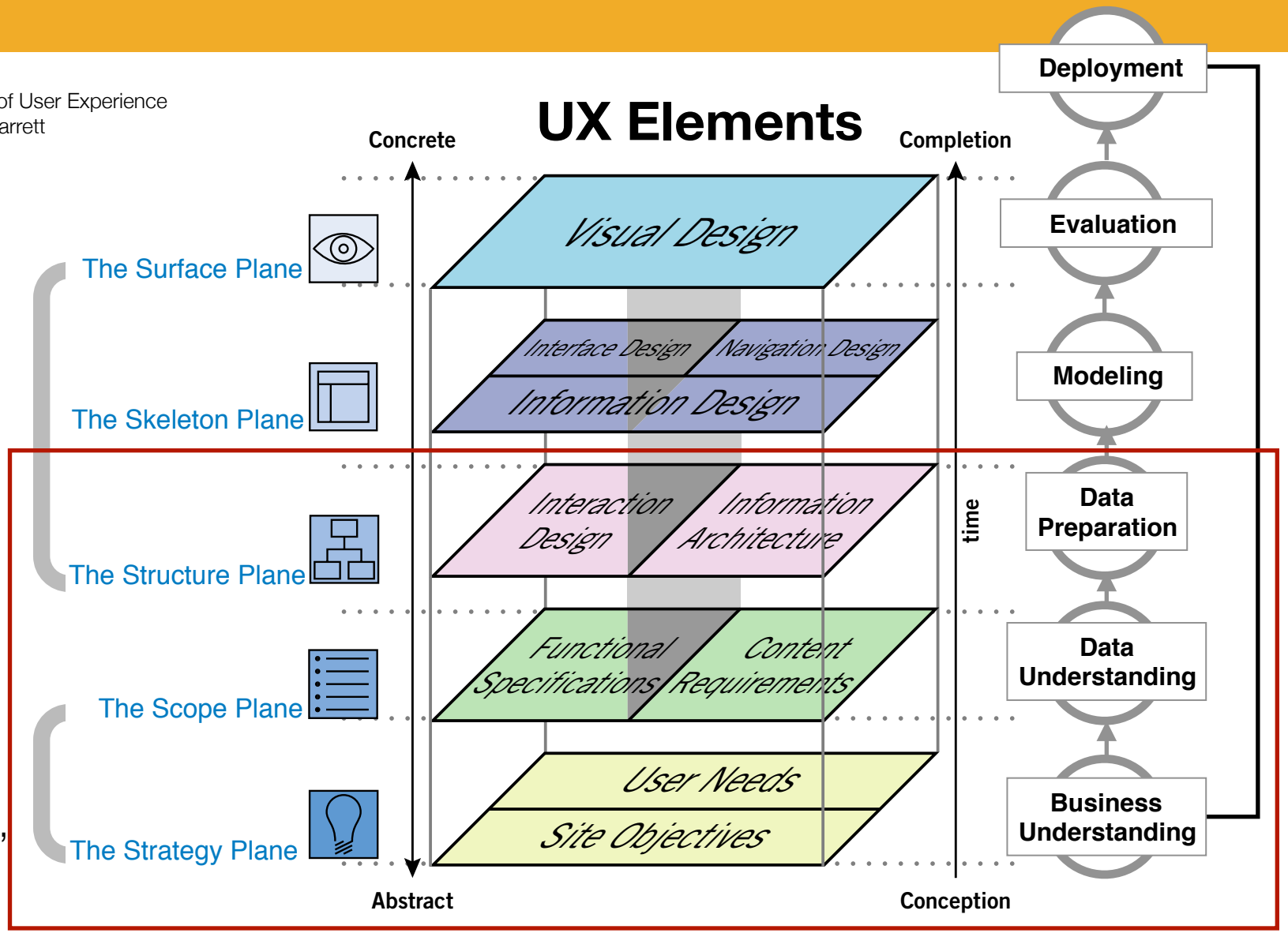
CRISP-DM Framework and Methodology Map

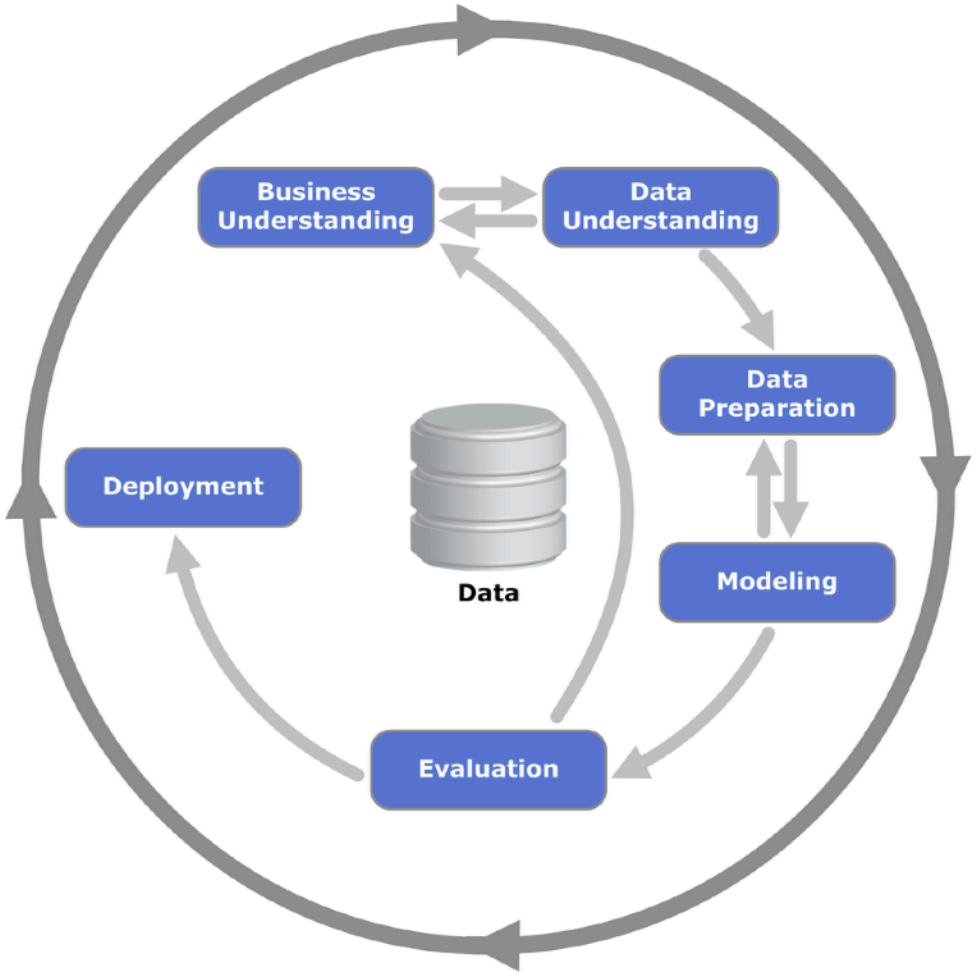
Supervised, Unsupervised and Reinforcement Learning

Applications of ML as a Black Box

CRISP-DM as a Problem Solving Framework

Source: Elements of User Experience
by Jesse James Garrett





CRISP-DM Framework

Source: Wikipedia

https://www.youtube.com/watch?v=CRKn-9gVNBw

Cross-industry standard process for data mining, known as **CRISP-DM**, is an open standard process model that describes common approaches used by data mining experts. It is the most widely-used analytics model.

Source: Wikipedia

Why is it important to learn the framework?

Harvard Business Review

DATA

Data Scientist: The Sexiest Job of the 21st Century

by Thomas H. Davenport and D.J. Patil

From the October 2012 Issue

https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century

What is Data Science?

https://www.youtube.com/watch?v=X3paOmcrTjQ

What does a data scientist do?

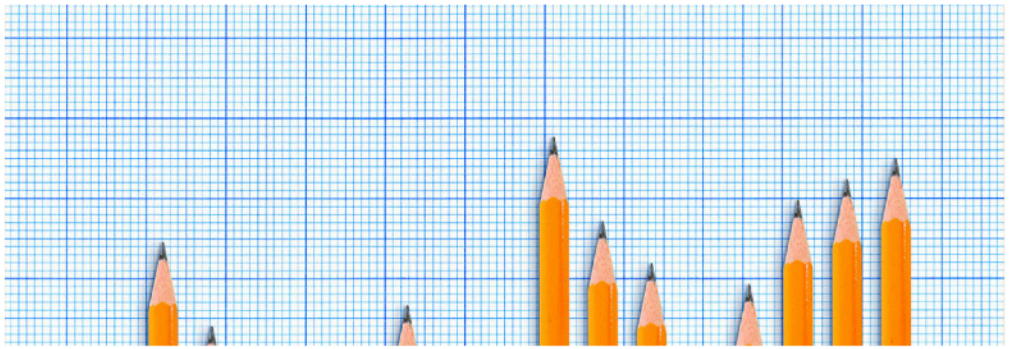
ANALYTICS

What Data Scientists Really Do, According to 35 Data Scientists

by Hugo Bowne-Anderson

August 15, 2018

Summary Save Share 8 Comment Print $8.95 Buy Copies



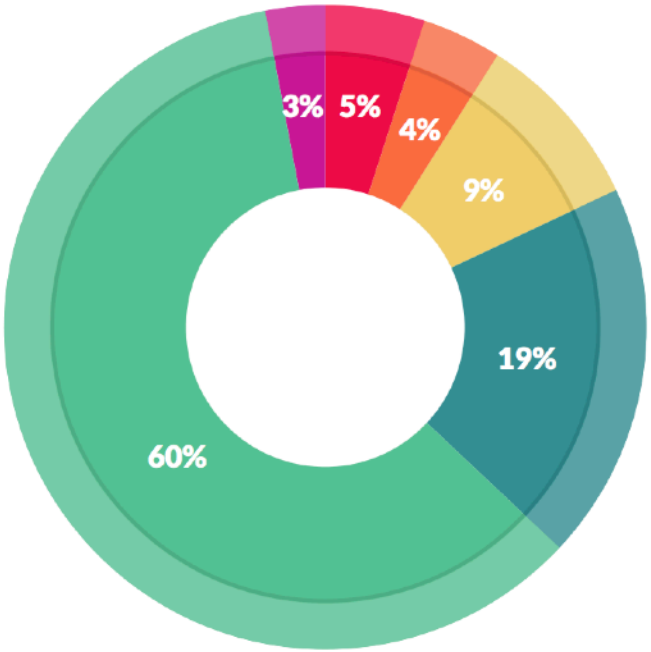
https://hbr.org/2018/08/what-data-scientists-really-do-according-to-35-data-scientists?referral=03758&cm_vc=rr_item_page.top_right

Source: By Hugo Bowen-Anderson
August 15, 2018
Harvard Business Review

What data scientists do. We now know how data science works, at least in the tech industry. First, data scientists lay a solid data foundation in order to perform robust analytics. Then they use online experiments, among other methods, to achieve sustainable growth. Finally, they build machine learning pipelines and personalized data products to better understand their business and customers and to make better decisions. In other words, in tech, data science is about infrastructure, testing, machine learning for decision making, and data products.

How a Data Scientist Spends Their Day

Here's where the popular view of data scientists diverges pretty significantly from reality. Generally, we think of data scientists building algorithms, exploring data, and doing predictive analysis. That's actually not what they spend most of their time doing, however.



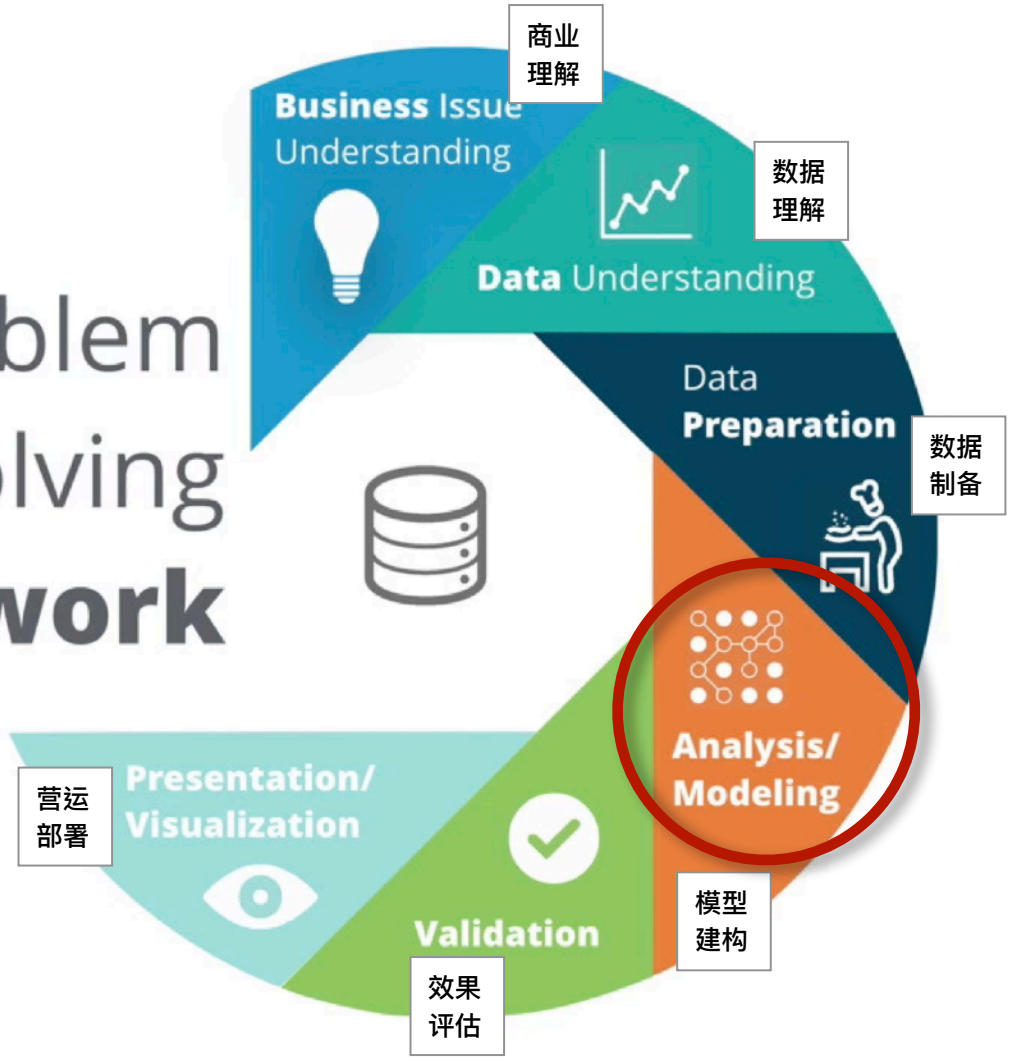
What data scientists spend the most time doing

- *Building training sets: 3%*
- *Cleaning and organizing data: 60%*
- *Collecting data sets; 19%*
- *Mining data for patterns: 9%*
- *Refining algorithms: 4%*
- *Other: 5%*

Source: Data Science 2016 Report by CrowdFlower

https://www.youtube.com/watch?v=bdHtEIHKfAo

Problem Solving **Framework**

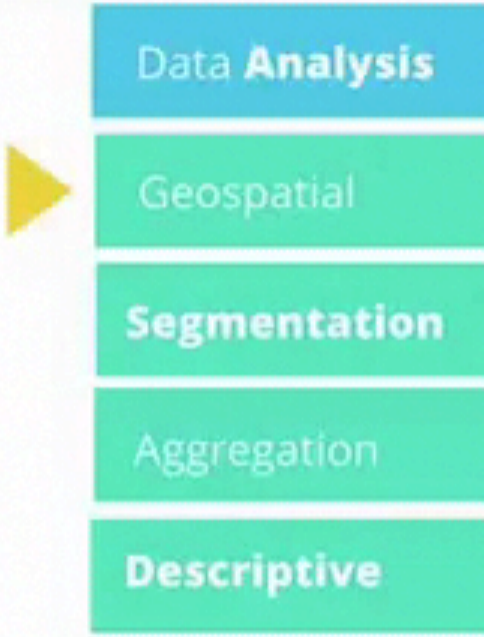


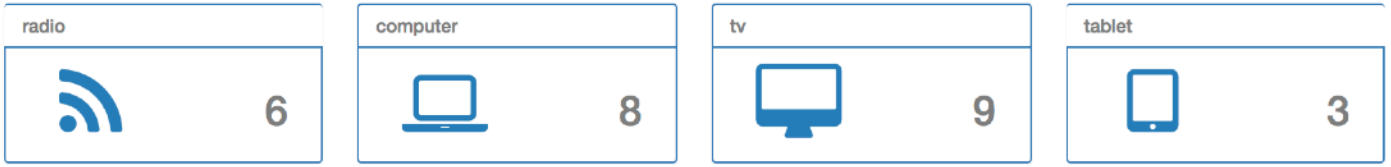
Business Problem

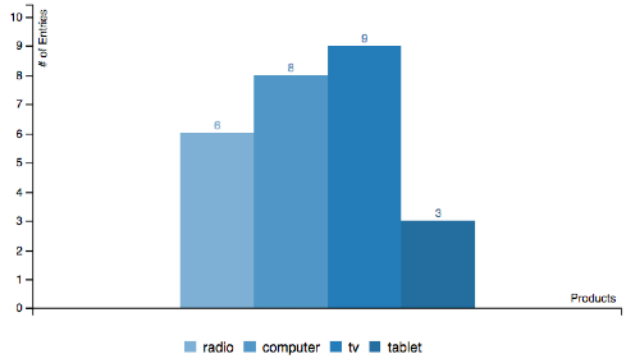
Predict Outcome				Data Analysis
Data Rich			Data Poor	Geospatial
Numeric		Classification	A/B Test	Segmentation
Continuous	Count	Binary	Non Binary	Aggregation
• Linear Regression • Decision Tree • Forest Model • Boosted Model	• Count Regression	• Logistic Regression • Decision Tree	• Forest Model • Booted Model	Descriptive

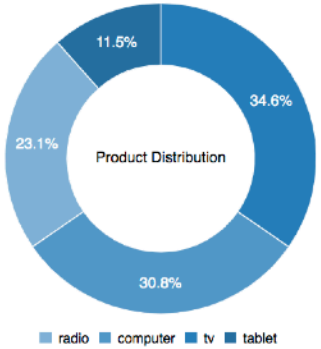
Source: Udacity Model Selection Methodology Map

Types of **non-predictive** data analysis





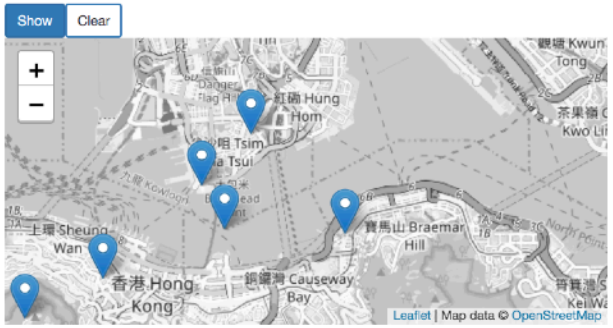




Show 10 entries Search:

Name	Total Entries
computer	8
radio	6
tablet	3
tv	9

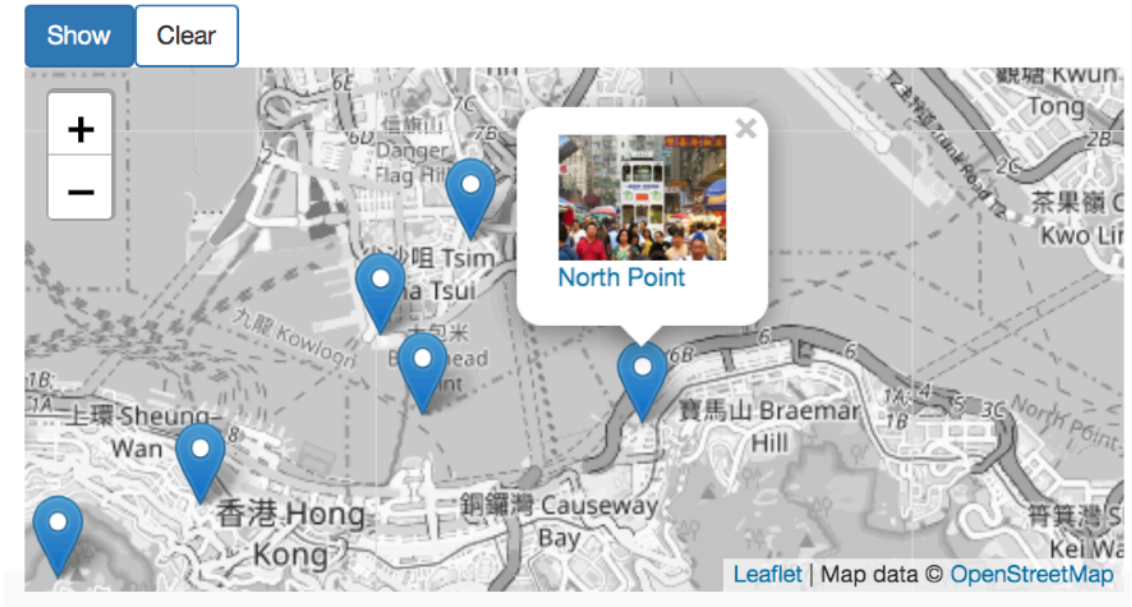
Showing 1 to 4 of 4 entries Previous 1 Next



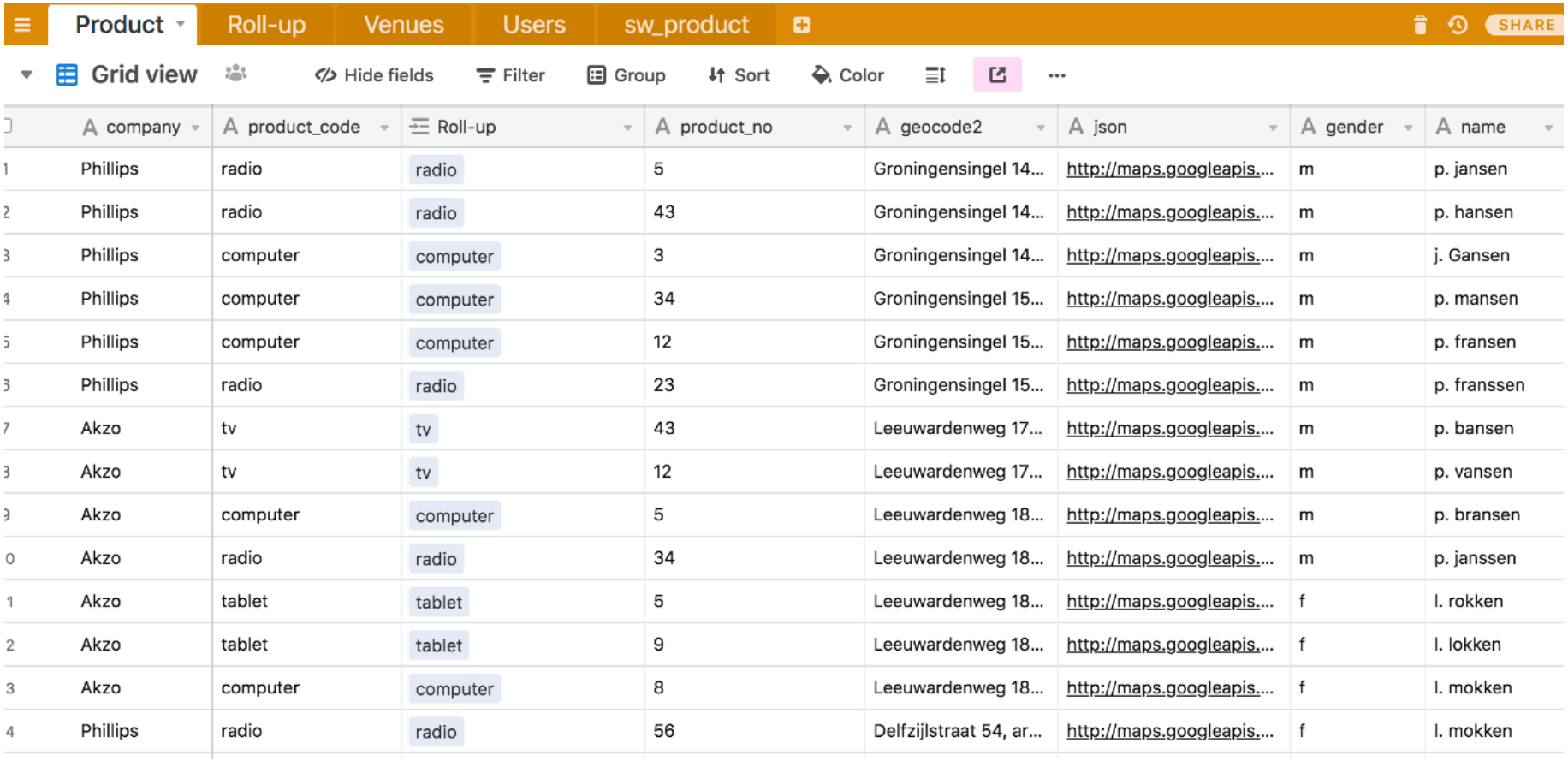
Product | Roll-up | Venues | Users | sw_product

Grid view | Hide fields | Filter | Group | Sort

	Name	Lat	Lng	url	Pic	img_url
1	North Point	22.287111	114.191667	https://en.wikipedia.org/...		https://dl.airtab
2	Mong Kok	22.322500	114.170556	https://en.wikipedia.org/...		https://dl.airtab
3	Happy Valley	22.266667	114.183333	https://en.wikipedia.org/...		https://dl.airtab
4	Victoria Peak	22.275469	114.143828	https://en.wikipedia.org/...		https://dl.airtab
5	Lan Kwai Fong	22.280972	114.155528	https://en.wikipedia.org/...		https://dl.airtab
6	Choi Hung	22.334484	114.210024	https://en.wikipedia.org/...		https://dl.airtab
7	HKBU	22.338936	114.181953	https://en.wikipedia.org/...		https://dl.airtab
8	CUHK	22.418498	114.204074	https://en.wikipedia.org/...		https://dl.airtab
9	HKU	22.284167	114.137778	https://en.wikipedia.org/...		https://dl.airtab
10	HK Science Musuem	22.301000	114.177655	https://en.wikipedia.org/...		https://dl.airtab
11	HK Cultural Center	22.293850	114.170323	https://en.wikipedia.org/...		https://dl.airtab
12	Victoria Harbour	22.287753	114.173619	https://en.wikipedia.org/...		https://dl.airtab

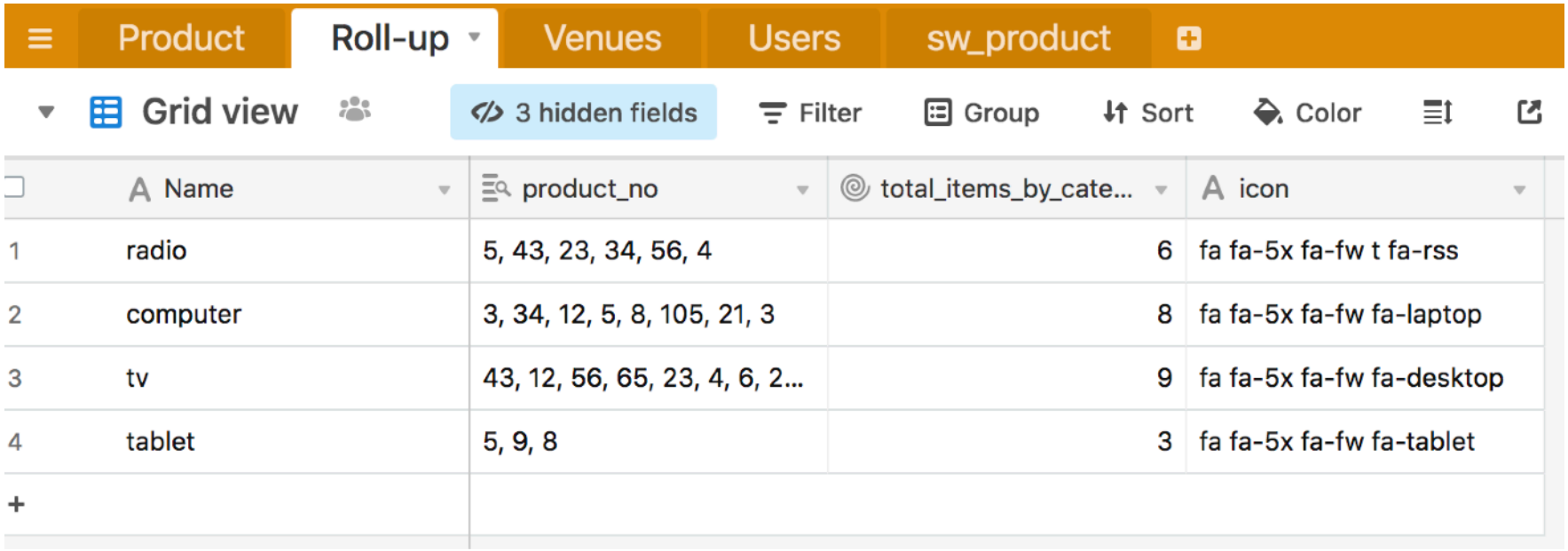

Show
Clear
North Point
Danger Flag Hill
Tsim Sha Tsui
Kowloon
上環 Sheung-Wan
香港 Hong Kong
銅鑼灣 Causeway Bay
寶馬山 Braemar Hill
觀塘 Kwun Tong
Kwo Li
Kei Wa
Leaflet | Map data © OpenStreetMap

Geospatial



	company	product_code	Roll-up	product_no	geocode2	json	gender	name
1	Phillips	radio	radio	5	Groningensingel 14...	http://maps.googleapis....	m	p. jansen
2	Phillips	radio	radio	43	Groningensingel 14...	http://maps.googleapis....	m	p. hansen
3	Phillips	computer	computer	3	Groningensingel 14...	http://maps.googleapis....	m	j. Gansen
4	Phillips	computer	computer	34	Groningensingel 15...	http://maps.googleapis....	m	p. mansen
5	Phillips	computer	computer	12	Groningensingel 15...	http://maps.googleapis....	m	p. fransen
6	Phillips	radio	radio	23	Groningensingel 15...	http://maps.googleapis....	m	p. franssen
7	Akzo	tv	tv	43	Leeuwardenweg 17...	http://maps.googleapis....	m	p. bansen
8	Akzo	tv	tv	12	Leeuwardenweg 17...	http://maps.googleapis....	m	p. vansen
9	Akzo	computer	computer	5	Leeuwardenweg 18...	http://maps.googleapis....	m	p. bransen
10	Akzo	radio	radio	34	Leeuwardenweg 18...	http://maps.googleapis....	m	p. janssen
11	Akzo	tablet	tablet	5	Leeuwardenweg 18...	http://maps.googleapis....	f	l. rokken
12	Akzo	tablet	tablet	9	Leeuwardenweg 18...	http://maps.googleapis....	f	l. lokken
13	Akzo	computer	computer	8	Leeuwardenweg 18...	http://maps.googleapis....	f	l. mokken
14	Phillips	radio	radio	56	Delfzijlstraat 54, ar...	http://maps.googleapis....	f	l. mokken

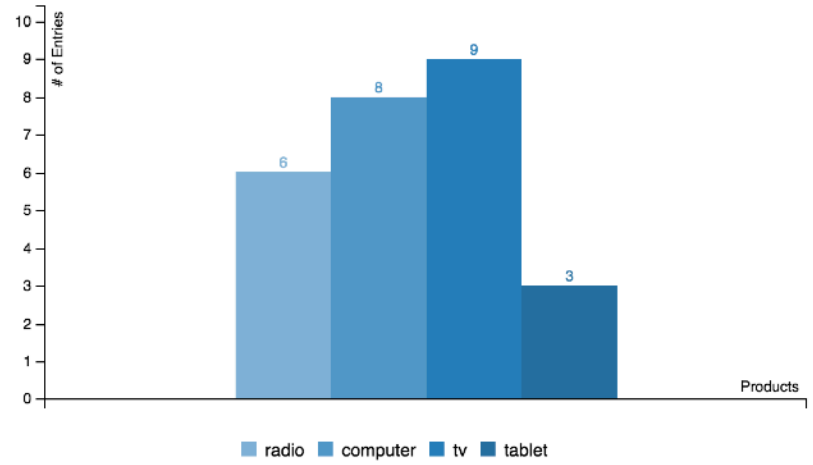
Segmentation and Aggregation

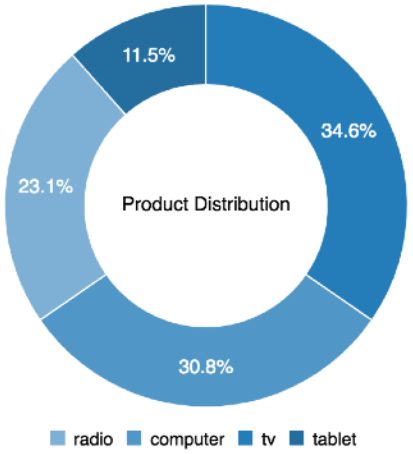


	Name	product_no	total_items_by_cate...	icon
1	radio	5, 43, 23, 34, 56, 4	6	fa fa-5x fa-fw t fa-rss
2	computer	3, 34, 12, 5, 8, 105, 21, 3	8	fa fa-5x fa-fw fa-laptop
3	tv	43, 12, 56, 65, 23, 4, 6, 2...	9	fa fa-5x fa-fw fa-desktop
4	tablet	5, 9, 8	3	fa fa-5x fa-fw fa-tablet
+				

Segmentation and Aggregation

radio	computer	tv	tablet
6	8	9	3





Business Problem

Predict Outcome				Data Analysis
Data Rich			Data Poor	Geospatial
Numeric		Classification	A/B Test	Segmentation
Continuous	Count	Binary	Non Binary	Aggregation
• Linear Regression • Decision Tree • Forest Model • Boosted Model	• Count Regression	• Logistic Regression • Decision Tree	• Forest Model • Booted Model	Descriptive

Source: Udacity Model Selection Methodology Map

Predictive Analytics/Machine Learning/Deep Learning
预测分析/机器学习/深度学习

- Linear Regression
- Decision Tree
- Forest Model
- Boosted Model

- Count Regression

- Logistic Regression
- Decision Tree

- Forest Model
- Booted Model

Inferential Statistics & Hypothe-sis Testing
推论统计与假设检验

Explorative Data Analysis & Descriptive Statistics探索性数据分析与描述性统计

Source: Udacity Model Selection Methodology Map

Inferential Statistics vs. Machine Learning

Business **Problem**

Predict Outcome

Data **Rich**

Data **Poor**

Comparison Between IS & ML

	Inferential Statistics (IS)	Machine Learning (ML)
Hypothesis Driven	X	
Inference from Sample	X	
Model Training		X
Prediction Focus		X
Interpretation Focus	X	
Data Rich		X

A/B Test is similar to hypothesis testing in Inferential Statistics in the sense that there will be comparison between outcome from a control group and outcome from an experimental group.

In a T-test, for instance, we can find out whether the null hypothesis can be rejected as a result of $p < .05$ or $p < .01$ that the difference between the groups is significant. Random sampling has to be observed in group assignment. The two groups have different independent variables.

CRISP-DM Framework and Methodology Map

Supervised, Unsupervised and Reinforcement Learning

Applications of ML as a Black Box

What is Machine Learning? How is it related to Predictive Analytics and Deep Learning?

“Machine Learning is a sub-field of Artificial Intelligence used in programming the computers to learn on its own from data fed to it. The data can be labelled, unlabelled and environmentally triggered through reinforced interactions.”

Supervised Learning

- Used for prediction of categorical and numerical outcome.
- Data has to be labelled and separated into training set and testing set before model building.
- Apply different algorithms and evaluate which one has best fit.

Unsupervised Learning

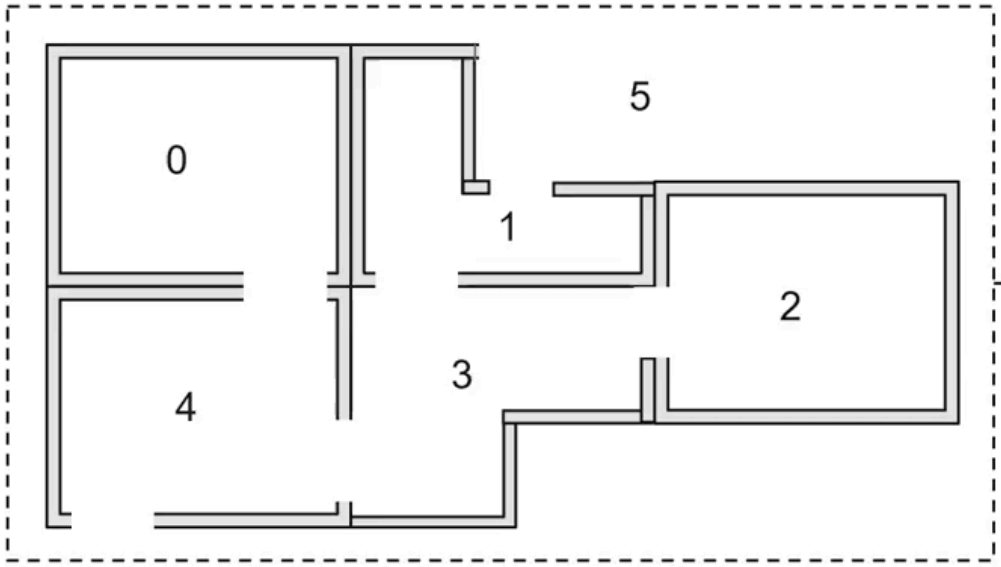
- Used in the exploratory stage of data preparation to find out patterns (clusters).
- Data is not labelled.
- Used in performing dimension reduction to help extract the essential features for preparing datasets used in model building.

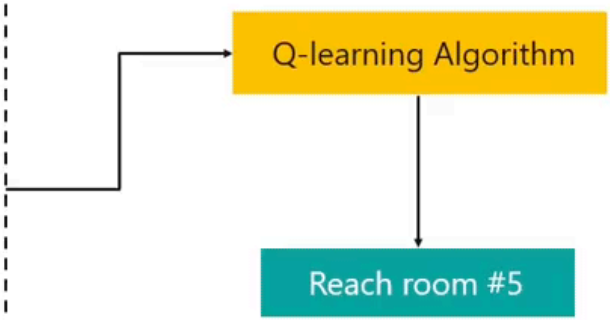
Reinforcement Learning

- The ML program is turned into a software agent, navigating through a problem space to reach a goal by trial and error.
- Throughout the course of interaction with the environment, feed-backs will be given to steer the agent toward the goal.

Use Case 5

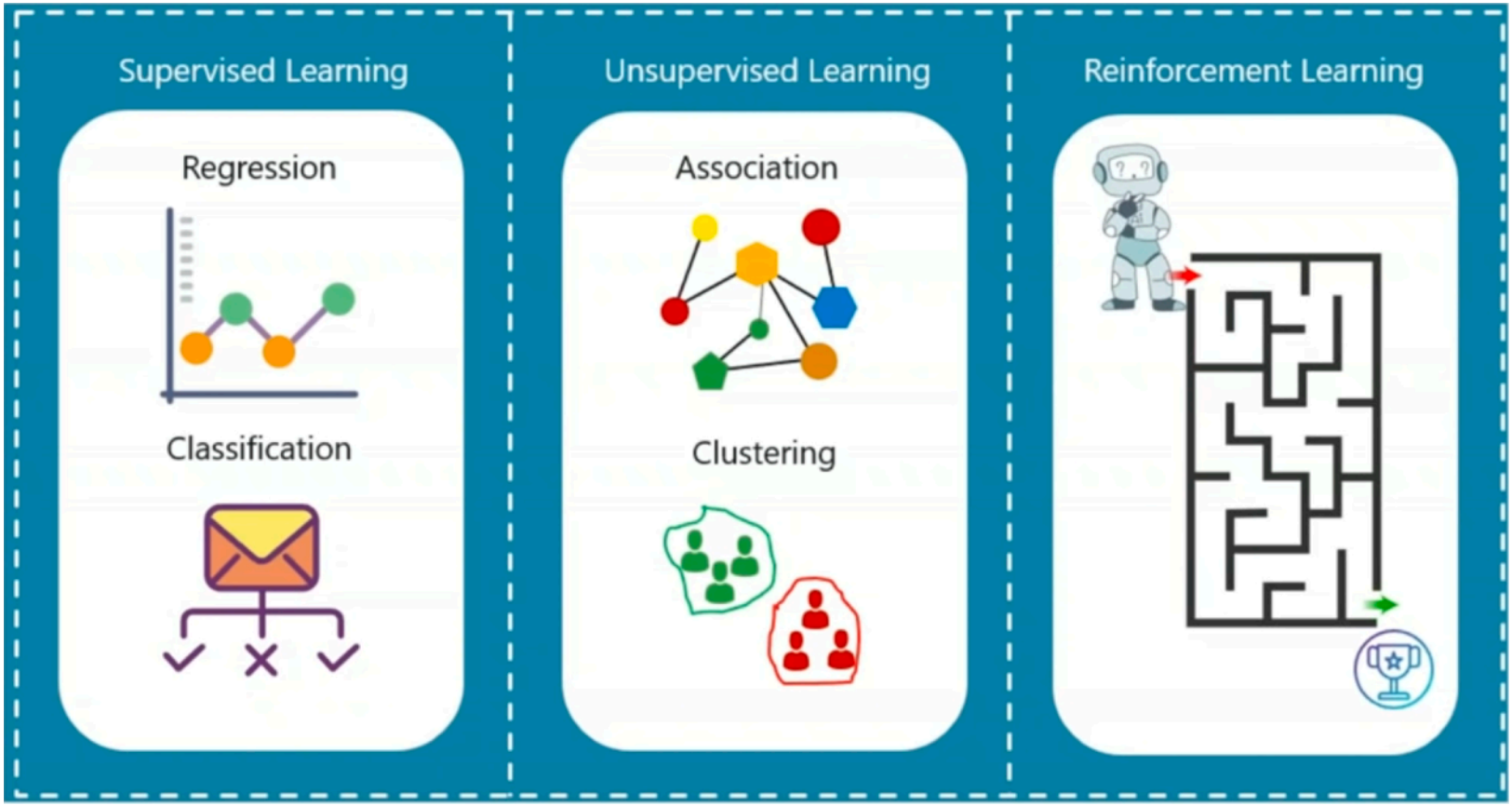
Problem Statement: Place an agent in any one of the rooms (0,1,2,3,4) and the goal is to reach outside the building (room 5)





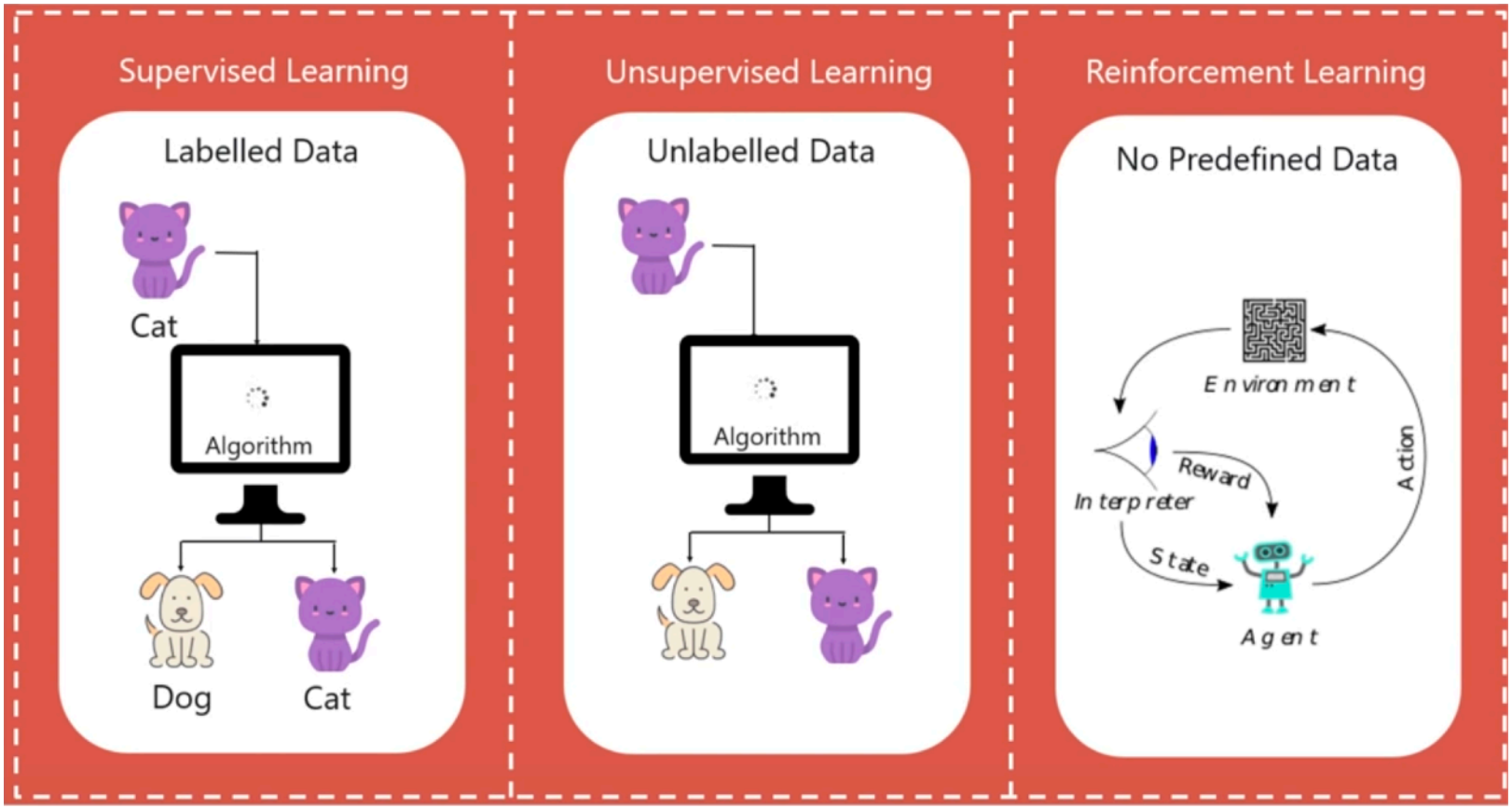
https://www.youtube.com/watch?v=xtOg44r6dsE&t=902s

Type of Problems



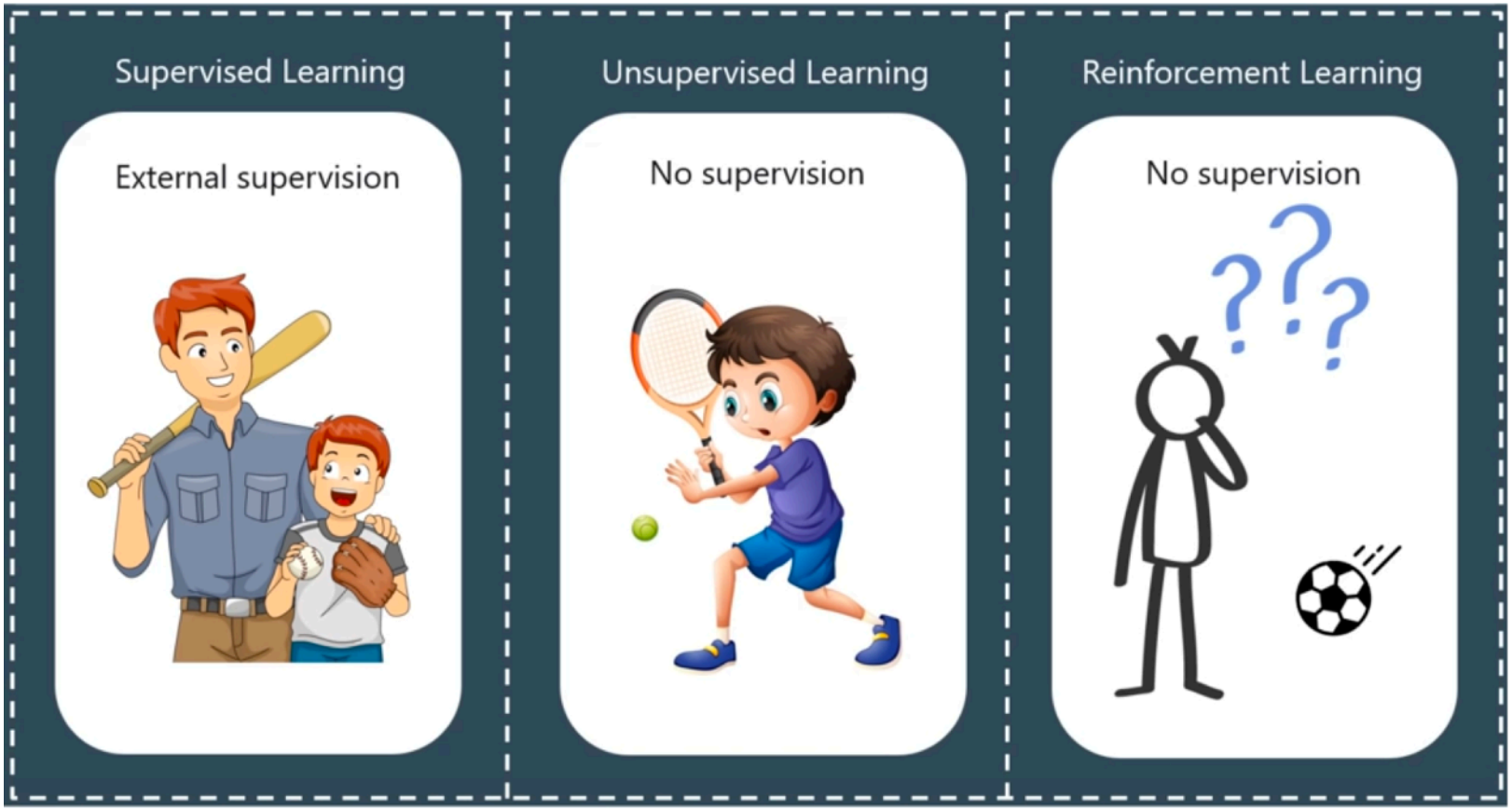
Source: Edureka!

Type of Data



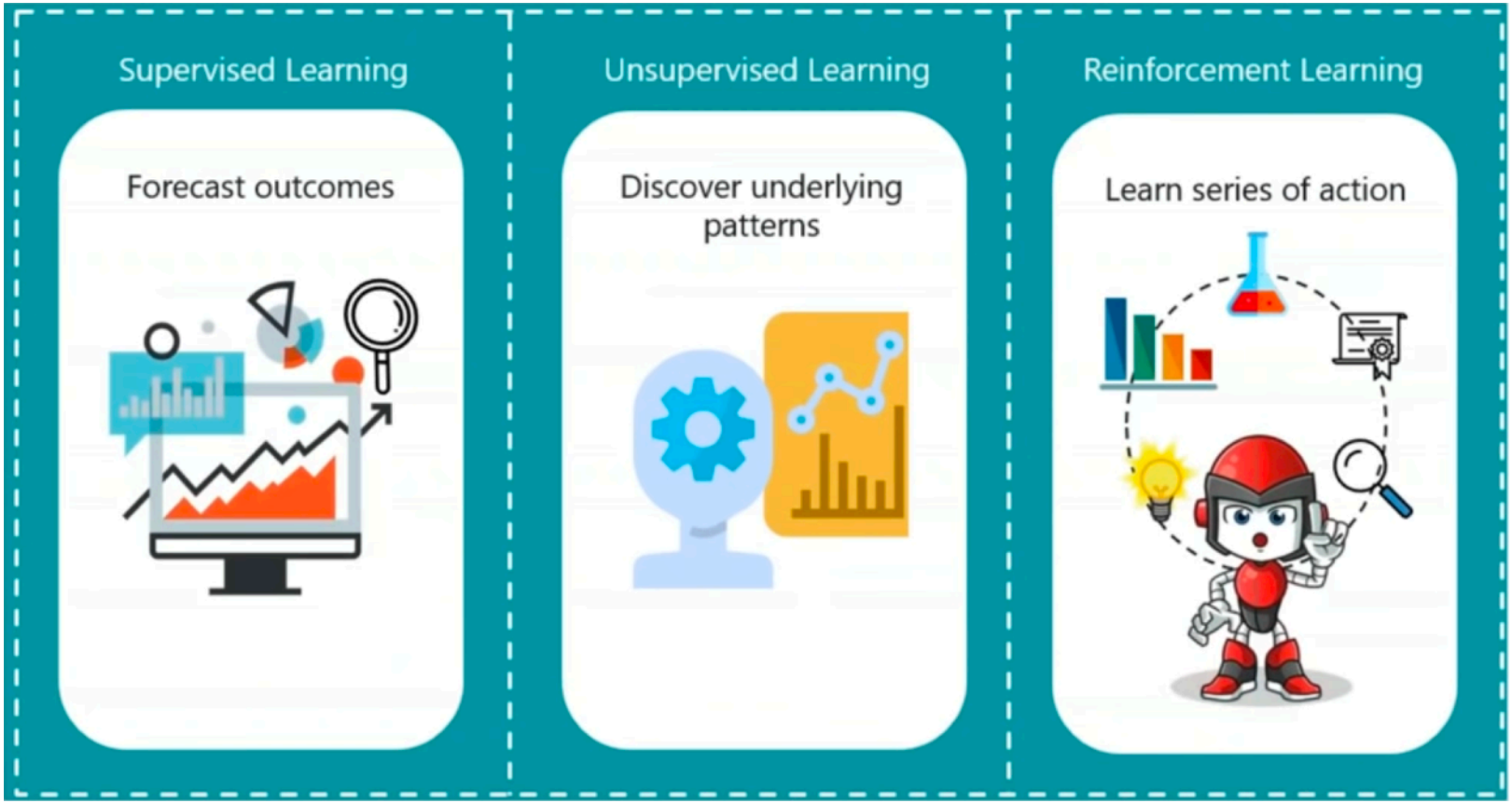
Source: Edureka!

Training



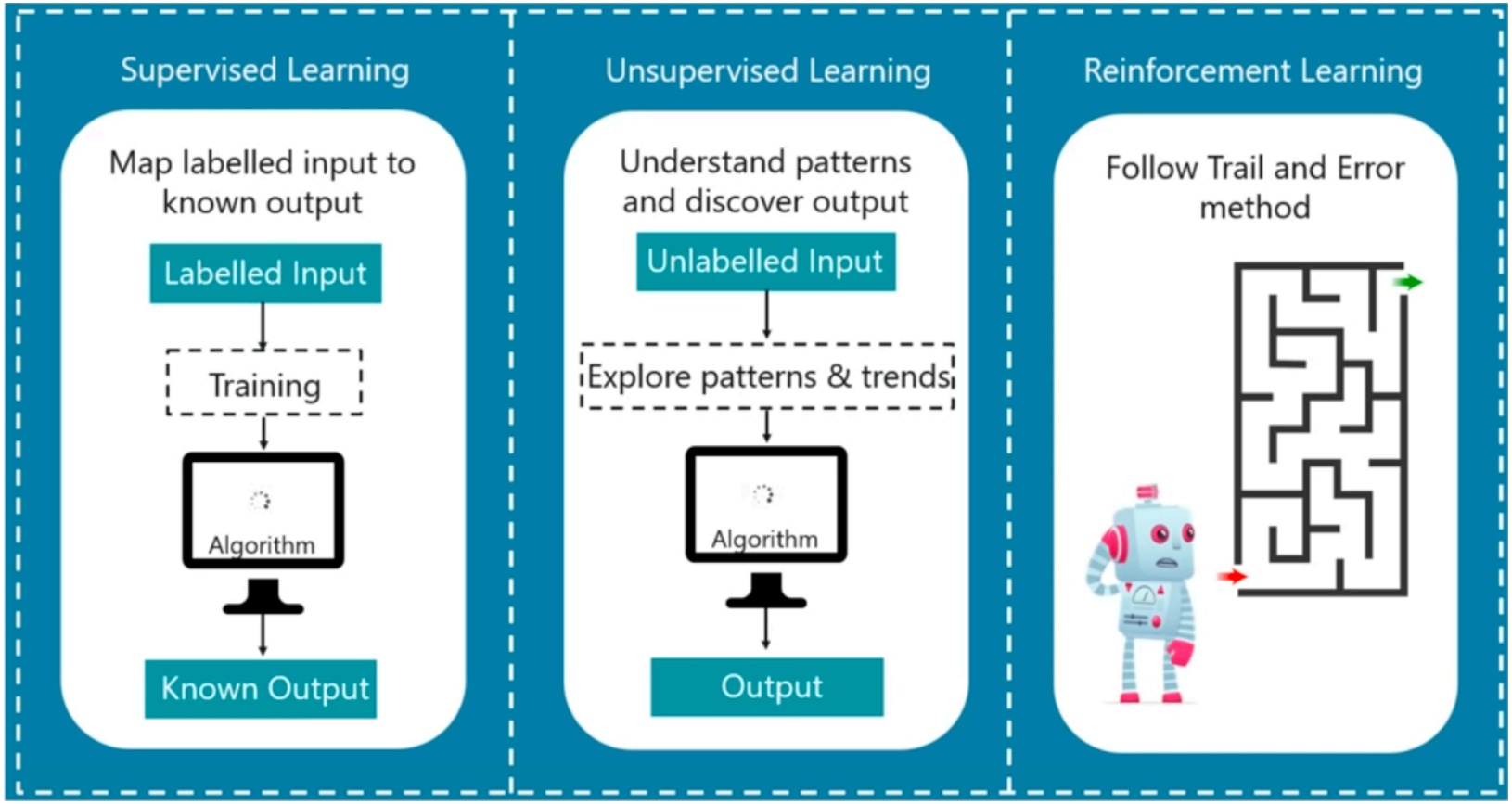
Source: Edureka!

Aim



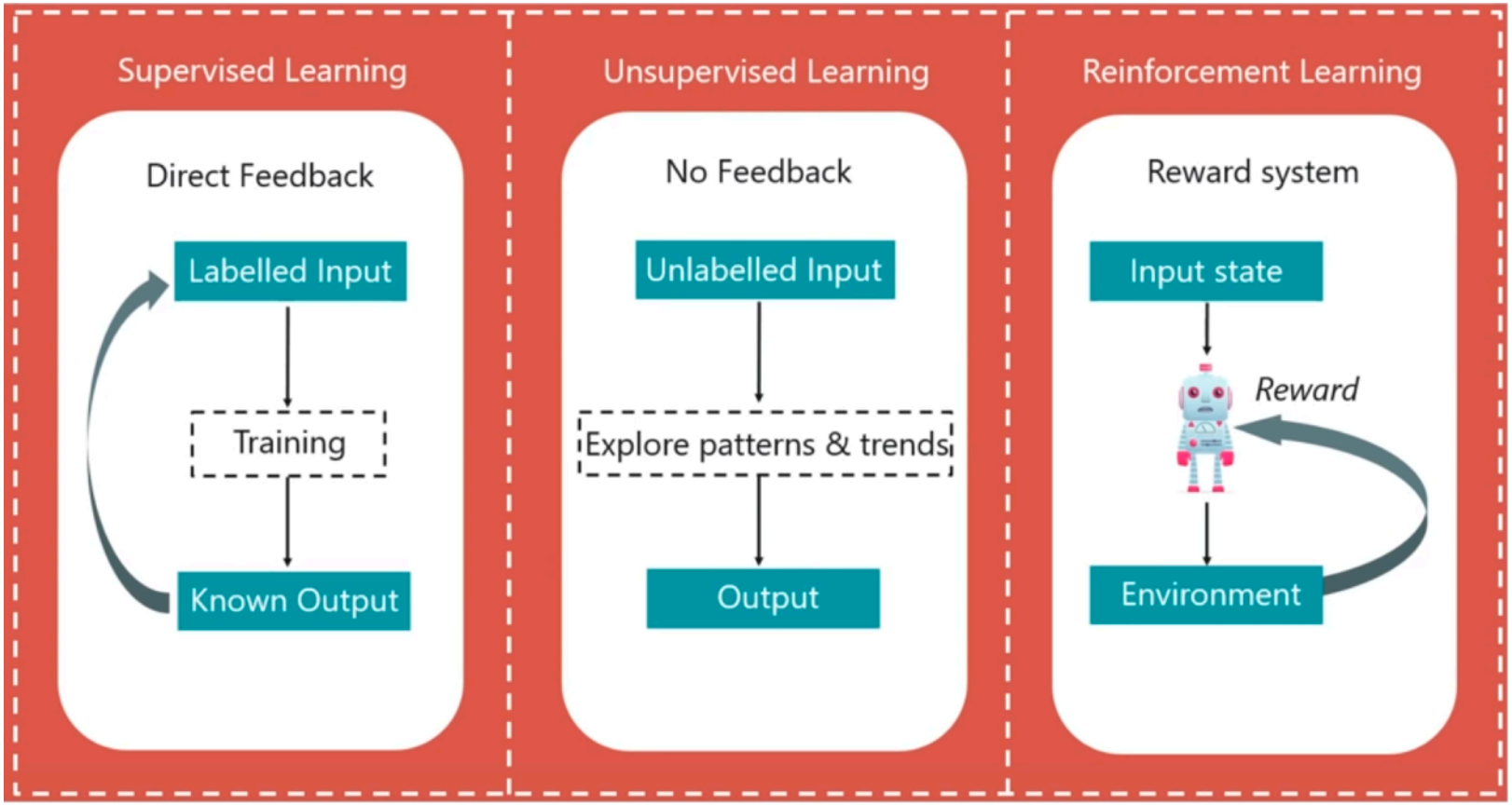
Source: Edureka!

Approach



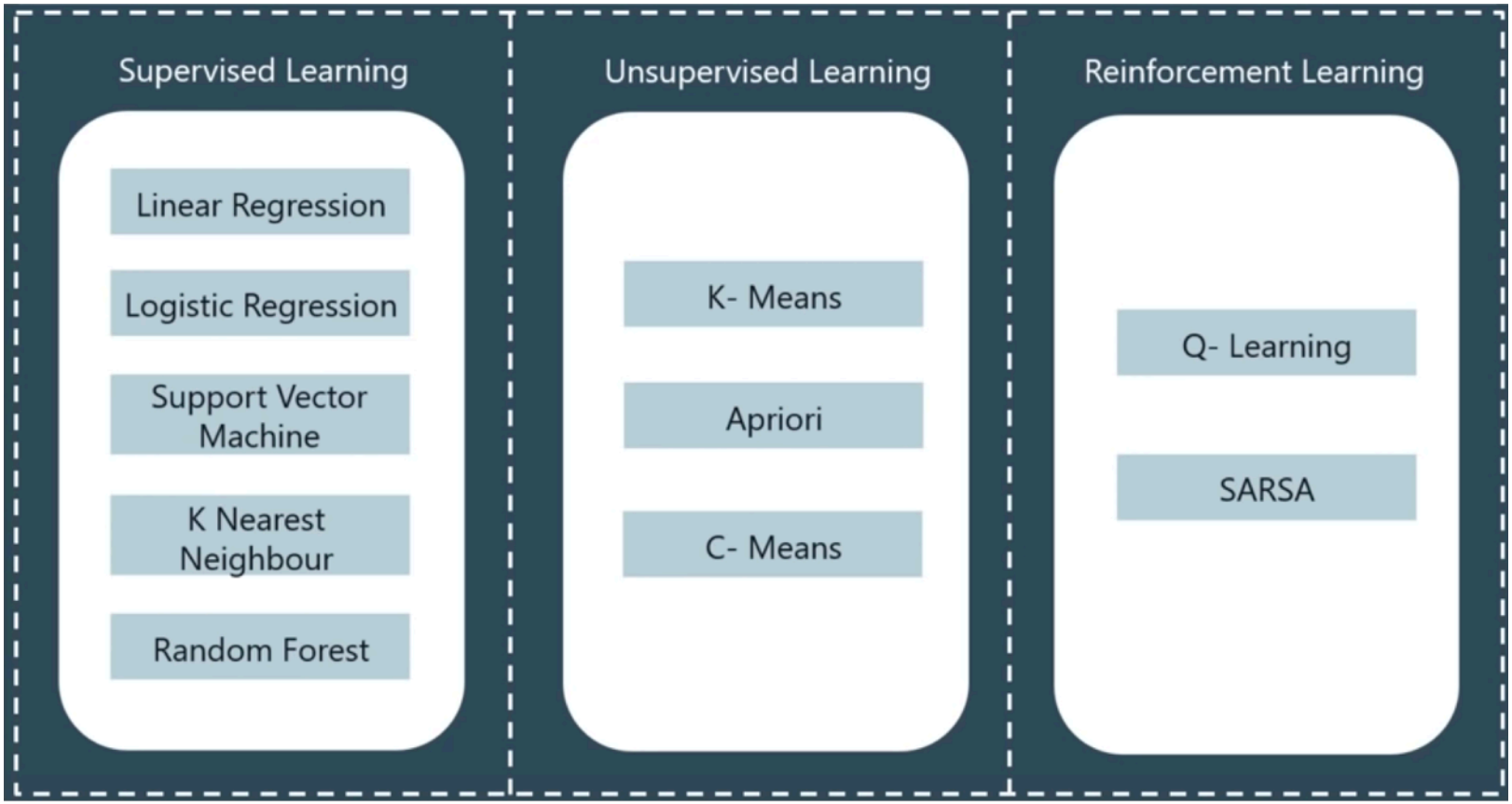
Source: Edureka!

Output Feedback



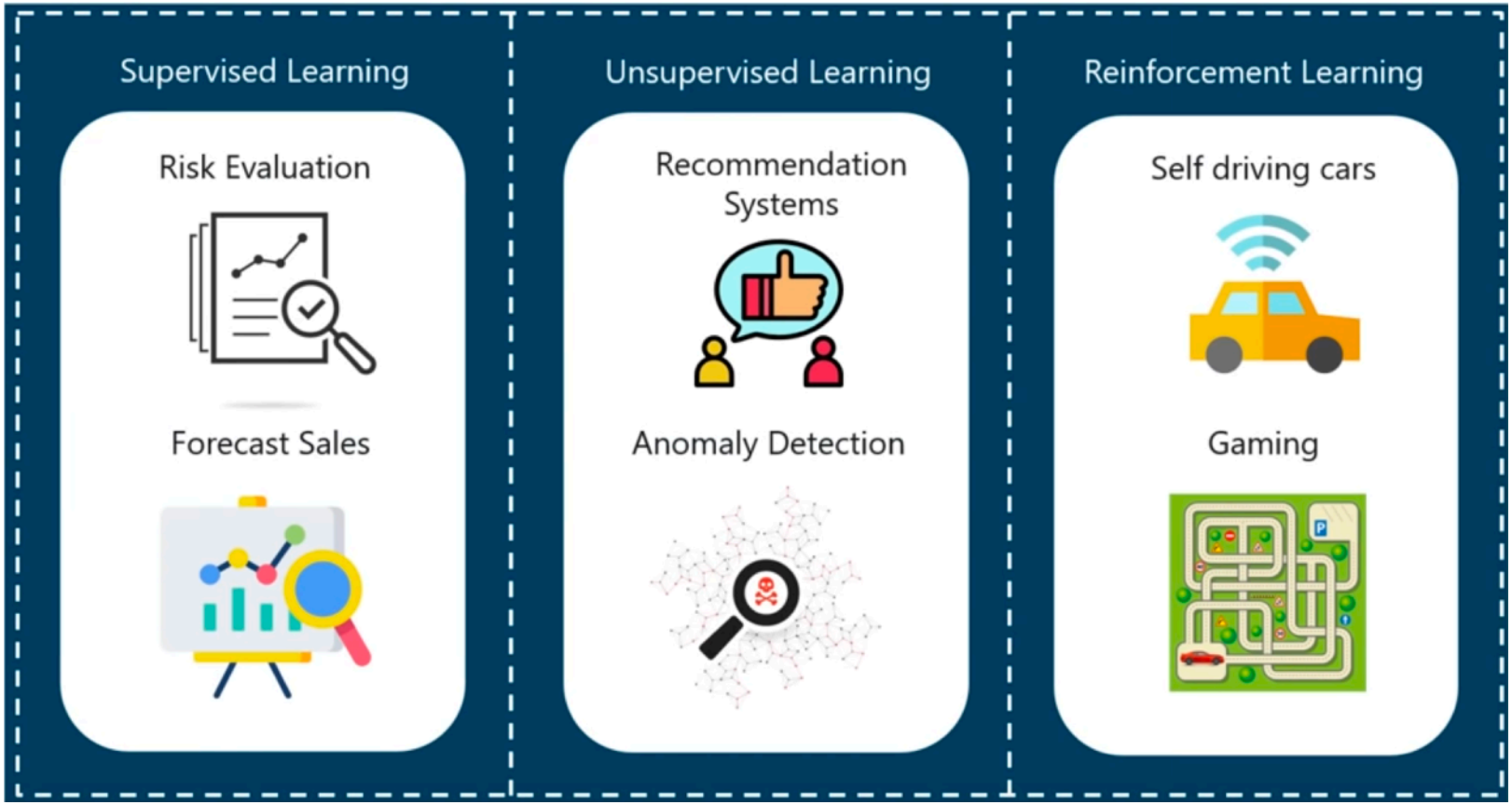
Source: Edureka!

Popular Algorithms



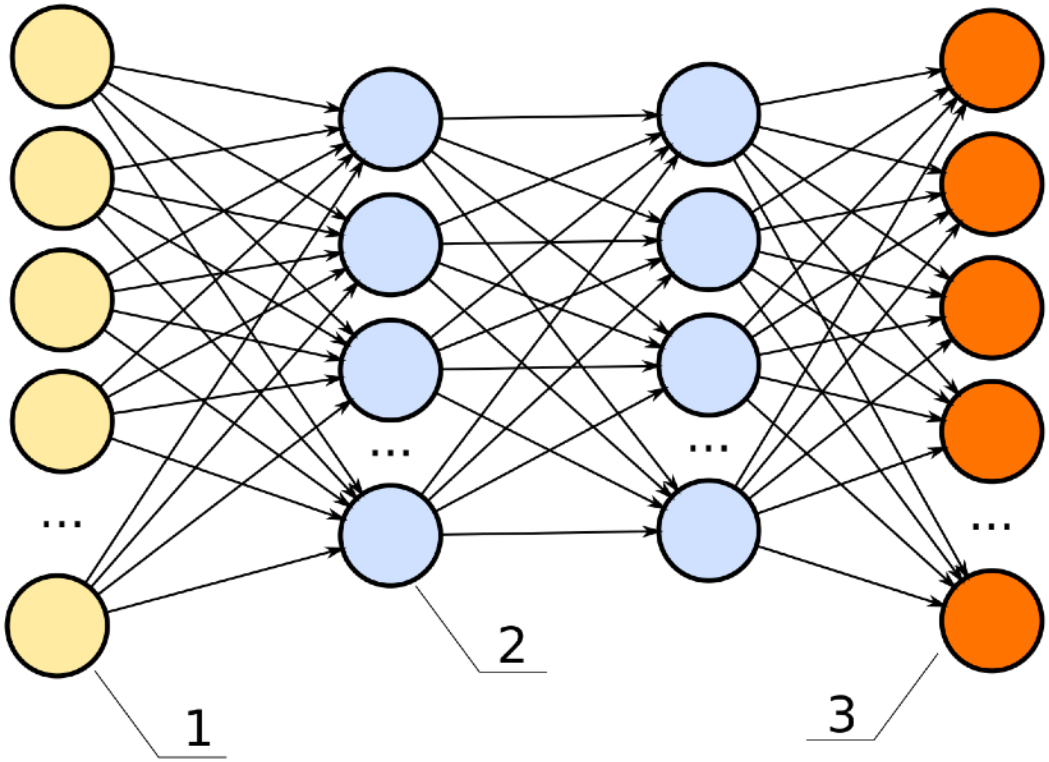
Source: Edureka!

Applications



Source: Edureka!

"Deep Learning is a sub-field of Machine Learning which utilises the artificial neural network model to learn on its own from data. The reason that it is called "deep" is because the network is formed with many layers of connection. Output from one layer will become input to another layer, leading to the solution of the problem."



Source: Wikipedia

Comparison Between PA, ML & DL

	Predictive Analytics	Machine Learning	Deep Learning
Supervised Learning	X	X	X
Unsupervised Learning		X	X
Reinforcement Learning		X	X
Data Intensive			X
GPU Processing Intensive			X

©Simplilearn. All rights reserved.

simplilearn

https://www.youtube.com/watch?v=6M5VXKLf4D4

How does Machine Learning fit into Data Science?

TA SCIENCE

https://www.youtube.com/watch?v=nAWUFkDJLIs

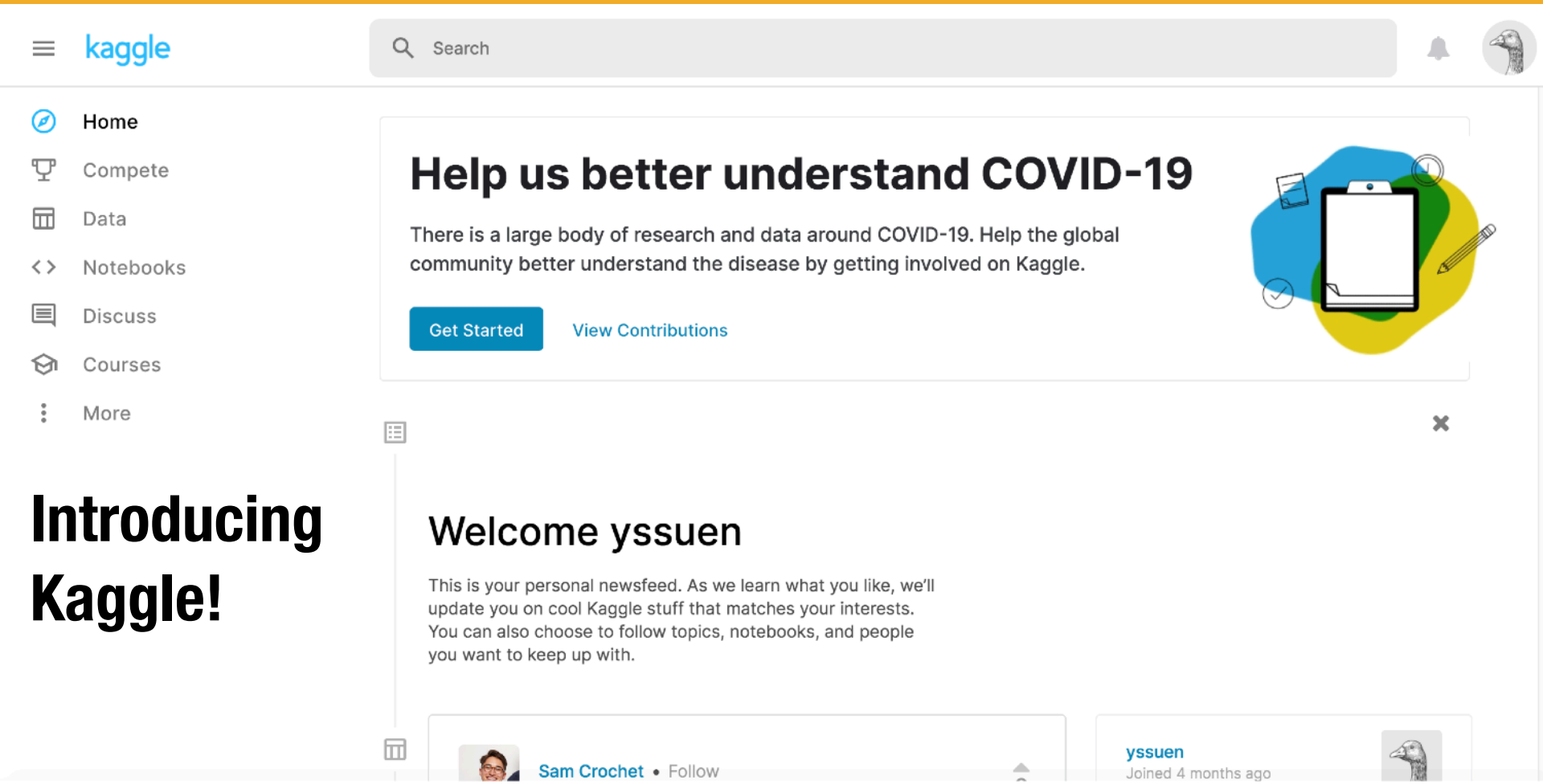
Business Problem

Predict Outcome				Data Analysis
Data Rich			Data Poor	Geospatial
Numeric		Classification	A/B Test	Segmentation
Continuous	Count	Binary	Non Binary	Aggregation
• Linear Regression • Decision Tree • Forest Model • Boosted Model	• Count Regression	• Logistic Regression • Decision Tree	• Forest Model • Booted Model	Descriptive

Source: Udacity Model Selection Methodology Map

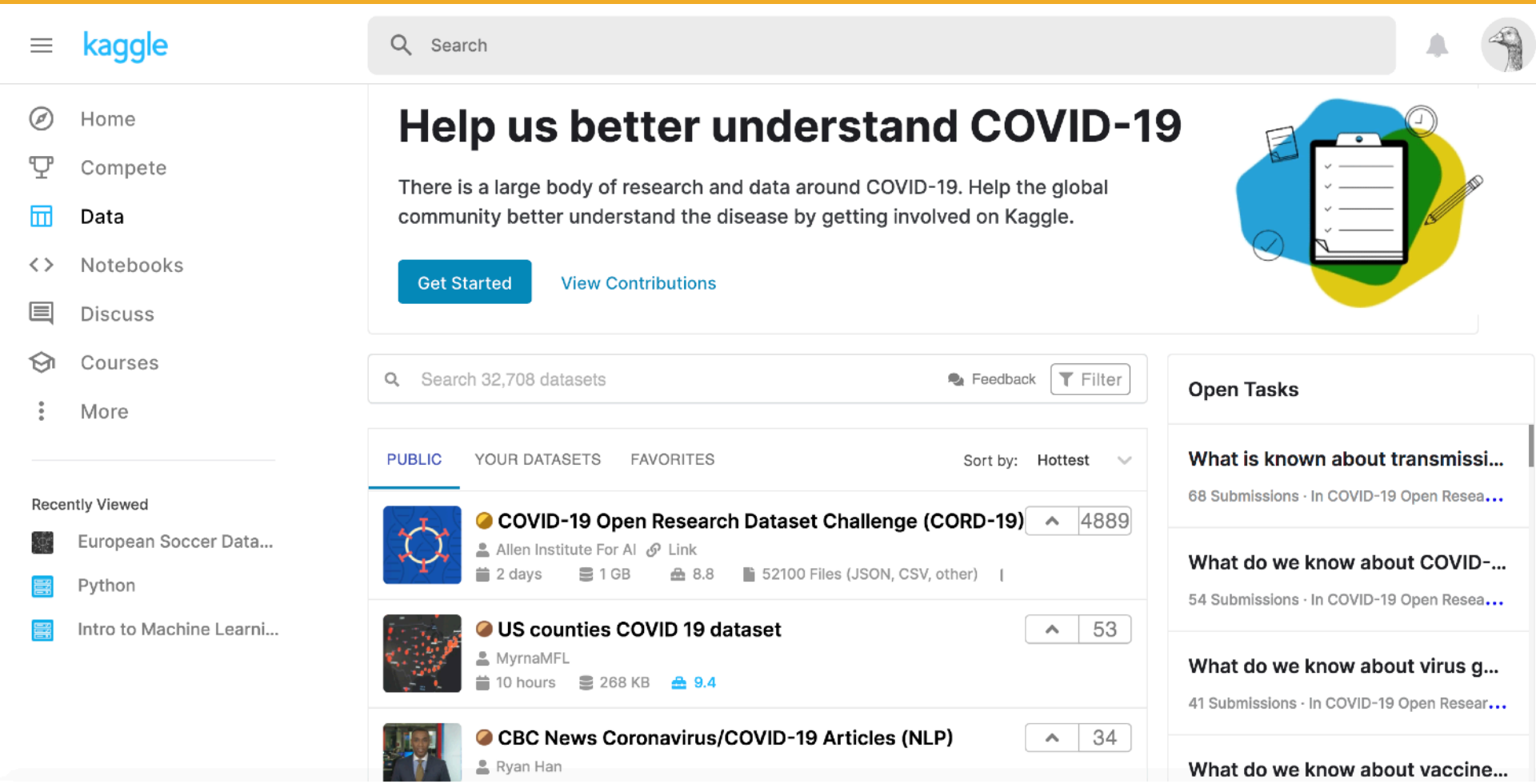
How to get started?

Introducing Kaggle!

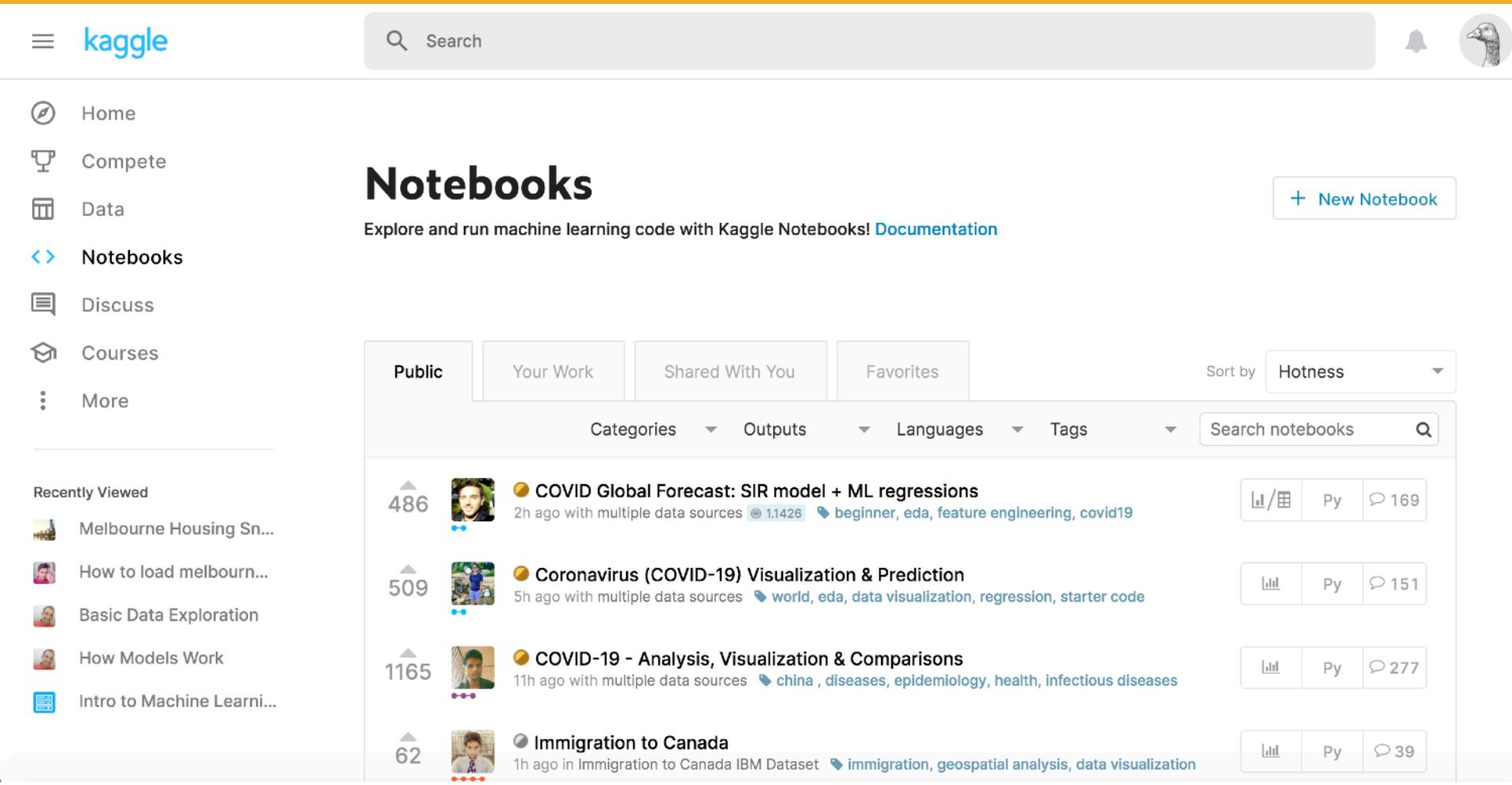


https://www.kaggle.com

https://www.youtube.com/watch?v=FloMHMOU5Bs&t=144s



https://www.kaggle.com/datasets



https://www.kaggle.com/notebooks

kaggle

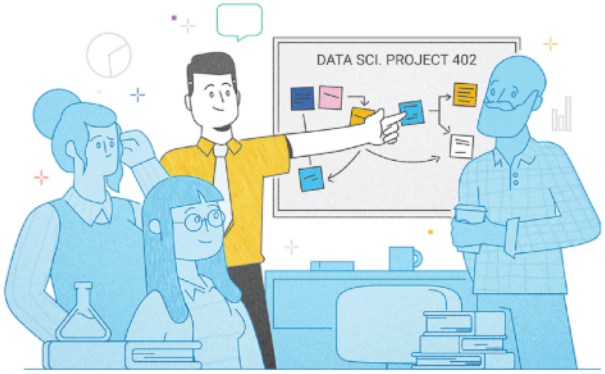
Search

- Home
- Compete
- Data
- Notebooks
- Discuss
- Courses
- More

Faster Data Science Education

Practical data skills you can apply immediately: that's what you'll learn in these free micro-courses.

They're the fastest (and most fun) way to become a data scientist or improve your current skills.



https://www.kaggle.com/learn/overview

Let’s do your 1st exercise on Kaggle!

kaggle

Search

- Home
- Compete
- Data
- Notebooks
- Discuss
- Courses
- More

Recently Viewed

- Intro to Machine Learni...
- Titanic, a step-by-step...
- Titanic ML step-by-ste...
- Titanic: Machine Learn...
- Titanic Tutorial

Lessons

Tutorial Exercise

1 **How Models Work**
The first step if you're new to machine learning

2 **Basic Data Exploration**
Load and understand your data

3 **Your First Machine Learning Model**
Building your first model. Hurray!

4 **Model Validation**
Measure the performance of your model ? so you can test and compare alternatives

Your Progress

0% Begin today!

Prerequisite Skills:
Python

Prepares you for these Learn Micro-Courses:
Deep Learning,
Machine Learning Explainability,
Intermediate Machine Learning

Instructor

Dan Becker
Data Scientist

Using Pandas to Get Familiar With Your Data

The first step in any machine learning project is familiarize yourself with the data. You'll use the Pandas library for this. Pandas is the primary tool data scientists use for exploring and manipulating data. Most people abbreviate pandas in their code as `pd`. We do this with the command

In [1]:
```
import pandas as pd
```

The most important part of the Pandas library is the DataFrame. A DataFrame holds the type of data you might think of as a table. This is similar to a sheet in Excel, or a table in a SQL database.

Pandas has powerful methods for most things you'll want to do with this type of data.

As an example, we'll look at data about home prices in Melbourne, Australia. In the hands-on exercises, you will apply the same processes to a new dataset, which has home prices in Iowa.

New Notebook

Select new notebook settings

You can change these settings at any time

Select language

Python

Select type

Notebook

Ideal for interactive data exploration and polished analysis. Shares insights through code & commentary

Script

Ideal for fitting a model and competition submissions. Shares code for review and RMarkdown reports

SHOW ADVANCED SETTINGS

Create

kernel5d8074e8... *Draft saved*

File Edit Insert View Run Add-ons Help

Share | Save Version 0

Run All | *Draft Session (44m)* | HDD CPU RAM

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

# Input data files are available in the "../input/" directory.
# For example, running this (by clicking run or pressing Shift+Enter) will list all files under the input directo

# save filepath to variable for easier access
melbourne_file_path = '../input/melbourne-housing-snapshot/melb_data.csv'
# read the data and store data in DataFrame titled melbourne_data
melbourne_data = pd.read_csv(melbourne_file_path)
# print a summary of the data in Melbourne data
melbourne_data.describe()
```

Out[4]:

	Rooms	Price	Distance	Postcode	Bedroom2	Bathroom	Car	Landsize	BuildingArea	Y
count	13580.000000	1.358000e+04	13580.000000	13580.000000	13580.000000	13580.000000	13518.000000	13580.000000	7130.000000	8205
mean	2.937997	1.075684e+06	10.137776	3105.301915	2.914728	1.534242	1.610075	558.416127	151.967650	1964
std	0.955748	6.393107e+05	5.868725	90.676964	0.965921	0.691712	0.962634	3990.669241	541.014538	37
min	1.000000	8.500000e+04	0.000000	3000.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1196
25%	2.000000	6.500000e+05	6.100000	3044.000000	2.000000	1.000000	1.000000	177.000000	93.000000	1940

Sessions

Interactive 3/10 CPU 0/1 GPU 0/1 TPU

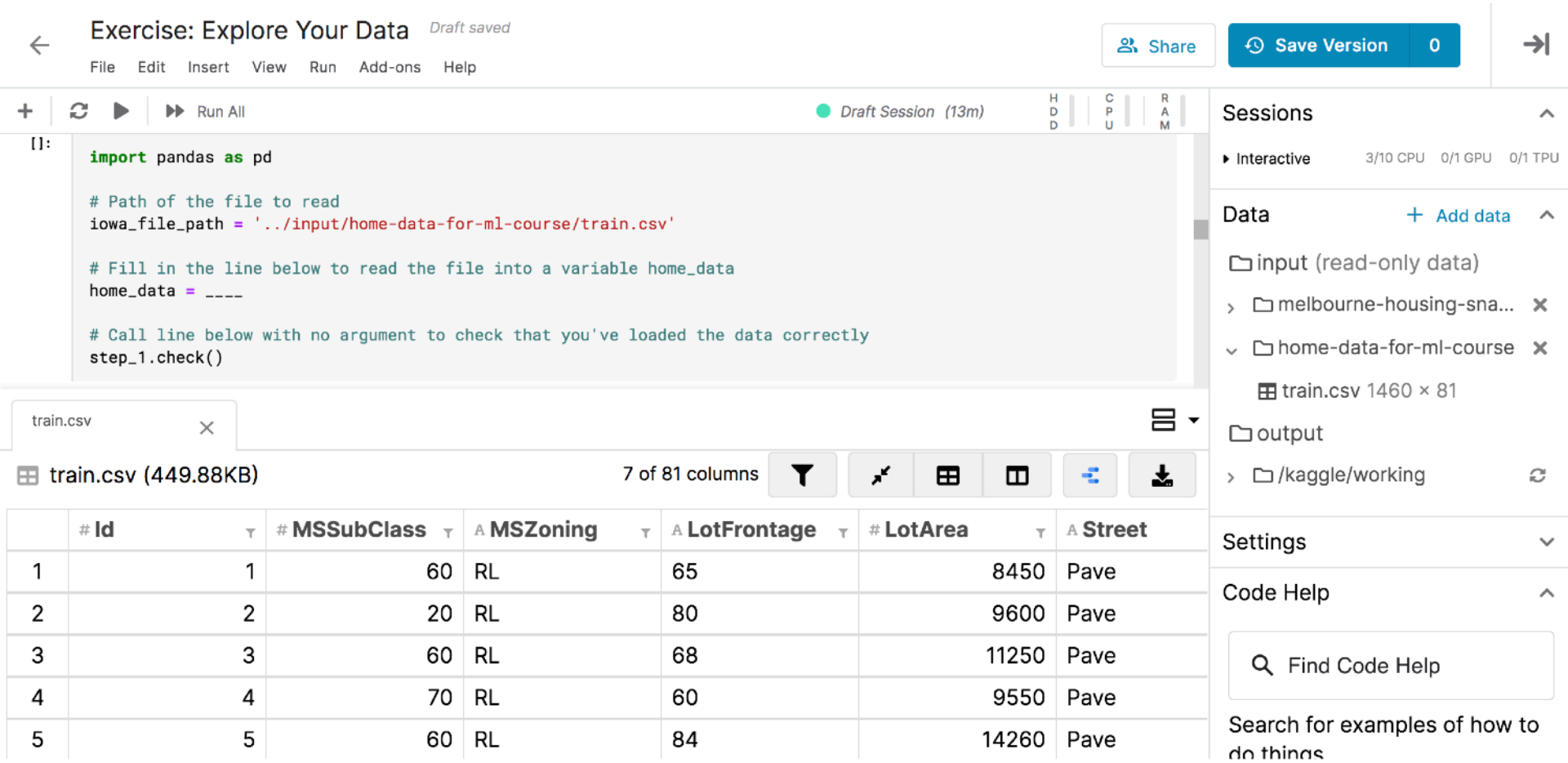
Data + Add data

Settings

Code Help

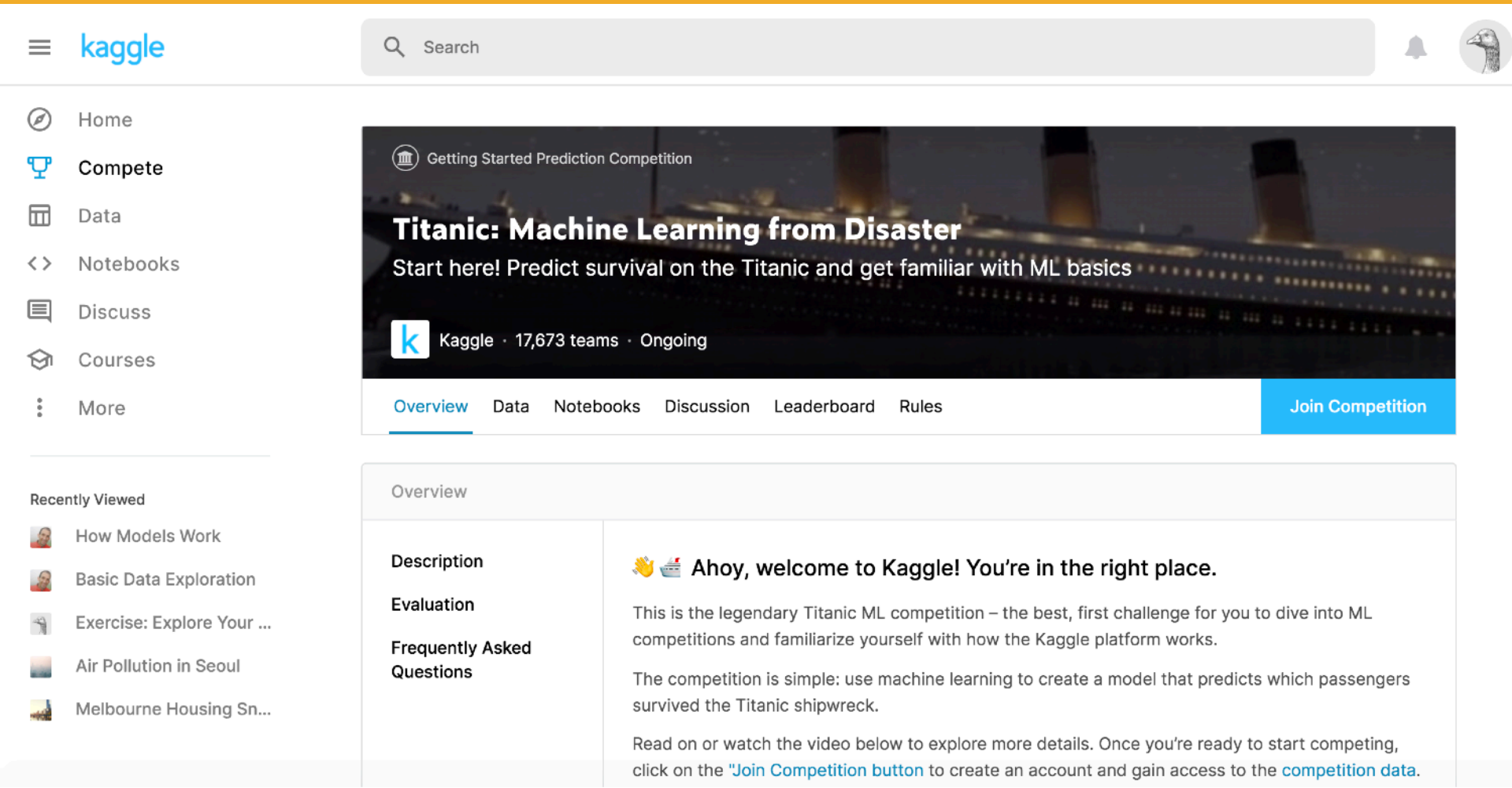
Find Code Help

Search for examples of how to do things



	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street
1	1	60	RL	65	8450	Pave
2	2	20	RL	80	9600	Pave
3	3	60	RL	68	11250	Pave
4	4	70	RL	60	9550	Pave
5	5	60	RL	84	14260	Pave

When you're ready, join the Kaggle competition.



https://www.kaggle.com/c/titanic/


kaggle
How to Get Started with Kaggle's Titanic Machine Learning Competition

WIKIPEDIA
The Free Encyclopedia

Article | Talk | Read | Edit | View history | Search Wikipedia

Iris flower data set

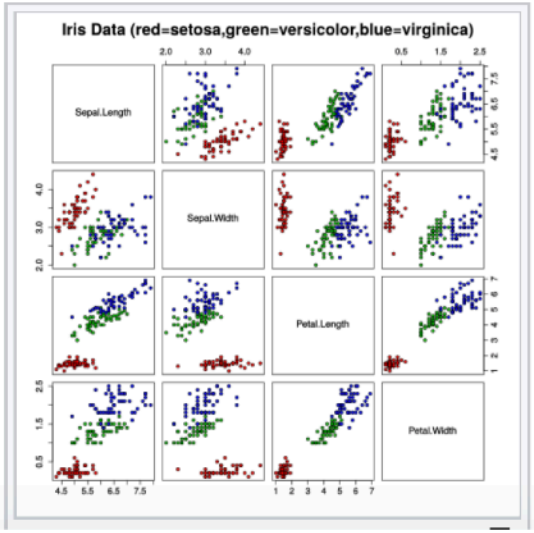
From Wikipedia, the free encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction

Help
About Wikipedia

The ***Iris* flower data set** or **Fisher's *Iris* data set** is a multivariate data set introduced by the British statistician and biologist Ronald Fisher in his 1936 paper *The use of multiple measurements in taxonomic problems* as an example of linear discriminant analysis.[1] It is sometimes called **Anderson's *Iris* data set** because Edgar Anderson collected the data to



https://en.wikipedia.org/wiki/Iris_flower_data_set

Traiing the Iris Dataset with 3 ML Models

In [10]:
```
# Begin by importing all necessary libraries
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score
from sklearn.neighbors import KNeighborsClassifier # K-Nearest Neibour Model
from sklearn.naive_bayes import GaussianNB         # Gaussian Naive Bayes Model
from sklearn.svm import SVC                        # Support Vector Machine Model
```

In [11]:
```
data = pd.read_csv('iris.csv')
# Check the first 5 entries
print(data.head(5))
```

Features **Labels**

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

Define the features and labels in Pandas by slicing the data table and choosing certain rows/columns with iloc(). The slicing notation above selects every row and every column except the last column which contains the label of the species).

Define the features and labels in Pandas by slicing the data table and choosing certain rows/columns with iloc(). The slicing notation above selects every row and every column except the last column which contains the label of the species).

In [12]:
```
data.drop('Id', axis=1, inplace=True)
print(data.head(5))
# Pandas ".iloc" expects row_indexer, column_indexer
X = data.iloc[:,:-1].values
# Alternate way of selecting columns:
# X = data.iloc['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm']
# Specify which column we want for the target/labels.
y = data['Species']
```

Features **Labels**

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
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

Next split the data into training and testing sets using the train_test_split() function

In [13]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=27)
```

In [14]:
```
print(X_train)
print(y_train)
```

```
[6.3 2.9 5.6 1.8]
[5.9 3.2 4.8 1.8]
[7.2 3.2 6.  1.8]
```

```
[6.2 2.9 4.3 1.3]
[6.7 3.1 4.7 1.5]
[4.4 3.  1.3 0.2]
[6.8 2.8 4.8 1.4]
[6.3 2.5 5.  1.9]
[5.8 2.7 3.9 1.2]
[4.8 3.1 1.6 0.2]
[4.6 3.4 1.4 0.3]
[6.3 2.9 5.6 1.8]
[5.9 3.2 4.8 1.8]
[7.2 3.2 6.  1.8]
[4.6 3.6 1.  0.2]
[6.  2.9 4.5 1.5]
[6.8 3.  5.5 2.1]
[7.4 2.8 6.1 1.9]
[6.1 2.9 4.7 1.4]
[5.  3.5 1.6 0.6]
[5.4 3.7 1.5 0.2]
[5.1 3.5 1.4 0.2]
```

Specify the training models (i.e. SVC, KNN, GNB)

In [15]:

```
# SVC_model = svm.SVC()
SVC_model = SVC(gamma='scale')
# KNN model requires you to specify n_neighbors,
# the number of points the classifier will look at to determine what class a new point belongs to
KNN_model = KNeighborsClassifier(n_neighbors=5)
# KNN_model = KNeighborsClassifier(algorithm='scale', leaf_size=30, metric='minkowski',
#                     metric_params=None, n_jobs=None, n_neighbors=5, p=2)
# Gaussian NB model
GNB_model = GaussianNB()
```

Fitting the model means training it to identify the relationship between features and labels.

Fittig the models

In [16]:
```
SVC_model.fit(X_train, y_train)
KNN_model.fit(X_train, y_train)
GNB_model.fit(X_train, y_train)
```

Out[16]:
```
GaussianNB(priors=None, var_smoothing=1e-09)
```

Training completed. Use models to predict and store the outcome

In [17]:
```
SVC_prediction = SVC_model.predict(X_test)
KNN_prediction = KNN_model.predict(X_test)
GNB_prediction = GNB_model.predict(X_test)
```

Model Evaluation Using Accuracy Score and Confusion Matrix

In [18]:
```
# Accuracy score is the simplest way to evaluate
print(accuracy_score(SVC_prediction, y_test))
print(accuracy_score(KNN_prediction, y_test))
print(accuracy_score(GNB_prediction, y_test))
# But Confusion Matrix and Classification Report give more details about performance
print(confusion_matrix(SVC_prediction, y_test))
print(classification_report(KNN_prediction, y_test))
print(accuracy_score(GNB_prediction, y_test))
```

```
0.9333333333333333
0.9666666666666667
0.9
[[ 7  0  0]
 [ 0 10  1]
 [ 0  1 11]]
                 precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00         7
Iris-versicolor       1.00      0.92      0.96        12
 Iris-virginica       0.92      1.00      0.96        11

       accuracy                           0.97        30
      macro avg       0.97      0.97      0.97        30
   weighted avg       0.97      0.97      0.97        30

0.9
```

Model Evaluation Using Accuracy Score and Confusion Matrix

```
# Accuracy score is the simplest way to evaluate
print(accuracy_score(SVC_prediction, y_test))
print(accuracy_score(KNN_prediction, y_test))
print(accuracy_score(GNB_prediction, y_test))
# But Confusion Matrix and Classification Report give more details about performance
print(confusion_matrix(SVC_prediction, y_test))
print(classification_report(KNN_prediction, y_test))
print(accuracy_score(GNB_prediction, y_test))
```

```
0.9333333333333333
0.9666666666666667
0.9
[[ 7  0  0]
 [ 0 10  1]
 [ 0  1 11]]
                 precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00         7
Iris-versicolor       1.00      0.92      0.96        12
 Iris-virginica       0.92      1.00      0.96        11

       accuracy                           0.97        30
      macro avg       0.97      0.97      0.97        30
   weighted avg       0.97      0.97      0.97        30

0.9
```

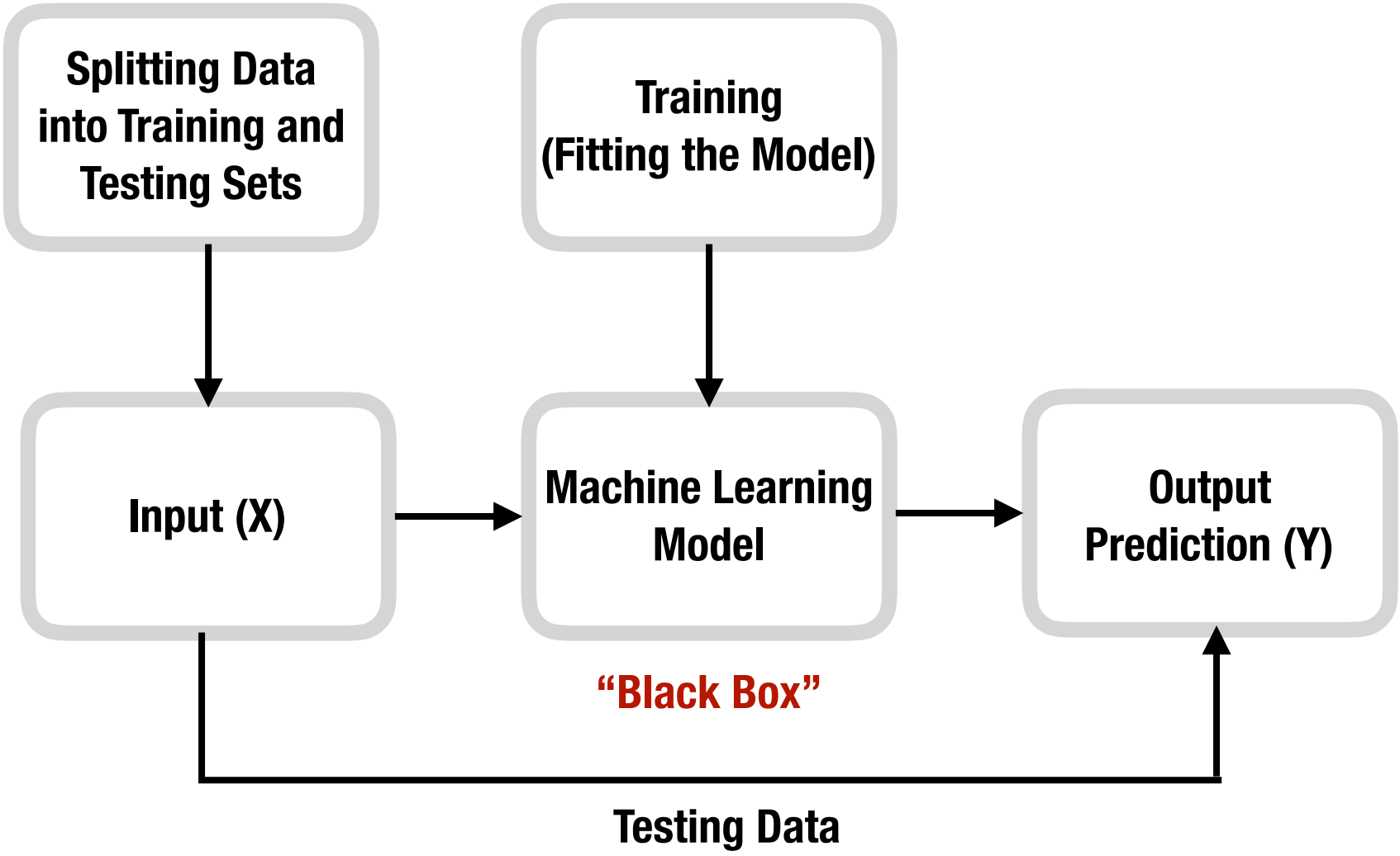
Accuracy score of each model

SVC accuracy: 0.9333333333333333 KNN accuracy: 0.9666666666666667 GNB accuracy: 0.9

CRISP-DM Framework and Methodology Map

Supervised, Unsupervised and Reinforcement Learning

Applications of ML as a Black Box

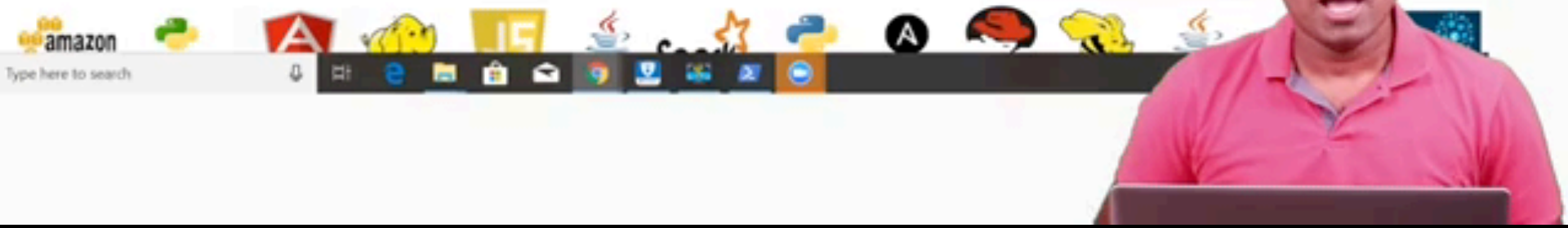

Splitting Data into Training and Testing Sets
Training (Fitting the Model)
Input (X)
Machine Learning Model
Output Prediction (Y)
“Black Box”
Testing Data

zekeLabs

Black Box Introduction to Machine Learning

Learning made Simpler !

www.zekeLabs.com



THANK YOU FOR YOUR TIME!